UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS

Semantic perspectives for learning over biomedical knowledge graphs

" Documento Definitivo"

Doutoramento em Informática

Rita Isabel Torres de Sousa

Tese orientada por:
Prof.a Doutora Cátia Luísa Santana Calisto Pesquita
Prof.a Doutora Sara Guilherme Oliveira da Silva

Documento especialmente elaborado para a obtenção do grau de doutor

2023

UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

Semantic perspectives for learning over biomedical knowledge graphs

Doutoramento em Informática

Rita Isabel Torres de Sousa

Tese orientada por:

Prof.a Doutora Cátia Luísa Santana Calisto Pesquita

Prof.a Doutora Sara Guilherme Oliveira da Silva

Júri:

Presidente:

- Doutor Manuel João Caneira Monteiro da Fonseca, Professor Associado com Agregação Faculdade de Ciências da Universidade de Lisboa

Vogais:

- Doutor João Rafael Landeiro de Sousa Gonçalves, Associate Director of Knowledge Representation Center for Computational Biomedicine da Harvard Medical School
- Doutora Mehwish Alam, Associate Professor Télécom Paris do Institut Polytechnique de Paris
- Doutora María-Esther Vidal Serodio, Full Professor Leibniz Universität Hannover
- Doutor André Osório e Cruz De Azerêdo Falcão, Professor Associado Faculdade de Ciências da Universidade de Lisboa
- Doutora Cátia Luísa Santana Calisto Pesquita, Professora Associada Faculdade de Ciências da Universidade de Lisboa

Documento especialmente elaborado para a obtenção do grau de doutor

Esta tese é financiada pela Fundação para a Ciência e a Tecnologia através da Unidade de Investigação LASIGE, UIDB/00408/2020 e UIDP/00408/2020, da Bolsa de Doutoramento SFRH/BD/145377/2019.

2023

Acknowledgements

In the first place, I would like to express my gratitude to my supervisors, **Professor Cátia Pesquita** and **Professor Sara Silva**. From day one to the very end, their dedication, guidance, patience and support in all aspects were truly remarkable. I could not have wished for more inspiring role models in my academic journey. I hope that now, without my endless meetings with thousands of questions, they will have the rest they deserve. I express my deep gratitude to **Heiko Paulheim** and to all my colleagues in Mannheim for their warm welcome and the incredibly enriching experience they provided during my time in Germany. I also want to thank **Daniel Faria** for offering invaluable advice and the opportunity to teach. It was a privilege to work with him.

To my every day **LASIGE colleagues/friends**, and especially to those who share my office. I have spent so many hours in the office that I almost feel I can call them roommates. Every conversation, whether about work or personal matters, holds a special place in my memories. I extend my gratitude to **Alexandra** and **Carla** for their infinite patience in addressing all my questions, for the care they always showed me, and, of course, for their helpful advice. My appreciation also goes to my **LISEDA colleagues**, who taught me so many things. Witnessing the arrival of new students and observing their progress has been a genuine pleasure. A heartfelt thank you to **Chalupas** for the countless coffee breaks, after-work dinners, and other shared moments that undeniably added a layer of fun and joy to my Ph.D. journey. A special acknowledgement is reserved for my Ph.D. sister, **Diana**. This journey would not have been the same without her by my side every step of the way.

I am profoundly grateful to my **family** and **friends** for their constant love and encouragement. Without the support of my **mother**, my **father** and my **sister**, I would not be delivering this thesis. They provided me with everything I needed to pursue my dreams. To my beloved **grandparents**, I hope they are proud of me. They keep on inspiring me, shaping the person I am today.

Finally, I would like to thank **Fundação para a Ciência e a Tecnologia**, which provides the funding under LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020) and through my Ph.D. grant (ref. SFRH/BD/145377/2019).

Since this work is about knowledge graphs, I believe an illustration of a graph with different types of relationships and node types is required. Each node in Figure 1 represents a special person who marked my Ph.D. journey and whom I would like to thank.

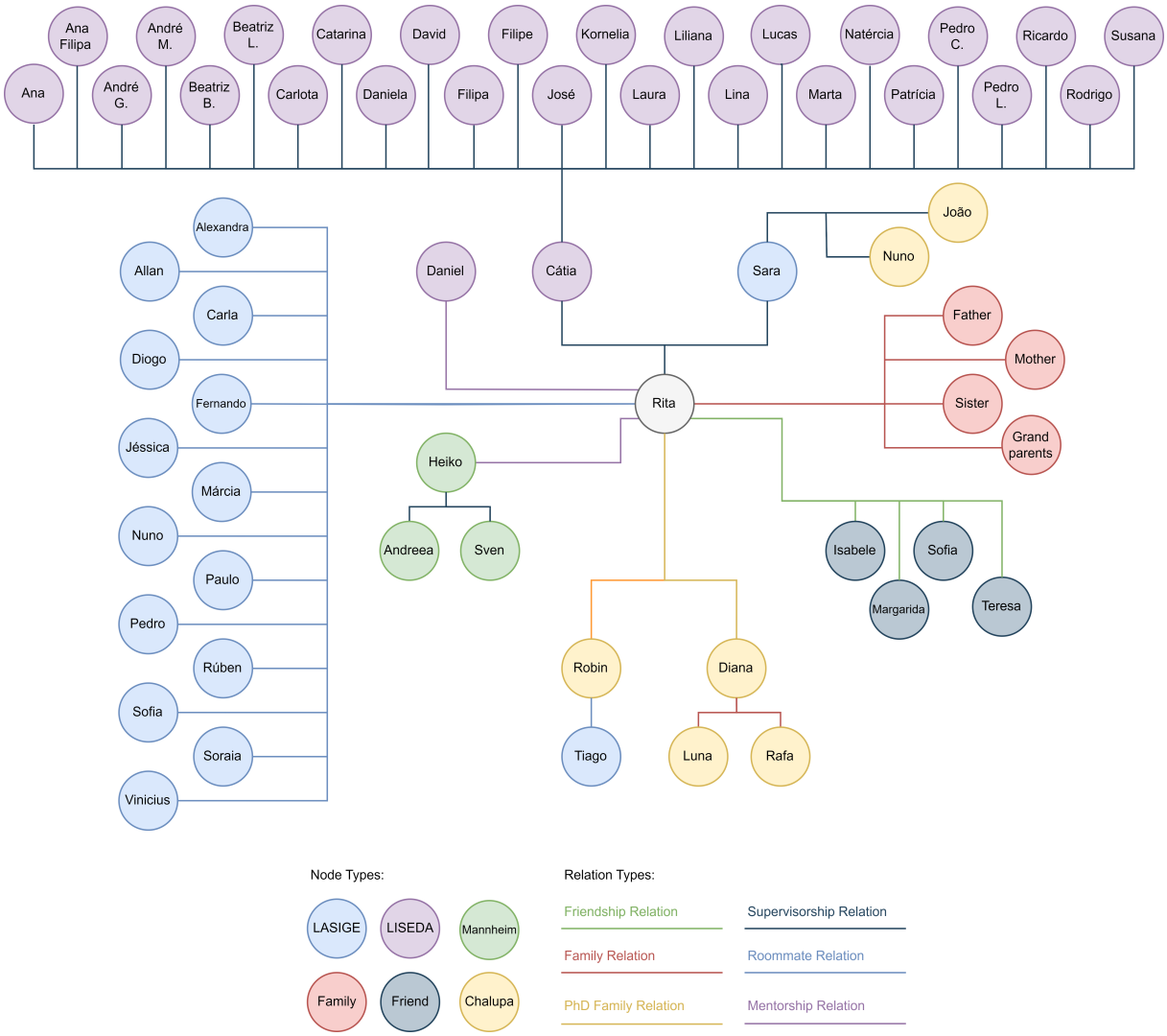


Figure 1: My support network. The idea for this picture emerged from a coffee break involving Diana, João, Robin, and Tiago.

In loving memory of my grandmother Alice.
”Na hora de pôr a mesa, seremos sempre cinco”
(Poem of José Luís Peixoto in ”A Criança em Ruínas”)

Abstract

Knowledge graphs represent an unparalleled opportunity for machine learning in the biomedical domain, given their ability to enrich data with meaningful context through semantic representations, such as knowledge graph embeddings and semantic similarity. However, the specificity of many biomedical tasks contrasts with the broad domains covered by large and successful biomedical knowledge graphs that describe entities according to several perspectives — semantic aspects. This is particularly challenging for predicting specific relations between entities described in the knowledge graph when the graph itself does not encode these relations.

Current semantic representation methods consider the knowledge graph as a whole, ignoring the different semantic aspects. This thesis hypothesizes that semantic representations that are able to distinguish semantic aspects can improve the performance and explainability of biomedical relation prediction tasks.

This work investigated different paradigms for defining semantic aspects based on classes and properties and developed multiple semantic representation techniques for both individual entities and entity pairs, with a focus on their explainability. Extensive experiments in protein-protein interaction and gene-disease association predictions supported the empirical evaluation of the proposed methods and demonstrated that semantic aspect-oriented representations improve both predictive performance and explainability, fostering biomedical research. This work further highlights that in complex and multi-disciplinary domains, where a single knowledge graph is used to support a wide variety of tasks, it is essential to shift from viewing knowledge graphs as a whole to focusing on specific semantic perspectives.

Keywords: Knowledge Graph; Semantic Similarity; Knowledge Graph Embedding; Machine Learning; Biomedical Application.

Resumo

Os grafos de conhecimento no domínio biomédico representam uma oportunidade única para a aprendizagem automática, dada a sua capacidade de fornecer um contexto significativo aos dados através de representações semânticas, tais como métodos de embedding e semelhança semântica. No entanto, a especificidade das tarefas de aprendizagem automática contrasta com os amplos domínios cobertos por muitos destes bem-sucedidos grafos de conhecimento que descrevem entidades de acordo com diversas perspetivas — aspetos semânticos. Isto é particularmente desafiador na previsão de relações específicas entre entidades descritas no grafo de conhecimento, quando estas relações não estão incluídas no grafo.

Os métodos atuais de representação semântica consideram o grafo de conhecimento como um todo, ignorando os diferentes aspetos semânticos. Esta tese de doutoramento levanta a hipótese de que as representações semânticas capazes de distinguir aspetos semânticos podem melhorar o desempenho e a explicabilidade de tarefas de previsão de relações biomédicas.

Este trabalho investigou diferentes paradigmas para definição de aspetos semânticos baseados em classes e propriedades e desenvolveu múltiplas técnicas de representação semântica tanto para entidades individuais quanto para pares de entidades, com foco na explicabilidade. Experiências extensivas na previsão de interações de proteínas e previsão de associação entre genes e doenças apoiaram a avaliação empírica dos métodos propostos e demonstraram que as representações orientadas a aspetos semânticos melhoram tanto o desempenho da previsão quanto a explicabilidade, promovendo o avanço da investigação biomédica. Este trabalho destaca ainda que em domínios complexos e multidisciplinares, onde um único grafo de conhecimento é usado para apoiar uma ampla variedade de tarefas, é essencial deixar de considerar os grafos de conhecimento como um todo para focar em perspetivas semânticas específicas.

Palavras Chave: Grafo de Conhecimento; Semelhança Semântica; Embedding de Grafos de Conhecimento; Aprendizagem Automática; Aplicação Biomédica.

Resumo Alargado

Os grafos de conhecimento são reconhecidos como um recurso valioso no suporte a tarefas de aprendizagem automática, pois associam significado e contexto a dados de forma estruturada. Um dos maiores desafios enfrentados pelas abordagens que combinam métodos de aprendizagem automática e grafos de conhecimento é como transformar os grafos numa representação que seja adequada aos algoritmos de aprendizagem automática. A maioria das abordagens existentes constrói uma representação semântica dos dados (ou seja, cada instância é representada por um vetor), o que permite a aplicação subsequente da maioria dos algoritmos de aprendizagem automática existentes. Os métodos de embedding tornaram-se cada vez mais populares para gerar representações semânticas, uma vez que mapeiam entidades dos grafos de conhecimento em vetores, preservando propriedades sintáticas e estruturais. No entanto, estas representações sacrificam a rica explicabilidade oferecida pelos grafos de conhecimento, nomeadamente os grafos de conhecimento ricos em ontologias. Como alternativa, a semelhança semântica também pode ser usada como uma representação semântica interpretável, uma vez que reflete a semelhança das entidades de acordo com o domínio representado pela ontologia.

As atuais representações semânticas baseadas em grafos de conhecimento consideram o grafo completo. Em dados complexos, como é o caso dos dados biomédicos, isso representa um desafio, uma vez que a informação necessária para suportar a aprendizagem pode ser difícil de explorar usando representações semânticas gerais que utilizam o grafo de conhecimento completo. Em primeiro lugar, algumas porções do grafo são desnecessárias para as representações e podem diminuir o desempenho da previsão devido ao ruído. Em segundo lugar, mesmo representações semânticas interpretáveis, como a semelhança semântica, quando reduzidas a um valor único num grafo de conhecimento de propósito geral, sacrificam a explicabilidade.

Dado que as entidades do grafo de conhecimento são descritas segundo vários pontos de vista no grafo de conhecimento, essas diferentes perspetivas representam uma oportunidade única para melhorar as representações semânticas. Neste trabalho, os aspetos semânticos são definidos como subgrafos que representam múltiplas perspetivas da representação de entidades no grafo de conhecimento. Dois tipos de aspetos semânticos são distinguidos: aspetos semânticos baseados em propriedades, se os subgrafos forem definidos por um conjunto de propriedades, e aspetos semânticos baseados em classes, se os subgrafos forem definidos por um conjunto de classes. Esta tese de doutoramento aborda o desafio de usar um grafo de conhecimento

de propósito geral, propondo diferentes abordagens que consideram os aspetos semânticos do grafo de conhecimento para aprender representações semânticas adequadas para suportar a aprendizagem automática. Esta tese foca-se em grafos de conhecimento ricos em ontologias que usam ontologias para descrever instâncias individuais, enquanto as próprias instâncias não têm ligações entre elas. Em relação ao problema, esta tese foca-se no desafio de aprender uma relação entre duas entidades do grafo, quando a própria relação não está explicitamente definida no grafo de conhecimento.

Este trabalho avança o estado da arte na exploração de grafos de conhecimento biomédicos apresentando metodologias para melhorar as representações semânticas para a previsão de relações, não apenas em termos de desempenho, mas também em termos de de explicabilidade. Três metodologias distintas (KGsim2vec, SEEK e TrueWalks) foram desenvolvidas. Cada abordagem é caracterizada pela sua definição de aspetos semânticos, as representações semânticas utilizadas (semelhança semântica ou métodos de embedding), as técnicas de aprendizagem automática utilizadas para aprendizagem supervisionada e as aplicações biomédicas nas quais foram avaliadas (previsão de interações entre proteínas e previsão de associações gene-doença).

O KGsim2vec adota representações semânticas baseadas em semelhança semântica. O KGsim2vec representa um avanço significativo em relação aos seus predecessores, nomeadamente evoKGsim+ e a ferramenta de semelhança semântica supervisionada. Essas metodologias anteriores introduziram a incorporação de aspetos semânticos, definidos como subgrafos a uma distância de um do nó raiz, com subsequente geração de representações semânticas baseadas em semelhança semântica para cada aspeto. No entanto, o KGsim2vec oferece flexibilidade na seleção de classes que servem como raízes do subgrafo. Aumentando a distância até a raiz do grafo de conhecimento, a metodologia explora subgrafos enraizados em classes mais específicas, gerando representações mais interpretáveis. A avaliação foi focada na previsão de interações proteína-proteína e como as representações baseadas em semelhança semântica podem ser exploradas por métodos de aprendizagem automática com diferentes níveis de explicabilidade. As experiências revelaram que modelos de aprendizagem automática interpretáveis, juntamente com características combinadas com representações geradas pelo KGsim2vec, têm um desempenho melhor do que métodos opacos baseados em métodos de embedding ou redes neuronais de grafos. Para além disso, o KGsim2vec é capaz de produzir modelos globais que capturam fenómenos biológicos e elucidam vieses nos dados.

O SEEK diverge do KGsim2vec tanto no tipo de representação como na definição de aspetos semânticos. O SEEK representa pares de entidades através de subgrafos enraizados nos ancestrais em comum que são disjuntos, usando métodos de embedding. Uma vez que o SEEK identifica subgrafos partilhados relevantes entre entidades, produz uma representação multifacetada e explicável para previsão de relações entre entidades. Os resultados mostraram que o SEEK é capaz simultaneamente de melhorar em relação a outros métodos de embedding e de gerar explicações que podem identificar novas interações. A avaliação experimental de novas interações identificadas pelo SEEK corrobora o potencial das abordagens explicáveis em abordagens de inteligência artificial.

Enquanto o KGsim2vec e o SEEK utilizam aspetos semânticos baseados em classes, a inovação no TrueWalks reside na utilização de aspetos semânticos baseados em propriedades. No TrueWalks, os tipos de propriedades incluem declarações positivas e negativas que descrevem entidades. Consequentemente, para cada entidade, é gerada uma representação baseada em métodos de embedding para cada tipo de propriedade (uma representação gerada sobre o subgrafo que contem declarações positivas e outra representação semântica gerada sobre o subgrafo que contém declarações negativas). O TrueWalks também é avaliado para prever interações proteína-proteína e associações gene-doença. Os resultados mostraram que o TrueWalks supera os métodos de embedding popularmente usados em ambas as tarefas. Além disso, estes resultados abrem caminho para demonstrar que o conhecimento negativo e os resultados negativos são importantes e não devem ser descartados.

Em resumo, esta tese demonstra que a geração de representações semânticas que têm em consideração os diferentes aspetos semânticos do grafo de conhecimento pode alcançar previsões mais precisas, mas também gerar representações potencialmente explicáveis que ajudarão o avanço da investigação biomédica. Três perguntas de investigação foram respondidas com sucesso. A primeira questão de investigação está relacionada com as representações semânticas mais adequadas para serem usadas por métodos de aprendizagem automática. Dois tipos de representações semânticas são usados e comparados ao longo das diferentes metodologias: semelhança semântica e métodos de embedding. A segunda questão de investigação diz respeito à forma os aspetos semânticos podem ser explorados pelos algoritmos de aprendizagem automática para melhorar as representações semânticas. As metodologias propostas exploram definições distintas de aspetos semânticos. Finalmente, a terceira questão de investigação visa entender se as representações semânticas geradas pelas metodologias propostas são úteis para aplicações no domínio biomédico. Para todas as metodologias, a avaliação é focada em tarefas biomédicas que suporta a avaliação comparativa de representações que consideram todo o grafo de conhecimento e representações semânticas que consideram os aspetos semânticos. Além disso, vários conjuntos de dados de referência for gerados para avaliar a eficácia das diferentes abordagens em tarefas diversas: previsão de interação entre proteínas e de associações gene-doença.

No entanto, as contribuições desta tese não são apenas as três metodologias, mas também uma nova visão para as abordagens de aprendizagem automática que utilizam grafos de conhecimento. Do meu ponto de vista, é necessário deixar de ver os grafos de conhecimento como um todo que é considerado igualmente. Para beneficiar do total potencial dos grafos de conhecimento para diversas tarefas, incluindo no domínio biomédico, as representações devem capturar os diferentes aspetos semânticos do grafo de conhecimento.

Palavras Chave: Grafo de Conhecimento; Semelhança Semântica; Embedding de Grafos de Conhecimento; Aprendizagem Automática; Aplicação Biomédica.

Contents

List of Figures

List of Tables

Glossary

AI Artificial Intelligence. 33–35, 37, 38, 67, 68, 91, 141

BR Bayesian Ridge. 28

DT Decision Tree. 29, 30, 34, 36, 41–44, 47, 51, 56, 62, 73, 74, 80, 87, 136

GCNN Graph Convolutional Neural Network. 32, 43, 44, 51, 52, 59, 60

GDA Gene-Disease Association. 2, 5–8, 42–44, 51, 53, 55, 58–60, 68, 91, 92, 99–105, 108, 112, 113, 119, 122–124, 126, 127, 133, 134, 141, 143, XX

GNN Graph Neural Network. 1, 2, 4, 28, 32, 34, 35, 37, 42–45, 50, 58, 60, 61, 64, 69, 79, 80, 132, 141–143, XIX

GO Gene Ontology. 1, 2, 14, 15, 27, 46, 47, 53, 54, 56–60, 62, 68–71, 75, 76, 79, 82, 83, 85, 90, 94, 97, 100, 105, 114, 115, 117, 119–122, 138–141, XVII, XVIII, XX

GP Genetic Programming. 29, 30, 34, 41, 42, 73, 74, 78, 80, 136, 138, 140

HP Human Phenotype Ontology. 2, 14, 15, 53, 54, 57, 60, 100, 115, 119–122, 139, 141, XVII, XX

IC Information Content. 23–26, 56, 73, 74, 90, 136

KG Knowledge Graph. 1–8, 11–18, 21–23, 26–28, 32, 34, 36–39, 41–47, 49–62, 64, 68–72, 75, 76, 78–80, 90–94, 96–102, 108, 111–122, 124–127, 131–134, 137–143, XVII–XIX

LIME Local Interpretable Model-Agnostic Explanations. 74, 80, 81, 83–87, 134

LORE Local Rule-Based Explanations. 74, 80, 81, 83–86, 134

LR Linear Regression. 28, 34, 36, 41–44, 49, 52, 54, 56, 57, 140

ML Machine Learning. 1, 2, 4–8, 11, 16, 24, 28, 30, 32–37, 39, 41, 45, 49, 52–54, 56–58, 62, 64, 67–71, 73, 74, 76, 79–86, 90–93, 97, 101, 102, 107, 113, 118, 123, 127, 131–135, 137, 139–141, XVII, XIX, XX

MLP Multilayer Perception. 31, 32, 41, 101, 102, 109

NN Neural Network. 20, 21, 31, 32, 36, 41–46, 49–56, 58–60, 62, 101

OWL Web Ontology Language. 14, 96, 114–117, 119, 120, XVIII

PPI Protein-Protein Interaction. 1, 2, 5–8, 41, 42, 45–47, 51, 55, 57, 67–69, 71, 75–78, 81, 87, 90–92, 94, 99–106, 108, 109, 112, 113, 119, 121–127, 133–135, 138, 140, 142, 143, XX

RDF Resource Description Framework. 12, 21, 71, 94–96, 115–117

RF Random Forest. 30, 41–44, 49, 53, 54, 56, 57, 61, 73, 80, 81, 101, 102, 119, 124, 134, 135, XX

RQ Research Question. 5, 6, 69, 92, 112, 131, 132, 134, 137

SVM Support Vector Machine. 41–44, 49, 54, 56, 57

XAI Explainable Artificial Intelligence. 28, 34–38, 98, 134

XGB eXtreme Gradient Boosting. 30, 31, 41–44, 51, 53, 54, 56, 57, 73, 80, 81, 101, 102

Chapter 1

Introduction

The explosion in complexity and heterogeneity of data has motivated a new paradigm, where millions of semantically-described entities are represented in Knowledge Graphs (KGs) [Hogan et al., 2021]. KGs [Ehrlinger and Wöß, 2016] represent factual information about entities in the real world and how they relate to each other. Particularly in the biomedical domain, KGs have become highly relevant because they are typically built by integrating ontologies [Staab and Studer, 2010], allowing the description of complex natural phenomena that are not easily captured in mathematical form [Nicholson and Greene, 2020]. Large and successful biomedical KGs encode semantics that describe entities in terms of several perspectives, in this work, defined as semantic aspects. Given these biomedical KGs richness, they can be exploited in a wide variety of data mining and Machine Learning (ML) tasks, from explaining data to making predictions. The problem is that the specificity of the tasks contrasts with the broad domains covered by many biomedical ontologies. For instance, the Gene Ontology (GO) is a very successful biomedical ontology that describes protein function according to three domains: the molecular functions they perform, the biological processes they intervene in and the cellular components where they are active. GO and its associated annotations that link proteins to GO classes make up a KG. The GO KG has been used for multiple biomedical tasks, namely Protein-Protein Interaction (PPI) prediction. However, it is well established that the prediction is more accurate if only a portion of the KG is used (in this case, the one concerning biological process) rather than the whole KG [Bandyopadhyay and Mallick, 2017; Sousa et al., 2020]. This shows that depending on the analytical task for which the KG will be used, semantic aspects should be considered differently rather than relying on the whole KG.

Currently, two predominant paradigms exist for mining of KGs [Hamilton, 2020]. The traditional paradigm involves transforming data coming from the whole KGs into a numerical semantic representation that classical ML algorithms can process. In contrast, a more contemporary paradigm has gained substantial traction recently. It is based on defining Graph Neural

Network (GNN) architectures explicitly designed for graph structures rather than the conventional process of generating numerical representations as a bridge. However, this design does not align with the inherently heterogeneous structure of KGs [Hogan et al., 2021], particularly those enriched with ontological information. Furthermore, GNN relies on the presence of node features, which hinders their applicability to real-world problems where such features may be unavailable [Cui et al., 2022]. This constraint is particularly evident in ontology-rich KGs that typically provide only labels instead of numerical node properties that can be explored by the message-passing mechanisms of GNNs.

This brings up the challenge of generating a semantic representation of the KG entities that consider the different semantic aspects. While in node, type or link prediction, entity representations may be tailored to a particular task [Wang et al., 2017], in the scenario where an entity pair representation is needed, but the relationship between those entities is not a part of the KG, no such tailoring is possible. This last scenario has various bioinformatics and health informatics applications, such as the prediction of PPIs exploring the GO [Zhong et al., 2019], drug-drug interactions exploring Bio2RDF [Celebi et al., 2019], or the mining of Gene-Disease Associations (GDAs) based on the Human Phenotype Ontology (HP) [Asif et al., 2018]. Furthermore, interpretability and explainability have become a concern in ML. The effectiveness and usefulness of a semantic representation approach depend on the assumption that a semantic representation serves as a semantically meaningful representation of the entities they represent. To verify this assumption, these representations must be explainable, meaning they can provide a description understandable to humans regarding the logic, behavior, or factors that influence the process of learning the representations. This requirement is fundamental to ensure the scientific validity of KG semantic representation as a powerful tool that can be used to uncover new knowledge, help understand the mechanisms underlying natural phenomena, and distinguish meaningful predictions from spurious correlations [Barredo Arrieta et al., 2020]. However, when generating representations encompassing the entire graph, the potential for explainability promised by KGs may be compromised.

All these challenges create the need for approaches that improve semantic representations to support supervised tasks, not only in terms of the accuracy of the representations but also in terms of explainability.

1.1 Problem Formulation

Biomedical KGs are a recognized valuable source for background information in many ML tasks, encoding semantics that describe biomedical entities [Nicholson and Greene, 2020; Alshahrani et al., 2021; Mohamed et al., 2021]. Within the field of biomedical research, KGs have been used to prioritize genes relevant to diseases [Mukherjee et al., 2021; Binder et al., 2022], identify protein-protein and drug-drug interactions [Karim et al., 2019; Chen et al., 2019], perform drug

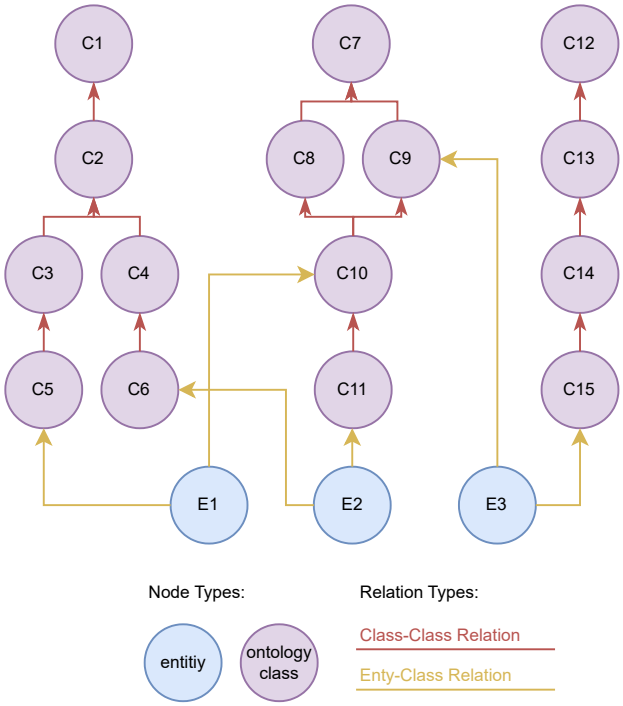


Figure 1.1: Example of an ontology-rich KG.

repurpositing [Sosa et al., 2019; Saadat et al., 2022], among others. In the vast majority of biomedical KGs, larger biomedical ontologies are used to describe entities through ontology annotation. Within these KGs, there are two types of vertices (those that correspond to biomedical entities and those that correspond to ontology classes), as well as two types of edges (those that relate ontology classes to each other and those that link individuals to the classes that describe them). Contrary to most KGs, there are no links between the individuals representing biomedical entities in ontology-rich KGs. Figure 1.1 shows an example of an ontology-rich KG, where entities represented in blue are described using ontology classes defined in purple. There are two types of relations, while the individuals themselves are usually flat with no connections between them. Although there is still no consensual definition of what a KG is, this thesis focuses on ontology-rich KGs and adopts the Definition 1.

Definition 1. An Ontology-rich KG is a labeled directed graph $KG = (V_c, V_i, E)$ where V is the set of vertices that represent either ontology classes V_c or individuals (real-world entities) V_i, and E is the set of edges that are established between vertices, representing either ontology-level axioms, such as subclass statements or property restrictions and the assignment of an individual to a class through type declarations.

Since classical ML algorithms accept typical vector-based representations as input, many approaches that explore KG generate numerical semantic representations (Definition 2). The state-of-the-art KG-based semantic representations are based on graph embeddings [Wang et al., 2017], which produce feature vector (propositional) representations of the entities in the KG. Semantic similarity [Harispe et al., 2015] can also be used as a semantic representation by comparing entities based on their properties and taxonomic relationships and computing similarity values. These approaches that generate semantic representations diverge from end-to-end approaches that use GNN architectures to process directly the KG [Zhou et al., 2020].

Definition 2. A semantic representation is a set of features describing a KG entity and obtained by processing the KG.

When a biomedical KG is explored in the context of an independent and specific learning task, it may very well be the case that large portions of the KG are irrelevant for the task. This is particularly challenging when predicting specific relations between entities that correspond to KG individuals but whose relation is not encoded and inferred in the ontology-rich KG. This thesis targets relation prediction tasks (Definition 3). This is a fundamentally distinct task from link prediction. The goal of link prediction is to find new links between entities in the KG, given the existing links among the entities [Kazemi and Poole, 2018]. In opposition, in relation prediction, the existing links between entities are not part of the KG.

Definition 3. Relation prediction is the task of learning a relation between two KG entities, a pair, when the relation itself is not explicitly defined in the KG.

Although most of the approaches rely on creating a representation of each entity using the whole KG, KG entities are described according to different perspectives or semantic aspects (Definition 4). This work distinguishes two types of semantic aspects: property-based semantic aspects (Definition 5) and class-based semantic aspects (Definition 6). Exploring semantic aspects presents a unique opportunity to improve semantic representations.

Definition 4. A semantic aspect is a KG subgraph that represents a perspective of the representation of KG entities.

Definition 5. A property-based semantic aspect is a subgraph extracted from the full KG, $KG_{SA} = (V'_c, V'_i, E')$, where each vertex $v'_i \in V'_i$ is an individual of a class in V'_c, and where each $e' \in E'$ corresponds to edge of type t between elements of $V'_c \cup V'_i$.

Definition 6. A class-based semantic aspect is a subgraph extracted from the full KG, $KG_{SA} = (V'_c, V'_i, E')$ rooted in class a, where each vertex $v'_c \in V'_c$ is a subclass (directly or through inference) of a, each vertex $v'_i \in V'_i$ is an individual of a class in V'_c, and where each $e' \in E'$ corresponds to an edge between elements of $V'_c \cup V'_i$.

This thesis aims to develop methodologies to improve semantic representations of data objects extracted from KGs to support supervised relation prediction tasks by taking into account the different semantic aspects. The methodologies are specifically tailored to handle the characteristics of ontology-rich KGs. The evaluation focuses on relevant relation prediction tasks, particularly in predicting PPIs and GDAs.

1.2 Research Objective and Research Questions

A severe limitation of several approaches for ML using biomedical KGs is that the construction of semantic representations often employs the full KG, blind to the fact that some semantic aspects may be irrelevant to the downstream ML task, potentially introducing noise. Another limitation is the lack of explainability of representations generated using the whole KG. This thesis hypothesises that considering the different KG semantic aspects can improve semantic representations to support biomedical relation prediction tasks, not only in terms of performance but also in terms of interpretability and explainability. This thesis addresses three Research Questions (RQs):

- **RQ1: Which are the semantic representations that are more suitable to support supervised learning over KGs?** This work considers two main types of semantic representations: KG embedding methods and semantic similarity [Kulmanov et al., 2021]. KG embeddings map each node of a KG to a lower-dimensional space in which its graph position and the structure of its local graph neighborhood are preserved as much as possible. Semantic similarity takes two entities as input and returns a numeric score that quantifies how similar the two entities are according to their description in the KG. Several semantic similarity measures have been used as semantic representations, with most measures falling in the category of taxonomic semantic similarity. Taxonomic semantic similarity measures extensively use the taxonomical aspect of an ontology, comparing classes based on subclass/superclass relations. One of the research topics in this work is the comparison of the different semantic representation techniques introduced in Chapter 2. The goal is to investigate the strengths and weaknesses of the individual semantic representation techniques or types of semantic representations for specific learning tasks. Chapter 4 focus

on semantic similarity-based representations that are inherently explainable. In Chapters 5 and 6, the emphasis is on KG embeddings that can capture the properties of the graph without reducing it to a single point. However, contrary to semantic similarity, the embeddings are not explainable by design.

- **RQ2: How can semantic aspects and ML be explored to improve semantic representations?** In most KG-based approaches, the construction of semantic representations uses the full KG, ignoring the learning task and the viewpoint of the domain. The purpose is to take a step ahead and develop ML-based methodologies to generate semantic representations that explore the different semantic aspects of the KG. An essential component of these methodologies is the definition of the semantic aspects. In this work, three different definitions of semantic aspects are explored in Chapters 4, 5, and 6. The methodologies are evaluated not only in terms of predictive performance but also explainability.

- **RQ3: Are the improved semantic representations useful to bioinformatics applications?** Since complex biomedical KGs have been widely used to represent biomedical entities and predicting relations between entities is a fundamental task in the biomedical domain, several tasks can benefit from the improved semantic representations. In this work, the evaluation is focused on two biomedical relation prediction tasks: PPI prediction and GDA prediction. PPIs are responsible for many critical functions in biology and are highly relevant to disease states. Identifying GDAs is also critical since it can contribute to improving medical care and understanding disease mechanisms. However, discovering new interactions or associations through laboratory experiments is expensive and time-consuming, leading to the need for computational approaches to predict candidate pairs to support a more targeted experimental analysis [Jimenez-Sanchez et al., 2001]. In addition, these tasks are backed by large ontologies with multiple semantic aspects and gold-standard datasets created based on experimental evidence. Furthermore, both tasks are interesting and representative but quite distinct in terms of semantics. Contrary to what happens for GDA prediction, the relationships between the different semantic aspects and PPI interaction are well established. Moreover, in the case of GDA, the goal is predicting a relation between two distinct entity types.

1.3 Contributions

This work advances the state of the art in biomedical KGs mining by presenting methodologies for improving semantic representations for relation prediction, not only in terms of performance, but also in terms of interpretability and explainability. The proposed approaches are summarized in Table 1.1. The code and data for all the proposed methodologies are publicly available.

- **Generation of similarity-based semantic representations for relation prediction.** KGsim2vec generate explainable vector representations using aspect-oriented semantic similarity features to represent pairs of entities in a KG. The quality of explanations is evaluated by considering both their size and informativeness. The results show that KGsim2vec improves explainability and performs better than opaque methods based on PPI embeddings for PPI prediction. Furthermore, KGsim2vec produces explanations that capture biological phenomena and elucidates data biases [Sousa et al., 2022].
- **Generation of explainable embedding-based semantic representations for relation prediction.** SEEK [Sousa et al., 2023c] is a novel approach for explainable representations to support relation prediction in KGs. It is based on identifying relevant shared semantic aspects between entities and learning representations for each subgraph, producing a multi-faceted and explainable representation. Extensive analysis using established benchmarks PPI and GDA prediction demonstrates that SEEK achieves significantly better performance than standard learning representation methods while identifying both sufficient and necessary explanations based on shared semantic aspects.
- **Generation of embedding-based semantic representations with negative statements for relation prediction.** TrueWalks [Sousa et al., 2023b] is a novel approach to incorporate negative statements into the KG representation learning process. In particular, TrueWalks includes a novel walk-generation method that differentiates between positive and negative statements but also takes into account the semantic implications of negation in ontology-rich KGs. This is particularly important for applications in the biomedical domain, where the inadequacy of embedding approaches regarding negative statements at the ontology level has been identified as a crucial limitation. TrueWalks is evaluated in ontology-rich biomedical KGs in PPI prediction and GDA prediction, using the benchmark dataset [Sousa et al., 2023a] generated to circumvent the difficulties in building benchmarks for KGs with negative statements.

1.4 Document Structure

Besides this introductory chapter, the thesis is organized into three main parts:

- **Foundations** provides the necessary background knowledge to understand the research work. Chapter 2 focuses on introducing the key concepts of two fundamental topics: KGs and ML. Understanding these concepts is fundamental for comprehending the core aspects of this research work. Chapter 3 gives a literature review of the ML approaches over KGs for biomedical applications. The analysis of the related work enables positioning this work within the current approaches.

Table 1.1: Summary of the proposed approaches according to the semantic representation generated, the definition of semantic aspect, the interpretability of the ML employed, the explainability of the features, and the application tasks. The ML methods are categorized into opaque (■) and transparent (□). The explainability of features is divided into two categories: non-explainable (✗) or explainable (✓).

Approach	Semantic Representation	Semantic Aspect	Interpretability of the ML	Explainability of the features	Task
KGsim2vec	Taxonomic semantic similarity	Class-based semantic aspects (subgraphs rooted in the classes at x distance from the root class)	□ ■	✓	PPI prediction
SEEK	KG embeddings	Class-based semantic aspects (subgraphs rooted in the shared disjoint common ancestors)	■	✓	PPI and GDA prediction
TrueWalks	KG embeddings	Property-based semantic aspects (subgraphs defined by the positive and negative statements)	■	✗	PPI and GDA prediction

- **Methodologies** describes the approaches proposed for improving semantic representations for ML tasks. Chapter 4 addresses the explainability challenge by proposing KGsim2vec that explores the different semantic aspects of a KG to generate explainable similarity-based semantic representations for relation prediction. It also describes an investigation of the possible data leakage in PPI approaches. Chapter 5 presents SEEK, an approach to generate explainable embedding-based semantic representations for entity pairs using the shared semantic similarity. Chapter 6 focuses on exploring the semantic implications of negative statements to generate embedding-based semantic representations for relation prediction, the TrueWalks embeddings.
- **Conclusions** presents a comprehensive overview of this research work and a summary of the main contributions. It also provides an outlook on potential future research directions and areas that could be explored to expand upon the contributions of this thesis.

The **Appendices** contain reproductions of published and submitted papers.

Part I

Foundations

Chapter 2

Fundamental Concepts

The research associated with this Ph.D. thesis builds on the state of the art from two domains: KGs and and ML algorithms. This chapter provides an overview of the key concepts for each domain needed to understand the foundations of this Ph.D. thesis, while also offering a brief overview of ML tasks over KGs.

2.1 Knowledge Graphs

The term Semantic Web was introduced in 2001 by Tim Berners-Lee to mean "an extension of the current web in which information is given well-defined meaning, better-enabling computers and people to work in cooperation" [Berners-Lee et al., 2001]. Supporting this evolution, a set of best practices for publishing and connecting structured data on the Web was established, known as Linked Data [Bizer et al., 2011]. Linked Data principles aim to ensure all published data is machine-readable and becomes part of a single global data space. The Linked Open Data cloud diagram [1] (Figure 2.1) shows the vast number of datasets that have been published in the Linked Data format across different domains.

Since the beginning, the Semantic Web has promoted a graph-based representation of knowledge. KGs contain factual knowledge about real-world entities and the relations between them in a fully machine-readable format. A large number of KGs, both free and commercial, have been created, including YAGO [Suchanek et al., 2007], DBpedia [Auer et al., 2007], NELL [Carlson et al., 2010], Freebase [Bollacker et al., 2008]. However, only after the introduction of the Google KG[2] as a backbone of a new Web search strategy in 2012, the term KG began to be

[1] http://cas.lod-cloud.net/

[2] https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html

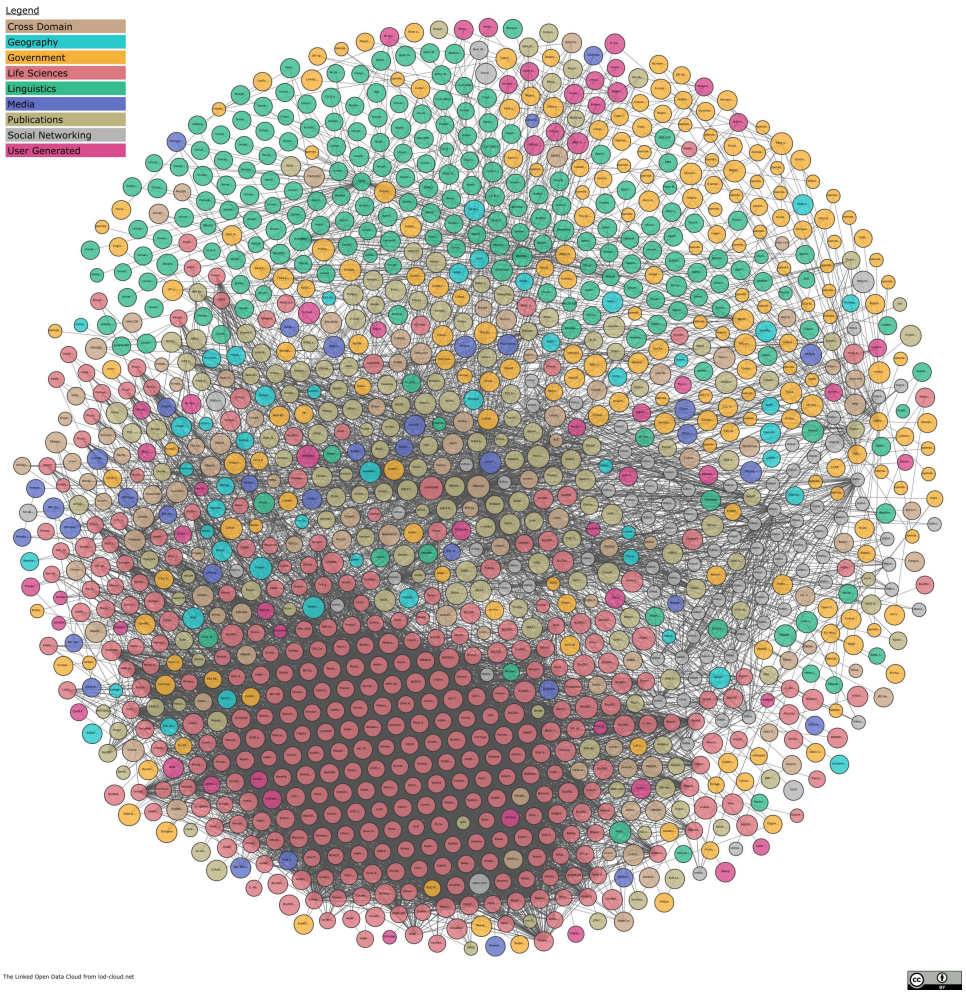


Figure 2.1: The Linked Open Data Cloud retrieved from `lod-cloud.net` on August 2023. Each node represents a dataset that has been published in the Linked Data format.

widely used by the semantic web community. Considering the considerable research focus on KGs, a diverse range of published definitions have been proposed since 2012 [Hogan et al., 2021; Ehrlinger and Wöß, 2016]. The abundance of definitions prompted researchers to gather several definitions of KG and proposed their own definition. These works include Ehrlinger and Wöß [2016] and Hogan et al. [2021]. Table 2.1 present some of those definitions.

In this work, a KG is defined as a labeled directed graph $KG = (V, E, R)$ where V is the set of vertices that represent entities, R is the set of relations and E is the set of edges that connect vertices through relations. The majority of KGs are represented in Resource Description Framework (RDF), a standard data model for Linked Data. In RDF terminology, a statement is a small piece of knowledge in the format of subject-predicate-object expressions, where the

Table 2.1: Definitions of KG.

Definition	Reference
A KG "(i) mainly describes real world entities and their interrelations, organized in a graph, (ii) defines possible classes and relations of entities in a schema, (iii) allows for potentially interrelating arbitrary entities with each other and (iv) covers various topical domains."	Paulheim [2017]
A KG is a "large network of entities, their semantic types, properties, and relationships between entities."	Krötzsch and Weikum [2016]
A KG is "an RDF graph. An RDF graph consists of a set of RDF triples where each RDF triple (s, p, o) is an ordered set of the following RDF terms: a subject $s \in U \cup B$, a predicate $p \in U$, and an object $U \cup B \cup L$. An RDF term is either a URI $u \in U$, a blank node $b \in B$, or a literal $l \in L$."	Färber et al. [2018]
A KG "acquires and integrates information into an ontology and applies a reasoner to derive new knowledge."	Ehrlinger and Wöß [2016]
A KG is "a graph of data intended to accumulate and convey knowledge of the real world, whose nodes represent entities of interest and whose edges represent potentially different relations between these entities. The graph of data (a.k.a. data graph) conforms to a graph-based data model, which may be a directed edge labelled graph, a heterogeneous graph, a property graph, and so on."	Hogan et al. [2021]

subject and the object are two things, and the predicate is the relation that connects these two things.

Ontologies are typically used to describe KG entities. An ontology defines the universe of classes and relationships relevant to a specific domain [Staab and Studer, 2010]. Ontologies consist of a set of classes organized in a hierarchy and a set of semantic links that elucidate the connections between classes. Ontologies can be divided into two main components: the TBox, which defines the hierarchical structure and relationships between classes and the axioms that describe domain knowledge; the ABox, which includes statements asserting the existence of individuals and their properties, populating the ontology with concrete data [Baader et al., 2004].

2.1.1 Biomedical Knowledge Graphs

One of the scientific areas where ontologies and KGs have been more successful is in the biomedical domain. Not only is the number of biomedical ontologies increasing, but their size is also growing, their relevance in research is rising, and they penetrate more areas of biology and biomedicine [Hoehndorf et al., 2013]. Biomedical ontologies are used in areas ranging from gene function [Consortium, 2021] to those used to characterize drugs [Degtyarenko et al., 2007]. Phenotype ontologies [Köhler et al., 2020] are also available for multiple species. Open repositories such as the BioPortal [Whetzel et al., 2011] prove the vast number of ontologies available in various formats. Two very relevant KGs that offer a comprehensive description of biomedical entities are GO KG and HP KG.

The GO KG is used to describe proteins and is built by integrating the GO [Consortium, 2021] and protein annotation data [Huntley et al., 2015]. The GO defines a hierarchy of classes that describe protein functions and their relationships. It can be represented as a graph where nodes are GO classes and edges define relationships between them (e.g., *is_a*; *part_of*; *has_part*; *regulates*; *negatively_regulates* and *positively_regulates*), being the majority *is_a* relations. Functions in GO are described concerning three domains: the biological processes a gene product is involved in, the molecular functions a gene product executes, and the cellular components where a gene product is. The three domains of GO (biological processes, molecular functions, and cellular components) are represented as separate root ontology classes since they do not share any common ancestor. A GO annotation is a statement about the function F of a protein P, and it is added in the KG as an assertion $P, hasFunction, F$. In GO KG, nodes represent proteins or GO classes, and edges represent links between GO classes or annotations (Figure 2.2).

The HP KG comprises the HP [Köhler et al., 2020] and HP annotation data to describe genes and diseases. HP characterizes the phenotypic abnormalities in human hereditary diseases concerning five aspects, namely phenotypic abnormalities, mode of inheritance, clinical course, clinical modifier and frequency. Regarding the HP annotations, they link genes and diseases to HP classes. In the HP KG, the nodes are HP classes or genes. The edges represent ontology relations or links between genes and HP classes via their annotations (Figure 2.3).

Ontology-rich KGs in the biomedical domain are typically defined using Web Ontology Language (OWL) [Grau et al., 2008] since biomedical ontologies are typically developed in OWL or have an OWL version. Many of these biomedical ontologies fall in the OWL 2 $\mathcal{EL}$ profile [Kulmanov et al., 2019]. For example, using OWL 2 whose constructs correspond to SROIQ(D), it is possible to indicate that a protein P carries out a function F described in the GO by declaring the axiom $P \sqsubseteq \exists hasFunction.F$.

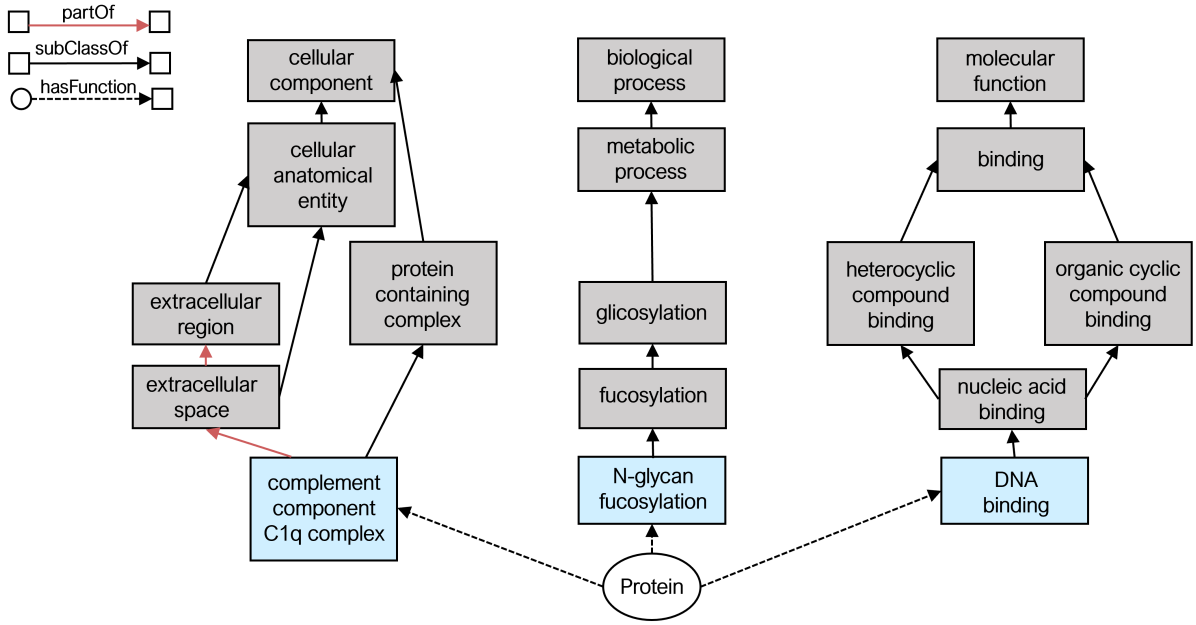


Figure 2.2: Example of a protein represented under three domains of the GO. For simplicity, only a small portion of GO KG is shown.

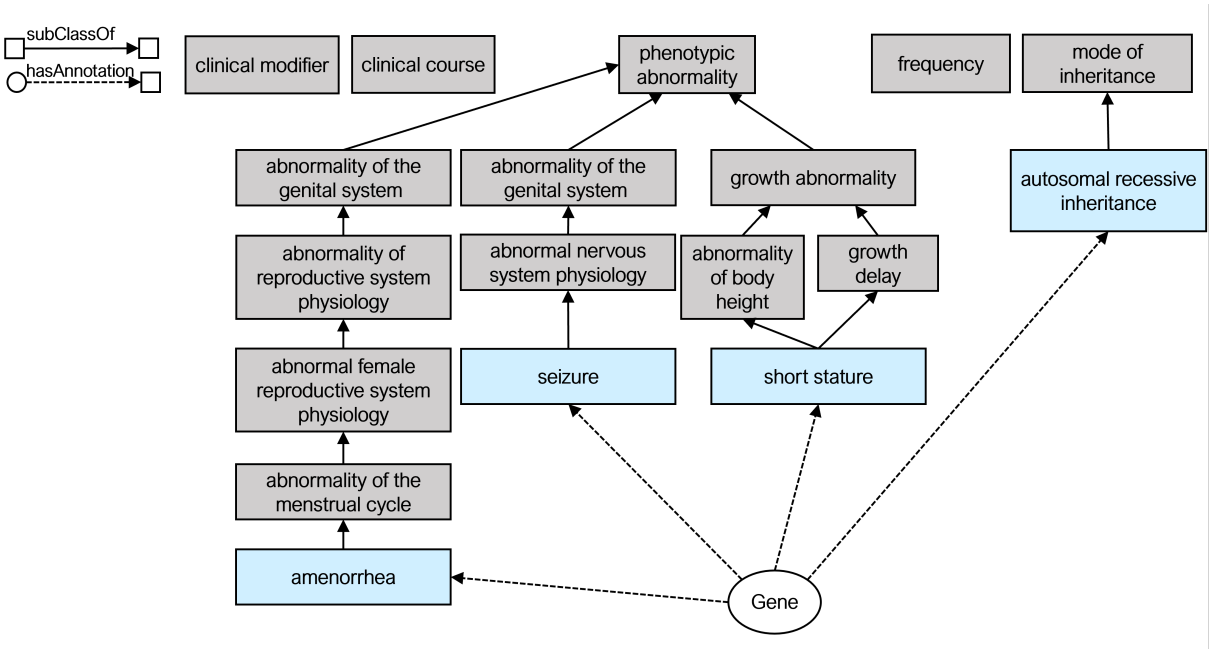


Figure 2.3: Example of a gene represented under HP. For simplicity, only a small portion of HP KG is shown.

2.1.2 Knowledge Graph Semantic Representations

As defined in section 1.1, a KG-based semantic representation is a set of features describing a KG entity and obtained by processing the KG. KG-based semantic representations bridge the gap between KGs and the typical vector-based representations of entities used by most ML techniques. Once a suitable representation is achieved, different ML algorithms can be employed for a wide variety of downstream learning tasks outside the KG, such as clustering and entity classification. However, semantic representations can have a role beyond serving as features for ML outside the KG, as demonstrated by their direct application in other contexts. A notable example are link prediction approaches that use a within-KG protocol, where the triples in the knowledge graph are divided into a training, testing, and validation set [Portisch et al., 2022].

Most state-of-the-art KG-based representations fall into the KG embedding methods. A less well-known alternative is to employ semantic similarity as a representation by comparing entities based on their properties and taxonomic relationships and computing similarity values for pairs of entities. Graph kernels methods, designed to learn representations through pairwise data point comparisons using kernels, determine the distance between two instances by assessing shared substructures like walks, paths, and trees within the KG [Ramon and Gärtner, 2003; Borgwardt and Kriegel, 2005; Shervashidze et al., 2009, 2011]. Despite their initial success, these methods lost popularity in recent years due to their inability to efficiently handle the growing size of KGs. Therefore, in the context of this Ph.D., two types of KG-based representations are considered: KG embeddings and semantic similarity.

A crucial difference exists between KG embeddings and semantic similarity regarding the underlying representation. Embedding methods are designed to generate representations of individual entities, making them particularly suitable for tasks like node classification. Conversely, semantic similarity measures are explicitly designed to create representations for pairs of entities, aligning well with relation prediction, the target task of this Ph.D. However, either embedding methods can be used to generate a pair representation by combining the individual representations of each entity within the pair, or semantic similarity can be used to derive a representation for a single entity by computing its similarity in relation to all other entities existing in the KG.

2.1.2.1 Knowledge Graph Embeddings

An embedding is a vector representation that maps each node to a lower-dimensional space where the underlying structure of the KG and other semantic information are preserved as much as possible [Cai et al., 2018]. These embeddings have been employed as features in a variety of downstream tasks, with particular success in the life sciences [Mohamed et al., 2021; Kulmanov et al., 2021; Chen et al., 2019; Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan, 2022].

There are various methods that can be used to generate embeddings [Roweis and Saul, 2000; Balasubramanian and Schwartz, 2002; Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015;

Ji et al., 2015, 2016; He et al., 2015; Nickel et al., 2011; Yang et al., 2015; Nickel et al., 2016; Trouillon et al., 2016; Wang et al., 2016; Cao et al., 2016; Perozzi et al., 2014; Grover and Leskovec, 2016; Ristoski and Paulheim, 2016a; Dong et al., 2017; Portisch and Paulheim, 2021; Smaili et al., 2018a,b; Chen et al., 2021a; Kulmanov et al., 2019; Xiong et al., 2022]. In this thesis, six categories of embedding methods are considered: matrix-factorization, translational distance, semantic matching, deep learning-based, random walks-based, and ontology-aware. There are some recent surveys of KG embeddings [Wang et al., 2017; Gesese et al., 2021], but they do not cover the recent advances in embeddings approaches that aim to tailor representations by considering different semantic, structural or lexical aspects of a KG and its underlying ontology, such as EL [Kulmanov et al., 2019] and BoxEL [Xiong et al., 2022]. Moreover, embeddings based on random walks are rarely included in these surveys. The random walk-based methods transform the KG into node sequences, and then natural language methods are applied to the sampled walks [Ristoski and Paulheim, 2016a; Perozzi et al., 2014; Grover and Leskovec, 2016]. While translational distance models or semantic matching models focus on exploring the KG triples solely, random walks-based and ontology-aware also include additional information, namely the hierarchical information. This hierarchical perspective seems highly advantageous, especially in the context of ontology-rich KGs [Xiong et al., 2022].

This work distinguishes between graph embeddings and KG embeddings [Su et al., 2018]. While graph embeddings focus on generating latent representations over graphs in a general sense, KG embeddings are tailored explicitly to KGs, emphasizing semantic understanding and capturing meaningful knowledge about the entities and their connections. Embedding methods such as node2vec [Grover and Leskovec, 2016] and deepWalk [Perozzi et al., 2014] are classified as graph embeddings, not KG graph embeddings, because they are tailored for undirected graphs. Despite this, generating embeddings for KG entities remains feasible using these methods, though with certain adaptations. Table 2.2 summarizes the embedding methods surveyed in this work for each category.

Matrix Factorization

In the early 2000s, embedding algorithms were mainly matrix factorization-based. Matrix factorization methods represent a graph in the form of a matrix (e.g., node adjacency matrix, Laplacian matrix, node transition probability matrix, among others) and factorize it to obtain the embedding [Cai et al., 2018]. Isomap [Balasubramanian and Schwartz, 2002] and locally linear embedding [Roweis and Saul, 2000] were the pioneer efforts.

The locally linear embedding algorithm is based on node proximity matrix factorization. Assuming the data lies on a nonlinear manifold which locally can be approximated linearly, it uses two main steps: (i) locally fitting hyperplanes around each node n_i, based on its k nearest neighbors, and calculating reconstruction weights, and (ii) finding lower-dimensional coordinates

Table 2.2: Summary of representative embedding methods.

Category	Method	Type
Matrix Factorization	Locally linear embedding [Roweis and Saul, 2000]	Graph
	Isomap [Balasubramanian and Schwartz, 2002]	Graph
Translational Distance	TransE [Bordes et al., 2013]	KG
	TransH [Wang et al., 2014]	KG
	TransR [Lin et al., 2015]	KG
	TransD [Ji et al., 2015]	KG
	TranSparse [Ji et al., 2016]	KG
	KG2E [He et al., 2015]	KG
Semantic Matching	RESCAL [Nickel et al., 2011]	KG
	DistMult [Yang et al., 2015]	KG
	HolE [Nickel et al., 2016]	KG
	ComplEx [Trouillon et al., 2016]	KG
Deep learning-based	SDNE [Wang et al., 2016]	Graph
	DNGR [Cao et al., 2016]	Graph
Random walk-based	DeepWalk [Perozzi et al., 2014]	Graph
	node2vec [Grover and Leskovec, 2016]	Graph
	RDF2Vec [Ristoski and Paulheim, 2016a]	KG
	Metapath2vec [Dong et al., 2017]	KG
	RDF2Vec order-aware [Portisch and Paulheim, 2021]	KG
Ontology-aware	Onto2Vec [Smaili et al., 2018a]	KG
	Opa2Vec [Smaili et al., 2018b]	KG
	OWL2vec* [Chen et al., 2021a]	KG
	EL [Kulmanov et al., 2019]	KG
	BoxEL [Xiong et al., 2022]	KG

y_i for each n_i, by minimising a mapping function based on these weights. The isomap algorithm is based on graph Laplacian eigenmaps. Isomap finds the shortest path between two nodes as the geodesic distance between them.

In these two approaches, the matrix that holds proximity in terms of similarity between nodes and their neighbors is factorized to obtain the embeddings. Thus, first-order proximity that captures the direct neighbor relationships between nodes is preserved. However, locally linear embedding assumes that every node is a linear combination of its neighbors' embedding space. In contrast, Laplacian eigenmaps obtained the embedding by extracting the eigenvectors of the graph Laplacian matrix. Since the proximity matrix construction and the eigendecomposition

of the matrix are time and space-consuming, matrix factorization methods are not scalable for large graphs.

Translational Distance

Translational distance embedding approaches exploit distance-based scoring functions [Wang et al., 2017]. The translational distance models' basic idea is that each fact represents the distance between the two entities, usually after a translation carried out by the relations [Wang et al., 2017]. TransE [Bordes et al., 2013] is the most representative translational distance model. Given an observed fact (h, r, t), the relation is interpreted as a translation vector r so that the embedded entities h and t can be connected by r with $h + r \approx t$ when (h, r, t) holds. The scoring function is then defined as the distance between $h + t$ and given by

$$-\|h + r - t\|_{1/2}\,. \tag{2.1}$$

A drawback of TransE is that it cannot deal well with one-to-many and many-to-many relations. To address this challenge, there are several extensions of TransE: TransH [Wang et al., 2014] introduces a hyperplane for each relation r (relation-specific hyperplane) and projects h and t into the hyperplane; TransR [Lin et al., 2015] introduces a space for each relation r (relation-specific space), rather than hyperplanes; TransD [Ji et al., 2015] and TranSparse [Ji et al., 2016] simplify TransR.

TransE and its extensions model entities and relations as deterministic points in vector spaces. However, entities and relations can also be modeled as random variables. For instance, KG2E [He et al., 2015] uses multivariate Gaussian distributions to draw random vectors of entities and relations.

Semantic Matching

The semantic matching approaches exploit similarity-based scoring functions by matching latent semantics of entities and relations embodied in their vector space representations [Wang et al., 2017]. RESCAL [Nickel et al., 2011] takes the inherent structure of relations into account by employing the tensor factorization. Entities have a unique latent-component representation, regardless of their occurrence as subjects or objects. Each relation is represented as a matrix $(M_r t)$ that models pairwise interactions between latent factors. The score of a fact (h, r, t) is defined by a bilinear function

$$\beta(h, r, t) = h^T M_r t\,, \tag{2.2}$$

where $h, t \in \mathbb{R}^d$ are d-dimensional vector representations of entities h and t, and $M_r \in \mathbb{R}^{d\times d}$ is a matrix associated with the relation. The idea is to associate a bilinear form β_r with each relation r in such a way that for all entities h, t it holds that $\beta_r\,(h, r, t) \approx 1$ if (h, r, t) holds and $\beta_r\,(h, r, t) \approx 0$ otherwise.

DistMult [Yang et al., 2015] simplifies RESCAL by restricting the matrix associated with the relation to diagonal matrices. However, it can only deal with undirected networks. HolE [Nickel et al., 2016] combines the simplicity of DistMult with the power of RESCAL. ComplEx [Trouillon et al., 2016] also extends DistMult by introducing complex embeddings to handle various binary relations. Table 2.3 shows the scoring functions of four semantic matching models.

Table 2.3: Scoring functions of semantic matching models for a given fact (h, r, t).

Model	**Scoring Function**
RESCAL	$h^T M_r t$
DistMult	$h^T diag(r) t$
HolE	$r^T (h * t)$
ComplEx	$Re\left(h^T diag(r) \bar{t}\right)$

Deep Learning-based

Due to its robustness and effectiveness, deep learning has been widely used to encode the graph into a low-dimensional space. An autoencoder is a type of artificial Neural Network (NN) and consists of two parts, the encoder and the decoder. The encoder maps input data to a representation space, while the decoder maps the representation space to a reconstruction space. Both the encoder and decoder contain multiple non-linear functions. An autoencoder aims to minimize the reconstruction error of the output and input by its encoder and decoder. The idea of adopting an autoencoder is similar to matrix factorization in terms of neighborhood preservation. For instance, if the autoencoder's input is the adjacency matrix, the reconstruction process will make the nodes with similar neighborhoods have similar embedding. SDNE [Wang et al., 2016] and DNGR [Cao et al., 2016] are based on deep autoencoder architecture.

SDNE is a semi-supervised deep model with multiple layers of non-linear functions to preserve the highly non-linear graph structure. Specifically, SDNE represents nodes by their high-dimensional neighborhood vectors and feeds to the autoencoder to preserve second-order proximity. It also incorporates laplacian embedding's proximity measure into the autoencoder to preserve first-order proximity. The second-order proximity is used by the unsupervised component to capture the global network structure. In contrast, the first-order proximity is used by the supervised component to preserve the local network structure.

DNGR uses a random surfing model to capture more global information than a random walk and generate a positive pointwise mutual information matrix. The random surfing model is inspired by the PageRank model [Bianchini et al., 2005]. Then, the positive pointwise mutual information matrix based on the probabilistic co-occurrence matrix is calculated. Finally, deep

NNs for graph representations generate embeddings by applying a stacked denoising autoencoder to the positive pointwise mutual information matrix.

Random Walk-based

In graph theory, random walks can be explored to capture structural relationships between nodes [Su et al., 2018]. A graph can be transformed into node sequences by performing truncated random walks, which preserve the nodes' neighborhood structures. The random walk-based approaches are built upon two main steps: (i) producing entity sequences from walks in the graph to produce a corpus of sequences that is akin to a corpus of word sequences or sentences; (2) using those sequences as input to a neural language model that learns a latent low-dimensional representation of each entity within the corpus of sequences.

DeepWalk [Perozzi et al., 2014] adopts skip-gram [Ling et al., 2015], a famous deep model for neuro-linguistic programming that embeds words into a low-dimensional space by incorporating the context of words in sentences. Skip-gram aims to maximize the co-occurrence probability among the words that appear within a window w. DeepWalk first samples a set of paths from the input graph using random walks. Each path corresponds to a sentence from the corpus, where a node corresponds to a word. The paths are then used to train a neural language model, which estimates the likelihood of a specific sequence of nodes appearing in a graph. Once the training is finished, each instance in the graph is represented as a vector of latent numerical features. Projecting such latent representations of instances into a lower-dimensional feature space shows that semantically similar instances appear closer to each other. Node2vec [Grover and Leskovec, 2016] introduces a different biased strategy for generating random walks and exploring diverse neighborhoods. The biased random walk strategy is controlled by two parameters: the likelihood of visiting immediate neighbors (breadth-first search behavior), and the likelihood of visiting entities that are at increasing distances (depth-first search behavior). Neither DeepWalk nor node2vec takes into account the direction or type of the edges, and therefore they are not considered KG embeddings.

Metapath2vec [Dong et al., 2017] proposes random walks driven by metapaths that define the node type order by which the random walker explores the graph. RDF2Vec [Ristoski and Paulheim, 2016a] is inspired by the node2vec strategy, but it considers both edge direction and type, making it particularly suited to KGs. In RDF2Vec, edge direction is taken into account, enriching the learning approach's semantics, and Word2vec methods are employed over random walks on the RDF graph to produce the embeddings. Several variants of RDF2vec have been explored in the last few years. RDF2Vec order-aware [Portisch and Paulheim, 2021] uses a structured word2vec model [Ling et al., 2015], a variation of skip-gram that is sensitive to the order of entities in the graph walks. This enables the differentiation between whether a predicate emerges prior to or subsequent to the entity in question. The generation of walks has been a

field of study. While the initial implementation employs random walks, other possibilities have been investigated. Cochez et al. [2017] performed extensive research about the use of different heuristics for biasing the walks, e.g., frequency of predicates, PageRank, etc. In Vandewiele et al. [2018], walk strategies with teleportation within communities are analysed.

Ontology-aware

More recently, KG embedding approaches aim to tailor representations by considering different semantic, structural or lexical aspects of the underlying ontology of a KG.

Onto2Vec [Smaili et al., 2018a] also uses language modeling approaches to generate vector representations of entities in ontologies by combining formal ontology axioms and annotation axioms from the ontology. An ontology is a set of axioms, each of which constitutes a sentence. OPA2Vec [Smaili et al., 2018b] extends Onto2Vec by considering the metadata contained in ontologies. Ontologies contain a large amount of metadata as annotation axioms that describe different aspects of ontology classes, relations or instances. OPA2Vec also offers the option of using a pre-trained language model over biomedical literature to bootstrap the KG embedding. However, both Onto2vec and OPA2Vec treat each axiom as a sentence and therefore do not explore the semantic relationships between axioms.

EL [Kulmanov et al., 2019] and BoxEL [Xiong et al., 2022] embeddings are geometric approaches that account for the logical structure of the ontology (e.g., intersection, conjunction, existential quantifiers). Both of them construct specific loss functions for logical axioms by transforming logical relations into geometric relations. However, while EL represents classes as open balls in the embedding space and relations as translations, BoxEL represents classes as boxes, entities as points inside the boxes that they should belong to, and relations as the affine transformation between boxes and/or points.

OWL2Vec and its successor OWL2Vec* [Chen et al., 2021a] encode the semantics of an ontology by considering the graph structure, the lexical information and the logical constructors. OWL2Vec* also uses neural language models that receive as input three distinct documents: (i) the structure document that contains the random walks performed in the KG; (ii) the lexical document that includes sentences generated with the lexical information, such as entity names, comments, and definitions; (iii) the combined document from the structure document and the entity annotations to preserve the link between entities and the lexical information.

2.1.2.2 Semantic Similarity

A semantic similarity measure can be defined as a function that estimates the closeness in meaning between two entities. Most state-of-the-art methods fall in the category of taxonomic semantic similarity (also referred to as ontology-based semantic similarity, or only semantic similarity), which compares ontology entities based on the taxonomic relations within the on-

tology graph [Pesquita et al., 2009]. KG embedding similarity can also be used as semantic representation [Kulmanov et al., 2021].

An expert generally designs taxonomy semantic similarity measures based on assumptions about how an ontology is used and what should constitute a similarity. They extensively use the taxonomical aspect of an ontology, comparing classes based on subclass/superclass relations. In KGs, this translates to measuring the similarity between either ontology classes or KG entities that are described through links to a set of ontology classes [Pesquita et al., 2009]. Taxonomic semantic similarity measures can be categorized according to the instances they intend to compare. The following subsections briefly explain the main characteristics of taxonomic semantic similarity measures for each category and describe how can KG embedding semantic similarity be computed.

Taxonomic Measures for comparing Ontology Classes

Several measures for calculating the semantic similarity between two classes have been developed, as shown in Table 2.4. There are essentially two types of semantic similarity measures for comparing classes in a graph-structured ontology: edge-based and node-based measures [Harispe et al., 2015].

Edge-based measures rely on algorithms designed for graph analysis, which are generally used straightforwardly. A broad diversity of techniques can be used to estimate the distance between two nodes in a graph. The most common technique selects either the shortest path or the average of all paths when more than one path exists. Rada et al. [1989] were among the first to use the shortest path technique. This approach compares two classes defined in a semantic graph by edge counting. Pekar and Staab [2002] proposed a measure based on the length of the longest path between two classes' lowest common ancestor and the root (maximum common ancestor depth) and the length of the longest path between each of the classes and that common ancestor. More recently, semantic similarity measures based on random walk techniques have been discussed [Fouss et al., 2007]. Most of these measures assume that the distance between all the relationships in an ontology is constant or depth-dependent. Neither assumption is valid in most of the existing ontologies, so edge-based measures are rarely used.

Node-based measures consider classes or instances as sets of properties distinguished from the graph. More recent measures explore the notion of Information Content (IC), a measure of how specific and informative a class is. IC gives semantic similarity measures the ability to weigh the similarity of two classes according to their specificity. IC can be calculated using external data or based on intrinsic properties.

- The extrinsic IC of a class c is defined as inversely proportional to $p(c)$, the probability to encounter an instance of c in a collection of instances. The first extrinsic IC definition was proposed by Resnik [1995] and was based on the number of occurrences of a class in

a corpus of texts and given by

$$\mathrm{IC}_{\mathrm{Resnik}}(c) = -\log p(c) \tag{2.3}$$

where $p(c)$ is the probability of class c in the corpus. The main drawback of measures based on extrinsic IC is that they automatically reflect the class usage bias.

- The intrinsic IC only considers structural information extracted from the ontology, such as the number of descendants, ancestors, or depth. For example, the formulation proposed by Seco et al. [2004] is based on the number of direct and indirect descendants and given by

$$\mathrm{IC}_{\mathrm{Seco}}(c) = 1 - \frac{\log\left[\mathrm{N\_descendants}(c) + 1\right]}{\log\left[\mathrm{N\_classes}\right]} \tag{2.4}$$

where N_descendants(c) is the number of indirect and direct descendants of class c (including class c), and N_classes is the total number of classes in the ontology.

Semantic similarity measures based on the concept of IC can be applied to the common ancestors that two terms have. There are two main approaches: consider the most informative common ancestor (i.e., ancestor with the highest IC) or consider all disjoint common ancestors (i.e., common ancestors that do not subsume any other common ancestor). For instance, the measures proposed by Lin [1998] and Jiang and Conrath [1997] relate the IC of the most informative common ancestor to the information of the classes being compared.

Recently, a few approaches combining taxonomic semantic similarity with ML have been proposed. GARUM [Traverso-Ribón and Vidal, 2018] is based on a supervised regression algorithm that receives several similarity measures of hierarchy, neighborhood, shared information, and attributes and then predicts a final similarity score.

Table 2.4: Examples of semantic similarity measures for comparing ontology classes.

Category	**Measure**
Edge-based	Rada et al. [1989]
	Pekar and Staab [2002]
Node-based	Resnik [1995]
	Lin [1998]
	Jiang and Conrath [1997]
	Seco et al. [2004]

Taxonomic Measures for comparing Knowledge Graph Entities

Calculating semantic similarity for two entities, each annotated with a set of classes, typically employs one of two approaches: pairwise, where pairwise comparisons between all classes annotating each entity are considered; groupwise, where set, vector or graph-based measures are employed, circumventing the need for pairwise comparisons [Pesquita et al., 2009].

In pairwise approaches, the semantic similarity is calculated between terms in one set and terms in the other (using node-based or edge-based measures for comparing terms). The most common methods of measuring the similarity between sets of classes have been pairwise approaches based on node-based class measures, namely Resnik [1995], Lin [1998], and Jiang and Conrath [1997]. Pairwise scores are then summarised using an aggregation strategy. There are two combination strategies employed in pairwise measures: all pair techniques, where every pairwise combination of terms from the two sets is considered; and best pairs techniques, where only the best-matching pair for each term is considered. The classic aggregation strategies are maximum, minimum, and average. More sophisticated strategies have also been proposed: best-match maximum, and best-match average. For example, $\text{Resnik}_{\text{Max}}$ and $\text{Resnik}_{\text{BMA}}$ are pairwise approaches based on the class-based measure proposed by Resnik [1995] in which the similarity between two classes corresponds to the IC of their most informative common ancestor. This pairwise approach is used with two combination variants, maximum

$$\text{Resnik}_{\text{Max}}(e_1, e_2) = \quad \max\{\text{sim}(c_1, c_2) : c_1 \in \text{A}(e_1), c_2 \in \text{A}(e_2)\} \tag{2.5}$$

and best-match average

$$\text{Resnik}_{\text{BMA}}(e_1, e_2) = \frac{\sum_{c_1 \in \text{A}(e_1)} \max_{c_2 \in \text{A}(e_2)} \text{sim}(c_1, c_2)}{2|\text{A}(e_1)|} + \frac{\sum_{c_2 \in \text{A}(e_2)} \max_{c_1 \in \text{A}(e_1)} \text{sim}(c_1, c_2)}{2|\text{A}(e_2)|} \tag{2.6}$$

where $|\text{A}(e_i)|$ is the number of annotations for entity e_i and $\text{sim}(c_1, c_2)$ is the semantic similarity between the class c_1 and class c_2 and is defined as

$$\text{sim}(c_1, c_2) = \max\{\text{IC}(c) : c \in \{\text{Ancestors}(c_1) \cap \text{Ancestors}(c_2)\}\} \tag{2.7}$$

where $\text{Ancestors}(c_i)$ is the set of ancestors of t_i.

In groupwise approaches, the measures can directly compare the sets of classes according to information defined in the ontology. Purely set-based and vector-based approaches are not common. In vector-based approaches, the sets are compared through their vector representations, with each term corresponding to a dimension, using vector similarity measures. In most cases, these measures are not relevant to be used, considering they do not take into account the similarity of the elements composing compared sets. Nevertheless, graph-based approaches are widely used. For example, SimGIC is a groupwise approach proposed by Pesquita et al. [2007],

based on a Jaccard index in which its IC weights each class, and given by

$$\text{SimGIC}(e_1, e_2) = \frac{\sum_{c \in \{\text{A}(e_1) \cap \text{A}(e_2)\}} \text{IC}(c)}{\sum_{c \in \{\text{A}(e_1) \cup \text{A}(e_2)\}} \text{IC}(c)} \tag{2.8}$$

where $\text{A}(e_i)$ is the set of annotations (direct and inherited classes) for entity e_i.

More recently, Traverso et al. [2016] also proposed a graph-based similarity measure based on the knowledge encoded in ancestors or hierarchies, neighborhoods, and node degrees or specificity.

Knowledge Graph Embedding Similarity

KG embedding similarity can also be used as semantic representation. After employing embedding methods, each node is represented by a vector. Therefore, it is possible to compute the embedding similarity of two nodes by computing the distance of their corresponding vectors. If v_i and v_j denote the vector representations of nodes n_i and n_j, respectively, the embedding similarity $sim(n_i, n_j)$ between nodes n_i and n_j is given by the distance between their vectors v_i and v_j in the Euclidean space. The distance can be computed by the cosine distance:

$$sim(n_i, n_j) = cos(v_i, v_j) == \frac{v_i.v_j}{\|v_i\| \, \|v_j\|}, \tag{2.9}$$

where $v_i.v_j$ is the dot product of v_i and v_j.

2.1.2.3 Comparison of Knowledge Graph Semantic Representations

Two types of semantic representations are analysed in detail: KG embedding methods and semantic similarity.

Within KG embedding methods, entities and relations are embedded into a low-dimensional continuous vector space while preserving syntactic or semantic properties. Section 2.1.2.1 outlines multiple types of KG embedding methods that explore different properties of the KG [Wang et al., 2017]. Depending on the nature of the KG, some embedding methods may be more appropriate. This is especially relevant when considering ontology-rich KGs. This is due to the fact that a majority of methods focus on triple and only a subset attempt to explore the graph structure, lexical information and logical constructors of the ontology [Chen et al., 2021a]. In the context of such KGs exploiting longer, or more indirect, relations, either through random walks on graphs or through utilizing the semantics, is more likely to achieve better results [Kulmanov et al., 2021]. A drawback of embedding methods, in general, is their lack of interpretability since each dimension of the embedding does not have any meaning.

In semantic similarity, two entities are compared based on their description in the KG. Section 2.1.2.2 distinguish two types of semantic similarity: taxonomic semantic similarity and KG

embedding similarity. Taxonomic semantic similarity measures, generally designed by an expert, rely on a set of assumptions about how an ontology is used and what should constitute a similarity. Numerous semantic similarity measures have been proposed, and the challenge lies in selecting the most suitable one for a given application, as the behavior of these measures varies based on their specific applications [Kulmanov et al., 2021]. Moreover, given that taxonomic semantic similarity measures rely on the assumptions made by experts about what similarity is, they may not be generalizable to all domains and applications. On the other hand, KG embeddings can also be used to compute similarity using distance measures applicable to real-valued vectors. Unlike taxonomic measures that focus solely on hierarchical relationships, embedding similarity takes into account all types of relationships, offering a potentially more accurate similarity. However, it is essential to note that embeddings are not explicitly trained to bring two similar entities together, potentially leading to representations in space that may not accurately reflect semantic proximity [Jain et al., 2021]. A general disadvantage of semantic similarity is that the information of the KG is reduced to a single point.

Beyond individually evaluating several KG embedding methods and distinct semantic similarity measures, it is important to discuss the difference between these two types of representations. The primary distinction lies in the fact that KG embedding methods, by design, generate single entity representations, whereas semantic similarity measures generate pair representations. This is relevant, depending on the application, because some applications, such as node classification, need a representation of an entity, while others, such as relation prediction, need pair representations. The second difference relies on interpretability. Semantic similarity generates features that are hand-crafted and interpretable. For instance, when depicting a pair of proteins with a semantic similarity value of 0.8 for GO, one can infer that the two proteins are 80% similar in the biological processes they perform. In contrast, KG embeddings generate non-interpretable features. This is relevant when considering applications in complex domains, such as the biomedical domain. Lastly, these two types of representation can also differ in their expressiveness. Similarity-based representations are normally reduced to one point, while embeddings represent entities in a multi-dimensional space.

There are limitations that extend to all existing semantic representations, namely the fact that they assume that KGs reside under the closed world assumption (which is not true for dynamic real-world KGs) and do not adequately use the knowledge in the form of negation [Kulmanov et al., 2021]. In addition, these semantic representations treat the entire input KG uniformly. Although, depending on the task, some portions of the KG are more relevant, the decision that the representation should only be generated using a subgraph remains a manual decision handled by experts rather than an automated process. Another limitation lies in the lack of explainability associated with these semantic representations when used as features. Even semantic similarity, when computed using the whole KG, provides little insight into the relationships between entities.

2.2 Machine Learning

ML focuses on the development of algorithms and statistical models that enable computer systems to improve automatically through experience [Jordan and Mitchell, 2015]. Supervised ML algorithms play a critical role in data mining processes as a tool for identifying valid, novel, potentially useful patterns in data and making predictions. Supervised ML algorithms learn some initially unknown function given a set of labeled training examples to predict an output based on input features. In supervised classification tasks, the output is a predefined label. In supervised regression tasks, the output is a continuous value.

In the scope of this Ph.D. research, a clear distinction is made between classical supervised methods, which are designed to handle numerical vectors as inputs, such as the previously introduced semantic representations, and GNN-based approaches specifically designed to process inputs structured as graphs such as KGs. This section explores several ML techniques, encompassing both classical methods and GNN approaches. Additionally, the performance metrics are analysed. These metrics are essential for quantitatively evaluating the ML techniques. The section finishes by discussing the emerging topic of Explainable Artificial Intelligence (XAI) to emphasise the importance of interpretability and transparency in increasingly complex ML models.

2.2.1 Classical Machine Learning Methods

Classical supervised ML methods are designed to learn patterns from labeled data, enabling them to make predictions on unseen data. The following subsections present several classical ML methods.

Linear Models

Linear Regression (LR) [Poole and O'Farrell, 1971] assumes there is a linear relationship between the independent and dependent variables. The equation for a simple LR with one independent variable can be written as

$$y = mx + b \tag{2.10}$$

where y is the dependent variable (the variable that is being predicted), x is the independent variable (the variable that is being used to make the predictions), m and b are the linear relationship coefficients. The goal is to find the coefficients of the linear relationship that minimize the difference between the predicted values and the actual observed values of the dependent variable. This difference is often quantified by calculating the sum of the squared differences between the predicted and actual values.

Bayesian Ridge (BR) [Brown and Zidek, 1980] is also a linear model but uses the Bayes theorem to find the posterior distribution over all parameters and avoid overfitting. BR incor-

porates regularization by introducing a penalty term that discourages the model from assigning too much importance to any single feature. In this way, the loss function minimizes the error but also penalizes large values of the parameters.

Decision Trees

Decision Tree (DT) [Quinlan, 1990] is a predictive model that predicts the value of a target variable by learning a set of decision rules inferred from the data features. The decision rules are structured in a tree, with a single root node at the top and branches that lead to leaf nodes at the bottom. Each non-leaf node represents a decision based on a specific feature, and each leaf node represents a class label (for classification) or a numeric value (for regression).

In DT, trees are constructed by beginning with the root node that contains the whole learning sample and then splitting a node into two child nodes repeatedly. The basic idea of tree growth is to choose, among all the possible splits at each node, a split whose resulting child nodes are the "purest". There are different methods to measure purity, such as Gini impurity for classification and mean squared error for regression. To make a prediction for a new instance, the input features are passed down the tree from the root node to a leaf node based on the decisions made at each non-leaf node. The prediction at the leaf node is then returned as the final prediction for the instance.

Overfitting is a frequent problem with DTs, which can cause them to learn the training data too well and perform poorly on unseen data. Overfitting usually happens when the model memorizes the training data and then fails to generalize. Methods like cost-complexity pruning can help with overfitting. Pruning involves removing branches from the tree to simplify it. Setting the minimum number of samples required at a leaf node or setting the maximum depth of the tree can also be used to avoid this problem.

There are several DT algorithms, such as ID3, C4.5, and CART, that handle features, splitting criteria, pruning, and missing values differently. ID3 was one of the earliest DT algorithms. It builds trees using a top-down, recursive approach. C4.5 is an improvement over ID3 that introduces the concept of using continuous attributes by converting them into discrete values. Furthermore, it incorporates pruning to reduce overfitting after building a full tree. CART employs a "greedy" approach by building binary trees directly while minimizing the chosen impurity measure.

Genetic Programming

Genetic Programming (GP) [Poli et al., 2008] is an evolutionary computation technique inspired by Darwinian natural selection and Mendelian genetics. It is a population-based search procedure that can evolve solutions to complex problems of different domains. Although GP is commonly perceived as a search algorithm, especially in the context of optimization problems,

it also can be used to automatically evolve solutions by using data [Shapiro, 1999]. This latter scenario aligns with the broader classification of classical ML methods.

Implementation-wise, GP differs from genetic algorithms in its representation of individuals. While the genetic algorithm representations are typically fixed numerical length strings, GP representations are variable-length structures containing whatever ingredients are given to solve the problem. One of GP's significant strengths is its ability to explore large search spaces with a diverse population of free-form individuals and produce potentially readable transparent models without compromising predictive ability.

Tree-based GP is the most common type of GP. Here, the solutions/programs are represented as parse trees that are readily translated to readable strings (e.g. LISP-like expressions). The variables and constants in the solution constitute the terminal set in GP, as they are only admitted as terminal nodes of the trees. In contrast, the function set contains operators that can be used to combine elements (terminals and functions) and can only appear as internal nodes in the trees. Regarding evolution, the first step is population initialization, which generates a set of potential solutions to a given problem. The second step is fitness evaluation, where each candidate solution is evaluated and assigned a fitness value that quantifies how well it solves the problem. The fitter programs are selected more often to breed and thus pass their characteristics to their offspring, so the population tends to improve in quality along with successive generations. The selected parents are engaged in breeding using independently applied genetic operators like subtree crossover and mutation. The selection for survival happens after the application of the genetic operators and decides, from among parents and offspring, which individuals will be part of the new generation. The evolutionary process continues until a given stop condition is verified (e.g., maximum number of generations or fitness reaching a certain value), after which the program with the best fitness is returned as the best model found.

Ensemble Models

The goal of ensemble methods is to combine the decisions from multiple models to improve the overall performance. Two very well-known ensemble methods are bagging (e.g., Random Forest (RF)) and boosting (e.g., eXtreme Gradient Boosting (XGB)).

RF [Breiman, 2001] builds several estimators independently and then combines their predictions through a voting scheme. It starts by creating multiple subsets of the training data through bootstrapping. Each subset is obtained by randomly selecting data points from the original training dataset with replacement. Then, for each subset, a DT is trained using only a random subset of features. This is a key difference between RF and DT. While RF only select a subset of those features, DT considers all features. When making predictions for classification tasks, the class that receives the most votes from the individual trees is chosen as the final prediction. For regression, the individual predictions are averaged.

XGB [Chen and Guestrin, 2016] builds base estimators sequentially and tries to reduce the bias of the combined estimator through gradient boosting. In XGB, each new model is trained to correct the errors made by the previous models. Models are trained sequentially, with each new model focusing on the samples that were misclassified or had high residuals by the previous models. XGB includes regularization techniques to control the complexity of the trees and prevent overfitting. The final prediction is obtained by adding up the predictions from all the individual trees, but each tree's contribution is weighted based on its performance. Trees that make more accurate predictions have higher weights.

Multilayer Perceptron

Multilayer Perception (MLP) [Rumelhart et al., 1986] is a class of feedforward artificial NNs that learn non-linear functions through backpropagation of errors. An MLP consists of multiple layers. The first layer of the network receives the raw input data. Each neuron in the input layer represents a feature of the input data. Then, hidden layers composed of multiple neurons perform computations on the data. These computations involve weighted sums of inputs followed by an activation function. The activation function introduces non-linearity into the network, allowing it to learn complex relationships. The output layer produces the predictions based on the computations performed in the hidden layers. The number of neurons in the output layer depends on the task the MLP is designed for. For instance, in a classification task, each neuron in the output layer might represent a class, and the neuron with the highest activation would be the predicted class.

Training an MLP involves using a process called backpropagation. The objective function measures the error between the output scores and the desired scores. The machine then modifies its internal adjustable weights to minimize this error. The process of backpropagation, used to calculate the gradient vector of an objective function with respect to the weights in a multilayer stack of modules, involves the application of the chain rule for derivatives. The backpropagation equation can be applied repeatedly to propagate gradients through all modules, starting from the top (where the network produces its prediction) to the bottom (where the external input is fed). Stochastic gradient descent is the most common procedure to update the weights of a model.

2.2.2 Graph Neural Networks

Deep Learning [LeCun et al., 2015] is a set of representation-learning methods. A deep-learning architecture is a multilayer stack of simple modules. Each module in the stack transforms its input to increase both the selectivity and the representation's invariance. With multiple non-linear layers, a system can implement extremely intricate functions of its inputs that are simultaneously sensitive to minute details and insensitive to large irrelevant variations.

Many applications of deep learning use feedforward NN architectures [LeCun et al., 2015]. The weighted sum of the inputs is passed through a non-linear function to go from one layer to the next. One of the most popular non-linear functions is the rectified linear unit. GNN are powerful deep NNs for graph-structured data [Gu et al., 2018]. However, handling graph data presents a challenge, differing from the conventional data employed in typical deep learning models. An intuitive approach would be flattening the adjacency matrix and employing the resulting vector as input for a MLP. Nevertheless, due to the unordered nature of adjacency matrices, the order represented in the vector becomes arbitrary, leading to a representation that fluctuates based on different permutations of the adjacency matrix. GNNs tackle this challenge using a permutation-invariant model. This approach enables the extraction of node representations that rely on the graph's structure and any available feature information.

Although there are already many variations of GNN, the defining feature of a GNN is that it employs a type of neural message passing in which vector messages are exchanged between nodes and updated using NNs. Every message-passing iteration updates the embedding of each node by aggregating information from the neighborhood of the node and integrating that information with the node's previous embedding. The result is an updated embedding that includes information from an increasing number of neighborhoods. For a GNN with two layers, this process ensures that the final embeddings of each node incorporate both structural and feature-based information from the entire two-hop neighborhood of the node.

Several distinct methods for message passing are utilized by GNN, namely Graph Convolutional Neural Network (GCNN). The "graph convolution" operation applies the same linear transformation to all the node neighbors, followed by mean pooling and non-linearity. By stacking multiple graph convolution layers, GCNNs can learn node representations by using information from distant neighbors [Bruna Estrach et al., 2014; Duvenaud et al., 2015; Defferrard et al., 2016; Kipf and Welling, 2017]. Very recently, relational GCNNs [Schlichtkrull et al., 2018] were proposed as a generalization of GCNNs developed for dealing with highly multi-relational data, such as KGs, and were applied to link prediction and entity classification.

2.2.3 Performance Metrics

In ML, performance metrics are essential to evaluate how well a model performs on a given task. Performance measures can vary based on the ML task. The most common metrics for classification tasks include:

- **Precision** measures the proportion of true positive predictions out of all positive predictions (true positives and false positives):

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{2.11}$$

- **Recall** measures the proportion of true positive predictions out of all positive instances (true positives and false negatives).

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{2.12}$$

- **F-measure or F1-score** is the harmonic mean of precision and recall.

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.13}$$

- **Weighted average F-measure** accounts for class unbalance by computing the F-measure for each class and then calculating the average of all computed F-measures, weighted by the number of instances of each class:

$$\text{Weighted average F-measure} = \frac{\sum_{c \in C} \text{F-measure}_c \times \text{Support}_c}{\sum_{c \in C} \text{Support}_c} \tag{2.14}$$

  where C is the set of classes, F-measure_c is the F-measure computed for class c, and Support_c is the number of instances in class c.

- **Receiver operating characteristic curve** shows the trade-off between true positive rate and false positive rate.

The most common metrics for regression tasks include:

- **Mean squared error** the average of the squared differences between predicted values (X) and actual values (Y):

$$\text{Mean squared error} = \frac{1}{D} \sum_{i=1}^{D} (X_i - Y_i)^2 \tag{2.15}$$

- **Pearson correlation coefficient** measures the linear correlation between the predicted values (X) and actual values (Y):

$$\text{Pearson correlation coefficient} = \frac{covariance(X, Y)}{\sigma_X \sigma_Y} \tag{2.16}$$

2.2.4 Explainable Artificial Intelligence

The scientific community has long recognized the potential of Artificial Intelligence (AI) as a tool for scientific discovery, with ML, pattern mining, and reasoning playing crucial roles in several steps of the scientific process [Mjolsness and DeCoste, 2001]. Despite this, the vast majority of scientific projects that utilize AI do not prioritize explainability [Roscher et al., 2020]. In the

biomedical domain, both the complexity of the data and the natural phenomena under study highlight the necessity of domain knowledge to support explainability [Holzinger et al., 2017]. XAI is gaining traction as a potential solution.

XAI is capable of providing a human-understandable description of the logic, behavior or factors that influence the learning process [Barredo Arrieta et al., 2020]. Furthermore, XAI aims to address several key objectives, including promoting algorithmic fairness, detecting potential biases or issues in training data, ensuring that algorithms function as intended, and bridging the gap between the ML community and other scientific disciplines [Gilpin et al., 2018]. Any means to reduce the model's complexity or simplify and explain its outputs can be considered an XAI approach. An explanation should be domain-dependent, which means that it should be set within a context that depends on the task, background knowledge, and expectations of the user [Gunning et al., 2019].

The definitions of interpretability and explainability are still under debate. These terms are often interchangeably used in literature. However, some works distinguished them. For the sake of convenience, this work adopts the distinction proposed by Barredo [Barredo Arrieta et al., 2020]: (i) interpretability is the ability to identify relationships within the models' inputs and outputs; (ii) explainability is the ability to provide explanations that are simultaneously an accurate proxy of the predictor and comprehensible to humans.

Several taxonomies have been proposed to classify explainable AI techniques. XAI approaches can be classified into two types: models that are transparent by design, such as DT, LR models, and GP models [Mei et al., 2022], or post-hoc explainability techniques that are used to improve the interpretability of models that are not transparent by design. Post-hoc explainability techniques can be categorized as either model-specific (tailored to explain a particular ML model) or model-agnostic if they are applicable to any ML model. Post-hoc techniques may include visual explanations, explanations by example, explanations by simplification (one of the most used techniques for producing simplified models that are only representative of certain sections of a model), or feature relevance explanations (consist of computing the relevance score of each variable in the ML model, since the comparison of the importance values unveils the most relevant variables to the predictions made by the model).

XAI techniques that focus on explaining opaque models, such as KG embeddings and GNN, with post-hoc techniques are currently gaining widespread popularity [Palmonari and Minervini, 2020]. CRIAGE [Pezeshkpour et al., 2019] explores how adding and removing facts affects the general performance of KG embedding models. Very similarly, Kelpie [Rossi et al., 2022b] also explores the removal of facts in embedding-based link prediction models. XTransE [Zhang et al., 2020b] and SemanticCrossE [d'Amato et al., 2022] are also approaches for generating explanations to link prediction. XTransE adopts TransE [Bordes et al., 2013] and generates rules to explain link predictions. SemanticCrossE adopts CrossE [Zhang et al., 2019] to compute predictions, but the explanations are based on semantic similarity. Concerning GNNs, Yuan et al.

[2022] presents a comprehensive analysis of existing explanation methods for GNN categorizing them into three main types: feature-based methods [Pope et al., 2019], perturbation-based methods [Ying et al., 2019; Luo et al., 2020; Lin et al., 2021], and surrogate methods [Huang et al., 2022; Vu and Thai, 2020]. Features-based methods rely on gradients or hidden feature map values as the approximations for feature importance. Perturbation-based methods explore the impact of different input perturbations on output predictions. GNNExplainer [Ying et al., 2019] is one of the first works that try to identify the important features of the GNN that influence a specific prediction using perturbations. Finally, surrogate-based models use interpretable and more simple models to approximate the predictions of the GNN for the neighboring areas of the input example.

Although most XAI techniques focus on explaining opaque models with post-hoc techniques, some works instead support using inherently interpretable ML models. Rudin [2019] identified several reasons why opaque explanations are not the best choice. In the first place, explanations do not represent what the original opaque model computes to the extent that a complex model would not be needed if the explanations were perfectly faithful. In addition, explanations often do not make sense, and even when they do, they do not provide enough detail to understand what the opaque model is doing.

XAI methods can also be classified according to the approach employed to generate explanations. Data-driven approaches generate explanations solely from data without relying on external sources such as prior knowledge. Knowledge-infused approaches include a representation of the domain knowledge in the field of application, which is explored to generate user-comprehensible and context-aware explanations of the mechanistic functioning of the AI system and the knowledge used [Chari et al., 2020]. Regarding the level of scope, explanations can be categorized into local and global. A local explanation focuses on explaining the prediction of a model for a specific instance or input. A global explanation, on the other hand, refers to an overall understanding of a model's behavior across its entire input space.

Durán [2021] also distinguishes scientific XAI and other types of XAI. According to Durán [2021], current approaches offer explanations that answer *how* the algorithm reached a given output but not why. To address this issue, it is proposed the concept of a genuine scientific explanation within the context of AI that should answer the *why* question. Durán [2021] also provides a structural definition of explanation, as composed of at least three essential components. The first component is the *explanans* that corresponds to the unit that generates the explanation. The second component is the *explanandum* that corresponds to the unit under explaination. The last component is the *explanatory relation* that establishes the links between the *explanans* to the *explanandum*. The evaluation of explanations is also addressed. A meaningful explanation for humans involves defining the explanatory power, specifying the conditions that make an explanation relevant to humans, and ensuring the ability to incorporate the explanation into broader knowledge.

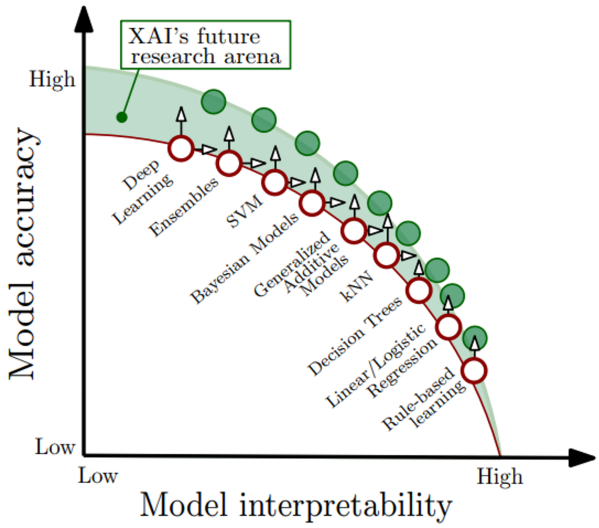


Figure 2.4: Distribution of distinct ML methods across the explainability axis. This figure was extracted and adapted from Arrieta et al. [2020].

Another ongoing debate is the trade-off between interpretability and performance [Arrieta et al., 2020]. In ML, models with high predictive performance, like deep NN, often come at the cost of being more complex and opaque. On the other hand, simpler and more interpretable models, such as DTs or LR, may sacrifice some predictive power. Figure 2.4 situates the different ML methods along the axis of explainability and illustrates the performance-explainability trade-off. Researchers are working on developing methods and techniques that enhance the interpretability of complex models without compromising their performance, contributing to advancements in the field of XAI.

Explanation evaluation and explainability quantification is still an open challenge [Barredo Arrieta et al., 2020] that must be addressed to further cement the role of XAI in science. Quantitative evaluation [Jiang et al., 2021], case-based evaluation [Padhiar et al., 2021], and user studies [Purificato et al., 2021] are being used to evaluate the quality of explanations.

2.3 Machine Learning over Knowledge Graphs

Semantic information is recognized as a valuable knowledge resource in supporting ML tasks [Ristoski and Paulheim, 2016b]. Given the ability of KGs to provide context to data, they are a unique opportunity for ML. KG tasks include entity classification, link prediction, graph classification, relation prediction, among others. There are two categories of tasks: in-KG tasks and

out-of-KG tasks [Wang et al., 2017]. While in-KG tasks, such as link prediction, are conducted within the scope of the KG, out-of-KG, such as entity classification, graph classification, and relation prediction, use external information that is not part of the KG.

Classification is a prevalent problem in ML that is based on predicting a label for an entity given some characteristics of the entity. The classification problem is supervised, i.e., it learns a classification model based on labeled training data. In the context of KGs, entity classification aims to assign a class label to each node. Since most KGs contain entities of more than two different types, multi-label classification, which allows assigning more than one class to an entity, is particularly important [Paulheim, 2017]. Various techniques exist to perform node classification on KGs [Breit et al., 2023], from using KG embedding methods combined with classifiers [Steenwinckel et al., 2022; Nickel et al., 2011] to GNN architectures that receive the graph directly [Rhee et al., 2018; Ioannidis et al., 2019].

Link prediction is concerned with predicting missing relationships (represented as edges in the KG) between entities (represented as nodes in the KG). Since KGs are often incomplete (e.g., in social networks, friendship links can be missing between two users who actually know each other) and some of the edges they contain are incorrect, link prediction is seen as a KG completion task, i.e., adding missing knowledge to the KG. Link prediction task has been tested extensively in the literature [Portisch et al., 2022]. There is a family of KG embedding methods that focus on link prediction [Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015; Ji et al., 2015; Yang et al., 2015] as a geometric task. The goal of these methods is to project the KG into a vector space so that, for each triple $< h, r, t >$, a translation operation between the head h and the relation r yields a result close to the tail t.

Graph classification shares similarities with entity classification, but instead of predicting the label of a node, the objective is to predict the label of a graph [Tsuda and Saigo, 2010]. In addressing this issue, the typical approach involves computing specific graph features across the entire graph. Graph kernels represent a widely adopted methodology in several domains [Borgwardt et al., 2005; Gärtner et al., 2003; Shervashidze and Borgwardt, 2009; Shervashidze et al., 2011, 2009]. Nevertheless, their scalability is limited. Consequently, deep learning approaches have been proposed [Zhang et al., 2018; Wu et al., 2021; Xie et al., 2022].

Another task, and perhaps one less known and defined in the literature, is relation prediction. This work defines relation prediction as the task of predicting the existence of a specific relation between two entities when the relation itself is not explicitly defined in the KG.

In addition to the role of KGs for many downstream tasks, KGs also play a crucial role in XAI [Hitzler et al., 2020]. XAI approaches benefit from being able to link ML models to representations of domain knowledge [Holzinger et al., 2017; Wollschlaeger et al., 2020]. Therefore, KGs and ontology-rich KGs represent an unparalleled solution to XAI. Since KGs provide a structured representation of the underlying knowledge, they can be used in every phase of AI system —pre-modeling, in-modeling, and post-modeling [Rajabi and Etminani, 2022]. In a

comprehensive literature review focused on the use of KGs for explainability, Rajabi and Etminani [2022] demonstrated that KGs are predominantly employed for post-hoc interpretability through inference and reasoning [Wang et al., 2019b]. However, despite its popularity, KG reasoning poses computational challenges due to properties such as transitivity, symmetry, and asymmetry [Chen et al., 2020b]. An alternative involves integrating KGs in pre-modelling XAI to extract features. KGs afford explainable features, i.e., features that are semantically enriched and contextualized, making them more comprehensible for end-users. Notably, KGs have found widespread application in various domains, with healthcare being particularly prominent. This aligns with the recognition that the lack of trust is the main barrier to the adoption of AI in clinical practice [Glikson and Woolley, 2020].

Chapter 3

Related Work

Since the focus of the evaluation approach of this Ph.D. research is to assess the ability of the novel approaches to improve KG-based semantic representations used in supervised learning in the biomedical domain, this chapter identifies the most relevant literature related to approaches that explore biomedical KGs for biomedical applications using ML.

The first step was to define which keywords should be used in the search: ”machine learning” *AND* (”knowledge graph” *OR* ”ontology”) *AND* (”biomedical” *OR* ”medical” *OR* ”life sciences” *OR* ”protein” *OR* ”gene” *OR* ”drug” *OR* ”disease”). However, since Google Schoolar's Advanced Search does not support two *OR* operators in a single query, the original search query was split into two distinct queries. The first query is: ”machine learning” *AND* ”knowledge graph” *AND* (biomedical *OR* medical *OR* ”life sciences” *OR* protein *OR* gene *OR* drug *OR* disease). The second query is: ontology *AND* ”machine learning” *AND* (biomedical *OR* medical *OR* ”life sciences” *OR* protein *OR* gene *OR* drug *OR* disease) *AND* -”knowledge graph”. Both searches were restricted to publications between 2019 and 2023. The first 300 results were collected for the first query, and the first 200 results for the second query. After this, the exclusion criteria were: non-English, non-open access entries, and non peer-reviewed. 441 publications met the requirements. The subsequent phase involved reviewing the titles of the selected papers and eliminating those that appeared to be entirely completely out-of-scope and those that fell under the category of surveys or literature reviews. The number of publications was reduced to 193. Subsequently, the final phase involved an in-depth assessment of the complete content of each paper, aimed at removing any paper that deviated from the established scope. Poster papers were also removed. This process led to a refinement in the number of publications, ultimately resulting in a collection of 74 papers (Figure 3.1).

Table 3.1 lists all the papers reviewed. The publications are categorized according to the

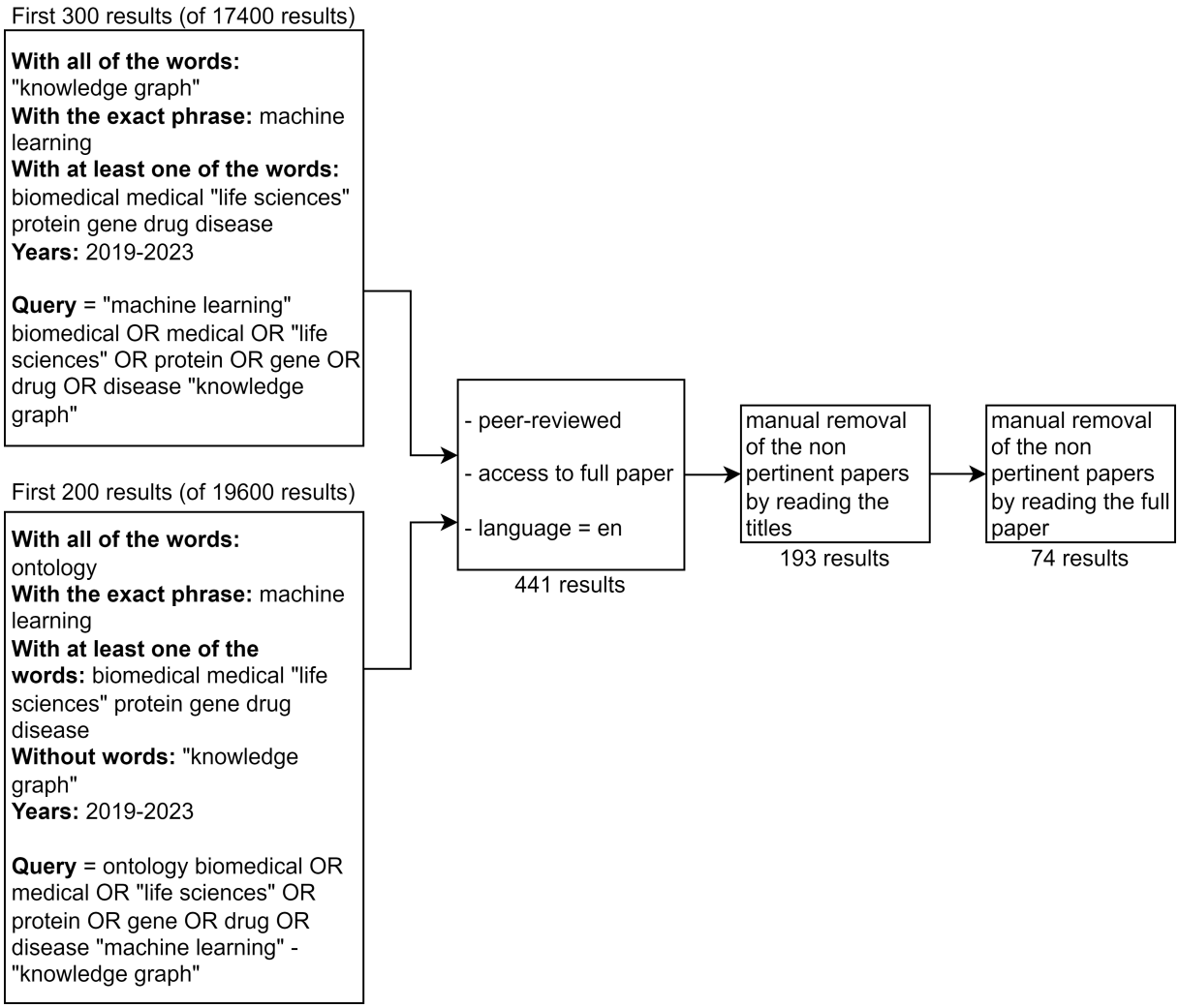


Figure 3.1: Literature review diagram.

ML approach, the semantic representation employed, the biomedical application, and whether they express concerns about explainability.

Table 3.1: Summary of existing work using KG-based semantic representation for biomedical applications. The awareness for explainability is divided into three categories: not aware (✗), aware (✓) or very aware (✓✓).

Reference	Semantic Representation	ML	Task	Explainability Awareness
Karim et al. [2019]	KG embedding	Convolutional Long Short-Term Memory	Drug-drug interaction prediction	✗
Shen et al. [2019]	KG embedding	Support Vector Machine (SVM), RF, MLP, DT, LR	Disease-phenotype association prediction	✗
Fang et al. [2019]	KG embedding	SVM	Chronic obstructive pulmonary disease prediction	✗
Wang et al. [2019c]	Semantic similarity	SVM	PPI prediction	✗
Kim et al. [2019]	Semantic similarity	RF, SVM, LR	Drug-disease association prediction	✗
Li et al. [2019]	Semantic similarity	Gradient boosting algorithm	Pathogenicity prediction	✗
Chen et al. [2019]	Semantic similarity	Ensemble Stacked Generalization	PPI prediction	✗
Celebi et al. [2019]	KG embedding	LR, Naive Bayes, and RF	Drug-drug interaction prediction	✗
Sosa et al. [2019]	KG embedding	Not applicable	Drug-disease association prediction	✓✓
Mohamed et al. [2019]	KG embedding	Not applicable	Drug-target interaction prediction	✗
Biswas et al. [2019]	KG embedding	Not applicable	Disease co-morbidity prediction	✗
Sousa et al. [2020]	Semantic similarity	GP	PPI prediction	✓
Gilvary et al. [2020]	Semantic similarity	XGB, SVM, LR	Drug-drug interaction prediction	✓✓
Kulmanov and Hoehndorf [2020]	Not applicable	Hierarchical classification NN	Genotype–phenotype prediction	✓

Table 3.1 Continued from previous page

Reference	Semantic Representation	ML	Task	Explainability Awareness
Zhang et al. [2020a]	KG embedding	NN	Protein function prediction	✗
Lei et al. [2020]	KG embedding	Artificial NN, gated recurrent unit, and ResNet	Disease prediction	✓
Nováček and Mohamed [2020]	KG embedding	Not applicable	Adverse drug reactions prediction	✗
Chai [2020]	KG embedding	Bidirectional long short-term memory network	Thyroid disease prediction	✗
Mohamed et al. [2020]	KG embedding	Not applicable	Drug-drug interaction prediction	✗
Lin et al. [2020]	Not applicable	GNN	Drug-drug interaction prediction	✗
Mukherjee et al. [2021]	Semantic similarity	RF	GDA prediction	✓✓
Sousa et al. [2021]	Semantic similarity, KG embedding semantic similarity	GP	PPI prediction	✓
Xiong et al. [2021]	Semantic similarity, KG embedding	NN	Drug-disease association prediction	✗
Ye et al. [2021]	KG embedding	Neural factorization machine	Drug-target interaction prediction	✗
Dai et al. [2021]	KG embedding	Not applicable	Drug-drug interaction prediction	✗
Zhang et al. [2021b]	KG embedding	Not applicable	Drug-target interaction prediction	✓✓
Zhang et al. [2021e]	KG embedding	DT	PPI prediction	✓
Hu et al. [2021a]	KG embedding	Not applicable	GDA prediction	✓✓
Wang et al. [2021a]	KG embedding	Naive Bayes	Adverse drug reactions prediction	✓✓
Zhang and Che [2021]	KG embedding	SVM, LR, RF, DT	Drug-disease association prediction	✗
Zhang et al. [2021a]	KG embedding	LR	Adverse drug reactions prediction	✗
Kawichai et al. [2021]	KG embedding	XGB	Drug-disease association prediction	✗

Table 3.1 Continued from previous page

Reference	Semantic Representation	ML	Task	Explainability Awareness
Zhang et al. [2021c]	KG embedding	TextCNNBiLSTM-Attention Network	Drug-drug interaction and drug-target association prediction	✗
Nunes et al. [2023]	KG embedding	RF, XGB	GDA prediction	✗
Wang et al. [2021b]	KG embedding	Not applicable	Drug-drug interaction prediction	✗
Krämer et al. [2021]	KG embedding	Linear Model	Gene function prediction	✓
Kanatsoulis and Sidiropoulos [2021]	KG embedding	Not applicable	Drug-disease association prediction	✗
Chen et al. [2021b]	KG embedding	NN	Drug-drug interaction prediction	✓
Yu et al. [2021]	Not applicable	GNN	Drug-drug interaction prediction	✓✓
Bourgeais et al. [2021]	Not applicable	Fully connected NNs	Phenotype prediction	✓✓
Wang et al. [2021c]	Not applicable	GNN	Synthetic lethality prediction	✓
Che et al. [2021]	Not applicable	Attention GNN	Drug-disease association prediction	✗
Lu et al. [2021]	Not applicable	GCNN	Patient readmission risk prediction	✗
Bresso et al. [2021]	Graph kernels	DT, XGB	Adverse drug reactions prediction	✓✓
Wang et al. [2022a]	Semantic similarity	Inductive matrix completion	GDA prediction	✗
Daluwatumulle et al. [2022]	KG embedding and KG embedding semantic similarity	SVM, Naive Bayes, RF, DT, XGB, LR, k-Nearest Neighbors	Drug-disease association prediction	✗
Bonner et al. [2022]	KG embedding	Not applicable	Drug-disease association prediction	✗
Wang et al. [2022b]	KG embedding	Conv-Conv Model	Drug-target interaction prediction	✗
Binder et al. [2022]	KG embedding	XGB	GDA prediction	✓
Ren et al. [2022b]	KG embedding	Deep NN	Drug-drug interaction prediction	✗

Table 3.1 Continued from previous page

Reference	Semantic Representation	ML	Task	Explainability Awareness
Ren et al. [2022a]	KG embedding	Deep NN	Drug-drug interaction prediction	✗
Su et al. [2022a]	KG embedding	NN	Drug-drug interaction prediction	✗
Joshi et al. [2022]	KG embedding	NN	Adverse drug reactions prediction	✗
Ye et al. [2022]	KG embedding	NN	GDA prediction	✗
Yao et al. [2022]	KG embedding	Not applicable	Adverse drug reactions prediction	✗
Gavali et al. [2022]	KG embedding	RF	Kinase–substrate interaction prediction	✓✓
Su et al. [2022b]	Not applicable	GNN	Drug-drug interaction prediction	✓
Lan et al. [2022a]	Not applicable	Graph Attention Network	GDA prediction	✗
Lan et al. [2022b]	Not applicable	Transformer	GDA prediction	✗
Gao et al. [2022a]	Not applicable	GCNN	Drug-disease association prediction	✗
Gao et al. [2022b]	Not applicable	GCNN	GDA prediction	✗
Saadat et al. [2022]	Not applicable	GCNN	Drug recommendation	✗
Ma et al. [2023a]	KG embedding	RF	Drug-disease association prediction	✓✓
Carvalho et al. [2023]	KG embedding	LR, RF, Naive Bayes, SVM	Patient readmission risk prediction	✗
Hao et al. [2023]	KG embedding	Convolutional NN	Drug-drug interaction prediction	✗
Soman et al. [2023]	KG embedding	RF	Disease prediction	✓
Vilela et al. [2023]	KG embedding	Not applicable	GDA prediction	✗
Ye et al. [2023]	KG embedding	FMT-KNR	Syndrome differentiation	✗
Quan et al. [2023]	KG embedding	SVM, DT, RF	Drug-target interaction prediction and targetability prediction	✓
Bang et al. [2023]	KG embedding	XGB	Drug-disease association prediction	✓✓

Table 3.1 Continued from previous page

Reference	Semantic Representation	ML	Task	Explainability Awareness
Ma et al. [2023b]	KG embedding	Temporal convolutional NN	Organ failure prediction	✓✓
Zhu et al. [2023]	Not applicable	Graph Attention Network	Synthetic lethality prediction	✓✓
Krix et al. [2023]	Not applicable	GNN	Adverse drug reactions prediction	✓✓
Wu [2023]	Not applicable	GNN	Synthetic lethality prediction	✓

By analyzing the selected papers over time, it is possible to identify some trends. Figure 3.2 illustrates the distribution of different types of approaches across the reviewed papers and organized by year. The use of semantic similarity as a semantic representation seems to have lost popularity in recent years. In contrast, KG embeddings have become increasingly popular in biomedical applications, maintaining their relevance to the present day. GNN-based approaches, a more contemporary direction of research, have gained some popularity in recent years. Regarding explainability, Figure 3.3 depicts the distribution the explainability awareness across the reviewed papers and organized by year. It is evident that this aspect has become a hot topic, and there is an increasing concern about having explanations behind predictions in the context of biomedical research. However, there is still a long way to go. For the distribution of tasks, Figure 3.4 shows that a vast variety of tasks has benefited from the integration of KGs and ML, but it is worth highlighting the prevalence of applications that center around predicting relations between two biomedical entities.

In the following sections, the selected papers in the literature review and some older important works are explained in more detail for each semantic representation type.

3.1 Knowledge Graph Embeddings-based Approaches

As semantic representations, KG embeddings have been widely used for several tasks in the biomedical domain [Yue et al., 2019]. In scenarios where pair representations are used, most of these methods first learn distinctive vector representations for each KG entity. Subsequently, they often employ operators to merge the representations of the two entities forming the pair. Another option is to use the KG embedding directly, where scores computed by the trained embedding model are used to infer links between entities.

Protein or gene function prediction is the task of inferring the biological roles of proteins/genes, which is crucial for understanding their roles in various biological processes, cellular pathways, and disease mechanisms. Most approaches are based on PPI networks, amino acid

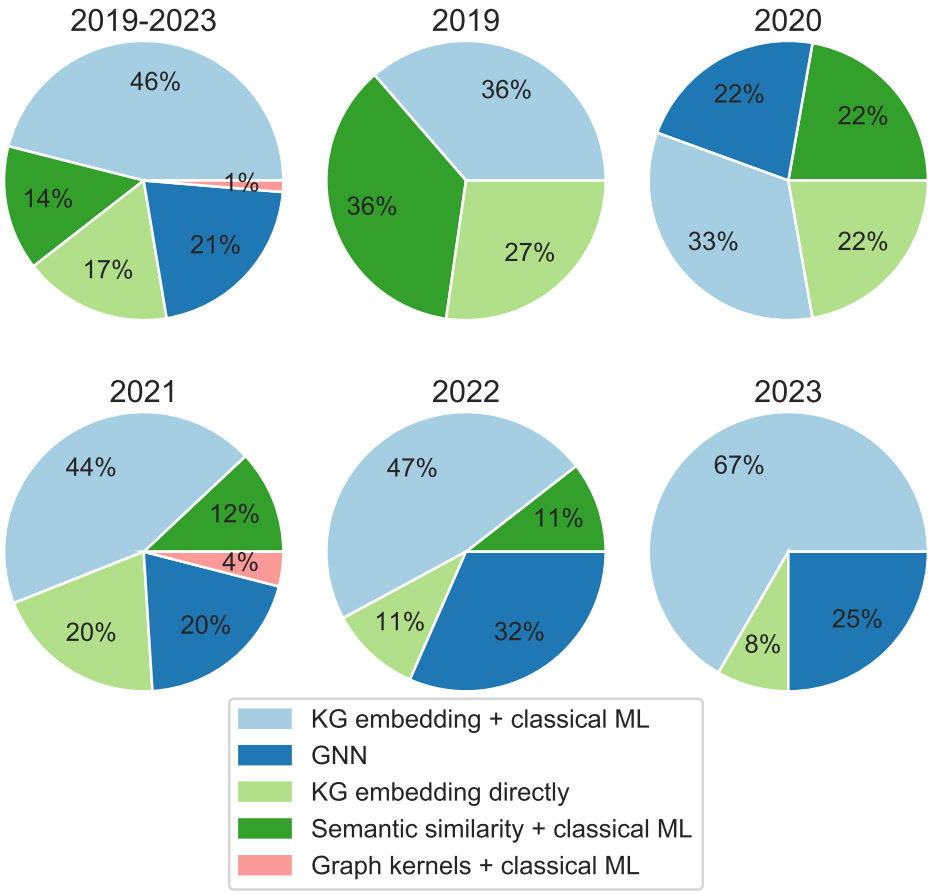


Figure 3.2: Distribution of different types of approaches across the reviewed papers and organized by year.

sequences, or other available information. Kulmanov et al. [2018] propose DeepGO, a more complex approach that predicts protein function using sequence and PPI networks. For PPI network, KG embedding methods are used to generate protein features. For GO, they created binary label vectors for each protein. The feature vectors are then passed to hierarchically structured classification layers. Zhang et al. [2020a] present DeepGOA that also uses protein sequences and PPIs for protein function prediction. Deepwalk algorithm is employed to generate the embeddings of the proteins encoding PPI information. Then, the two types of representations from the sequence and the PPI network are concatenated together to predict protein functions using several layers of NNs. Krämer et al. [2021] present the Coronavirus Network Explorer, a web interface to show interactive network visualizations of an algorithm for predicting gene functions. The main idea of the algorithm is to use gene embedding vectors computed over the KG as features in a linear model to predict the effect of a given gene on a given function.

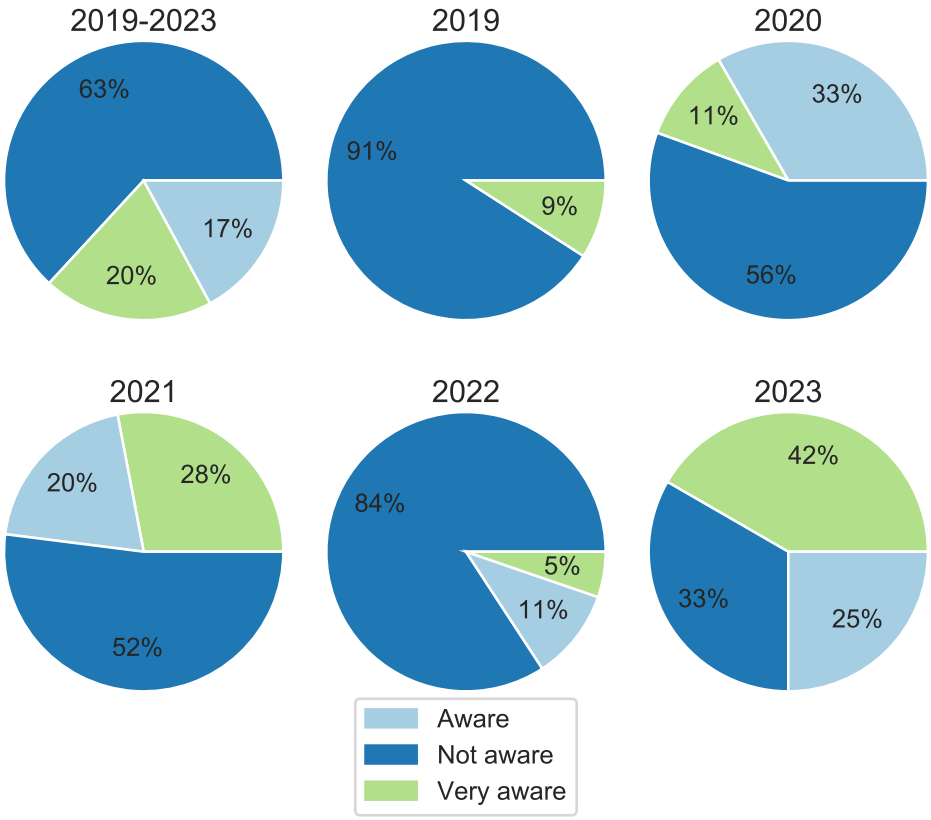


Figure 3.3: Distribution of the explainability awareness across the reviewed papers and organized by year.

PPI prediction corresponds to identifying interactions between different proteins that play a important role in various cellular processes. Zhang et al. [2021e] propose rule-based computational method that uses GO and kyoto encyclopedia of genes and genomes pathways to predict PPIs. The representation of each protein is a binary vector generated according to its functional annotation on each GO term or not. However, not all GO terms and pathways are used because Boruta feature filtering is applied to select the relevant features. DTs are then applied to learn a set of rules for PPI prediction. Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan [2022] present TransformerGO, an approach that uses a transformer architecture to predict PPIs using the GO KG. The first step is the generation of GO class embeddings using the node2vec approach. Then, the encoder and decoder receive as input those embeddings. Given the binary nature of the PPI prediction, the transformer architecture is trained to optimize the binary cross-entropy loss.

Disease prediction refers to the task of predicting the likelihood of an individual developing

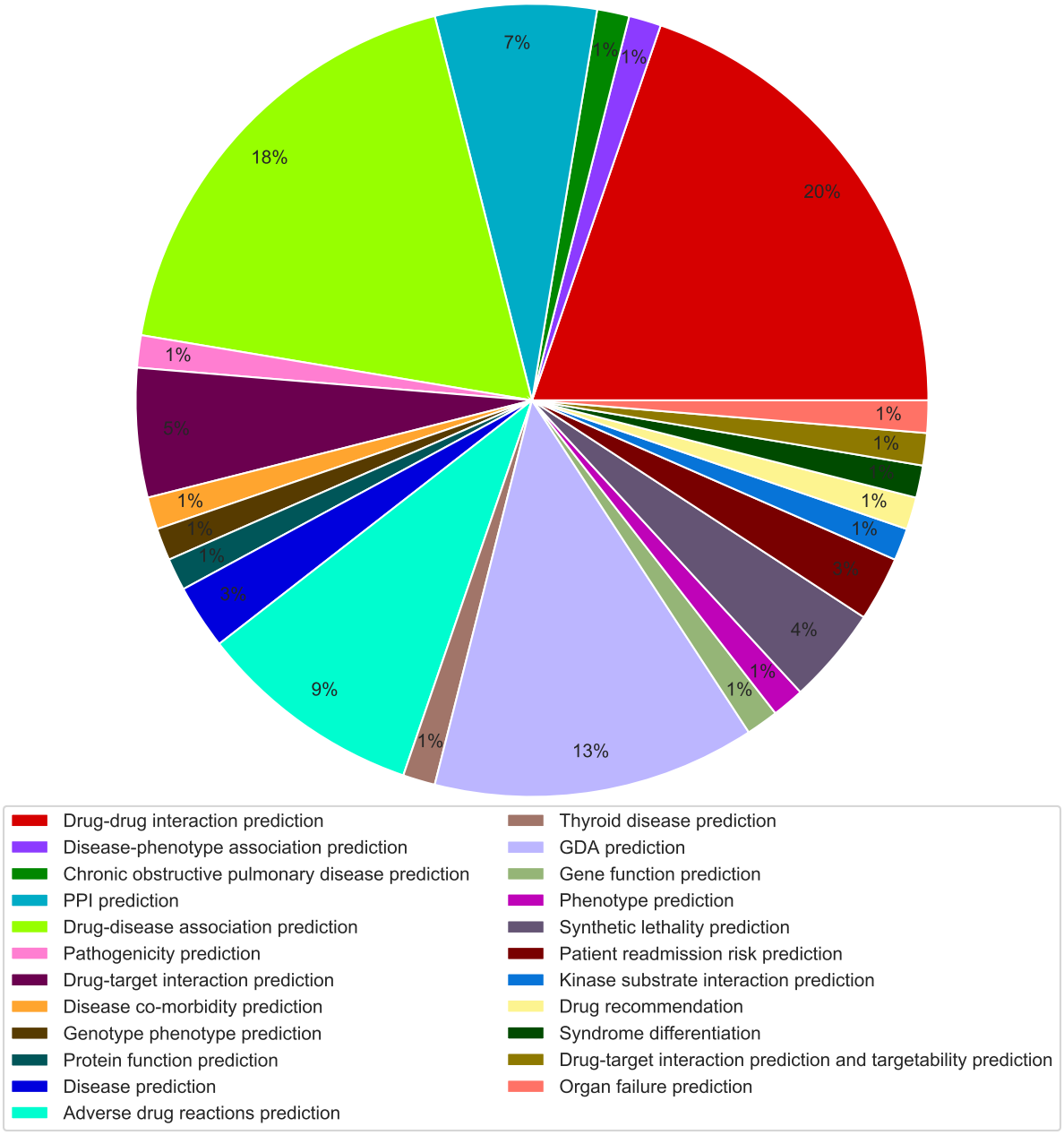


Figure 3.4: Distribution of the biomedical tasks across the reviewed papers.

a specific disease. This can involve analysing various factors, including phenotypes and medical history. Liu et al. [2018a] present DeepEHR, a framework combining medical notes and ontologies for disease prediction. DeepEHR train embeddings directly on medical notes using StarSpace, while diagnosis code embeddings are obtained using a bag-of-words approach. Then,

it uses several deep learning architectures for disease prediction. Fang et al. [2019] proposes a integrated method that uses a KG for chronic obstructive pulmonary disease for prediction. The KG is constructed from electronic medical records, and it includes relationships between diseases, symptoms, causes, risk factors, drugs, and side effects. The direct search simulated annealing algorithm integrated with SVM is used for classification. In Chai [2020], a KG where the entities represent thyroid patients, thyroid diagnosis results or thyroid medication, while relationships represent connections between different thyroid medical entities is generated. Then, the KG entities and relationships are transformed into low-dimensional vectors using a KG embedding method. Finally, a bidirectional long short-term memory network predicts the disease diagnosis. Soman et al. [2023] present TANDEM, a time-aware embedding approach for disease prediction. Using a modified version of PageRank, TANDEM generates several embeddings for an individual patient in a patient's timeline that are then aggregated and normalized to create a patient embedding vector. The temporal embeddings are then used to train a RF classifier for diagnosing Parkinson disease.

Drug-drug interaction prediction refers to the task of identifying potential interactions between two drugs. This task can have significant implications for patient safety and treatment outcomes, as they can lead to adverse effects, reduced efficacy, or even toxic effects. Some works apply existing KG embedding methods for drug-drug interaction prediction. Celebi et al. [2019] apply several classical ML methods, such as Naive Bayes, LR, and RF, on DrugBank KG using three KG embeddings - RDF2Vec, TransE, and TransD. Using the same KGs, in Karim et al. [2019], several embedding approaches (PGB, SimpleIE, KGloVe, TransE, CrossE, RDF2Vec) are also used to transform the information from the KG in a suitable format. The difference is that, instead of using classical ML algorithms, convolutional NN and long short-term memory network (Conv-LSTM) are combined for predicting drug interactions. Wang et al. [2021b] propose a novel framework for drug interaction prediction that uses biomedical KGs and biomedical texts to generate embeddings of the entities and labels. The KG embedding methods are obtained with translation methods such as TransE and TransR. The task of predicting drug interactions is cast as a link prediction problem. Chen et al. [2021b] present MUFFIN, a multi-scale feature fusion deep learning model for drug-drug interaction prediction. MUFFIN explores the information about the drug's molecular structure and the biomedical KG DRKG. Regarding the KG, MUFFIN employs TransE to obtain drug vector representations. Then, the representations (obtained from molecular structure and KG) are concatenated and then fed into the fully connected layer for binary-class, multi-class and multi-label drug-drug interaction prediction. Ren et al. [2022b] present BioDKG–DDI, an approach that uses local and global features extracted from biochemical KGs for predicting interactions between drugs. BioDKG–DDI employs three types of features: drug vectors using language models over the drug chemical structures; drug vectors extracted from the biochemical KG using the embedding method ComplEx; drug functional similarity matrix using the information about four types of receptors. All the features are fused

using a self-attention mechanism to predict drug interactions through deep NN. Very similar, in Ren et al. [2022a], DeepLGF is proposed. Instead of integrating a drug functional similarity matrix, DeepLGF extracts drug vectors using a GNN over drug–receptor matrices. Extensive experiments are conducted for DeepLGF, and the proposed approach is compared with baseline models. Hao et al. [2023] present 3WDDI a novel method that uses KG embeddings and three-way decision models to solve the uncertain decision of drug interaction prediction. 3WDDI employs ComplEx over a biomedical KG to generate embedding for candidate drugs. Then, based on the drug chemical structures of drug pairs, the pairs are divided into positive, negative and boundary. The KG embeddings are fed into a convolutional NN classifier to handle samples in the boundary region.

Other works propose novel KG embedding models for drug-drug interaction prediction. Mohamed et al. [2020] propose a novel KG embedding model, TriModel, to learn vector representations of drugs in a biomedical KG. TriModel extends DistMmult and ComplEx models, and it is based on tensor factorization. In this work, drug-drug interaction prediction is cast as a link prediction, and the embeddings are used to infer links between drugs based on their scores computed by the trained embedding model. Dai et al. [2021] propose a novel KG embedding method that employs adversarial autoencoders based on Wasserstein distances and Gumbel-Softmax relaxation for drug-drug interaction prediction. The autoencoder is used to generate high-quality negative samples. Drug-drug interaction prediction is cast as a link prediction task where the confidence score for each pair of drugs is estimated. Zhang et al. [2021c] present MHRW2Vec-TBAN, a novel approach that combines KG embeddings and NN. Regarding the KG embedding method, the MHRW2Vec model is employed. It includes metropolis hasting random walks and the language model, word2Vec. The embeddings are then the input for an improved network model TextCNNBiLSTM-Attention Network that is composed of TextCNN, Bi-LSTM and Attention layers. Su et al. [2022a] propose DDKG, an approach that starts by initializing drug representations using embeddings derived from drug attributes through an encoder-decoder layer. Then, it refines these representations by iteratively propagating and aggregating first-order neighboring information using attention weights. Finally, DDKG estimates the likelihood of drug interactions by considering pairwise drug representations and minimizing a binary cross-entropy loss function.

Drug-target association prediction is a task used in pharmaceutical research to identify potential biological molecules (usually proteins) that could be targeted by a drug to treat a specific disease or medical condition. Semantic matching and translational models have been widely used in drug-target association prediction. Mohamed et al. [2019] predict drug-target associations by applying KG embedding models, namely ComplEx, over a KG to enable scoring those associations in a link prediction task. Ye et al. [2021] present a unified framework, KGE_NFM, for predicting drug-target interactions by combining KG and recommendation system. This framework is divided into two important steps: learning low-dimensional representation for

KG entities using DistMult; and integrating the embeddings via neural factorization machine. Zhang et al. [2021b] propose a new approach for drug-target link prediction using KG embedding methods. The KG is built from SemMedDB and COVID-19 research literature. Regarding the embedding methods, the authors explore three types of KG embeddings that are directly used for link prediction: TransE and RotatE as translational models, DistMult and ComplEx as semantic matching models, and STELP. Wang et al. [2022b] propose KG-DTI, a novel KG-based deep learning method for drug-target interaction prediction. KG-DTI starts by using DistMult to obtain embeddings of the entities in KG. Subsequently, the drug-target vectors are combined and fed into a Conv-Conv module to extract relevant features. These extracted features are then input into a fully connected NN to predict drug-target interactions. In Quan et al. [2023], a novel approach, GraphEvo, is proposed for predicting the targetability and the druggability of genes. Regarding the KG, the ESKG is built by integrating GDAs, gene–gene interactions, biological processes, subcellular localization of proteins, drug–target associations, drug–disease associations and evolutionary data. For targetability prediction, TransE is used to generate embeddings for genes. The embeddings are then fed into an ensemble learning algorithm boosting. For drug-target interaction prediction, besides the gene embeddings, a GCNN is used to generate representations of drugs. For each drug-target pair, the representations are concatenated, and a DT is trained to compute a druggability score.

Random walk-based methods have also been used for drug-target association prediction. Hinnerichs and Hoehndorf [2021] develop DTI-Voodoo to predict candidate drugs for a given protein by learning representations of both drugs and proteins. To describe proteins, DTI-Voodoo uses protein functions, phenotypes resulting from a loss of function and amino acid sequences. Protein functions and phenotypes are classes of biomedical ontologies and, therefore, a KG embedding method based on random walks - DL2Vec - is used to generate representations. Finally, DTI-Voodoo calculates the similarity of drug and protein representations and predicts whether there is an interaction. In Bang et al. [2023], DREAMwalk (drug Repurposing through Exploring Associations using Multilayer random walk) is introduced. DREAMwalk is a random walk-based method that incorporates semantic information-guided teleportation. The generated walks are then fed to a neural language model to generate embeddings. An XGB classifier is trained to predict drug-disease associations.

In the same context, prediction of the adverse effects of a drug is also a very relevant task. Nováček and Mohamed [2020] employ a KG embedding model based on tensor decomposition for predicting possible adverse effects. It uses a KG containing facts about single drug side-effects, PPIs and protein-drug targets and casts adverse effects prediction as a link prediction task. TriVec is the chosen KG embedding method that extends the DistMult and ComplEx models. Wang et al. [2021a] build a tumor-biomarker KG from literature to predict potential adverse effects of antitumor drugs and provide explanations. The KG entities representing tumors, biomarkers, drugs and adverse drug reactions are transformed into matrix form that is then the

input for a Naive Bayesian model. Zhang et al. [2021a] propose a novel approach that uses a KG embedding method to generate embeddings for drugs and adverse reactions. Then, a binary classifier, LR, is trained to predict whether a given drug will cause the side effect. Regarding the KG embedding method, the KG is considered a corpus composed of triples that are given as input for the Word2Vec model. Joshi et al. [2022] combine a KG embedding method with a custom-made deep NN for predicting adverse drug reactions. The KG containing drugs, adverse reactions, indications, targets, pathways and genes is given as input to the Node2Vec algorithm. The drug and adverse reaction embeddings are concatenated and given as input features to a deep NN, which acts as a prediction model for adverse drug reaction classification. Yao et al. [2022] present MSTE that learns an embedding vector for drugs and side effects and then uses the scoring function to predict the complex relations of polypharmacy side effects as a link prediction task.

Another related task is predicting associations between drugs and diseases. It involves identifying and establishing connections between specific drugs and the diseases they can be used to treat. Several works use well-known KG embedding methods. Zhang and Che [2021] present a novel framework, DRKF, that builds a KG using medical literature and then uses that KG to generate embeddings that are given as input for a ML method to predict drug candidates for Parkinson disease. TransE, ConvE, ConTransE, distMult are employed as KG embedding methods. The last step of DRKF is using binary classifiers to predict a relationship between a drug and a disease. Xiong et al. [2021] present GraphPK, a multimodal framework that receives three types of features: biological features, known drug-diseases association-based features, and KG-based features. Regarding the biological features, label encoding is used for obtaining drug features and disease semantic similarity is used for generating disease features over disease descriptors. The known association-based features are extracted using a variant of a GCNN that receives a graph with drug-disease associations as input. With respect to KG-based features, TransD is employed to learn representations of entities in the KG. Finally, for each drug-disease pair, the three types of features for the disease and the drug are given as input to a multimodal NN to predict a score. Bonner et al. [2022] investigate the link prediction performance of five KG embedding models (ComplEx, DistMult, RotatE, TransE, and TransH) on two biomedical KGs, Heitonet and BioKG, for drug discovery.

The remaining works for drug-disease association prediction use other techniques. Sang et al. [2018] present SemaTyP, a method that constructs a biomedical KG with the relations extracted from PubMed abstracts and then learns entity representations to give as input for a LR model. To learn entity representations, SemaTyP first obtains all paths connecting diseases and drugs to consider as positive training data. Sosa et al. [2019] introduce a KG embedding method that models the uncertainty associated with literature-derived relationships and uses link prediction to generate new links. The idea behind this is that the proximity between the head, relation, and tail vectors is related to the confidence score associated with the triple. Kanatsoulis and

Sidiropoulos [2021] propose TeX-Graph that adopts a novel coupled tensor-matrix framework to generate embeddings and cast drug-disease association prediction as a link prediction problem. Ma et al. [2023a] propose KGML-xDTD that combines KG embedding methods and supervised ML methods for drug repurposing prediction. In the first step, the drug and disease embeddings are generated by concatenating the embedding generated with GraphSage over the KG and the embedding generated with PubMedBERT. In the second step, for each pair drug-disease, the corresponding embeddings are concatenated to generate a pair representation and used as input of an RF model to classify each drug–disease pair into three classes: "not treat," "treat," and "unknown".

Predicting GDA associations is another very relevant biomedical task that have significant implications for understanding of diseases, developing targeted therapies, and advancing personalized medicine. In Althubaiti et al. [2019], a neuro-symbolic deep learning approach to predict driver genes and mutations. It generates KG embeddings for gene functions and gene–phenotypes associations using OPA2Vec over a KG integrating several biomedical ontologies, namely GO, Cellular Microscopy Phenotype Ontology and Mammalian Phenotype Ontology. Subsequently, these embeddings serve as input for a deep artificial NN. Based on this approach, Hu et al. [2021a] present DGLinker, a web server that predicts new GDAs and includes tools to explore and interpret the results. Kawichai et al. [2021] present a meta-path-based approach that explores GO to link drugs and diseases and then train an ensemble method to predict drug-disease associations. Three types of associations are used to build a network: drug-disease, drugs-GO, and disease-GO. Similarly, three types of meta-paths are extracted: drug-GO-disease, drug-GO-drug-disease, and drug-disease-GO-disease. These meta paths are used to generate representations of drug-disease pairs, and XGB is then employed to classify drug-disease pairs. Binder et al. [2022] also present a metapath-based approach to identify potential disease-associated genes. The KG integrates heterogeneous datasets from the target central resource database. Then, the diseases and genes in the KG are converted to vectors by metapath matching. Finally, XGB is used as a predictive model. Ye et al. [2022] adopt a tensor factorisation model to generate two latent gene target and disease matrices that are then concatenated and fed into a dense NN to predict the clinical outcome of a gene target-disease pair. Vilela et al. [2023] also explore using KG embeddings to predict associations between genes and diseases. Three embedding algorithms - ComplEx, DistMult, and TransE - are used to produce vector representations of the KG entities. The prediction of GDAs is cast as a link prediction problem on the KG.

KG embeddings can also be used for several other biomedical tasks: disease categorization, phenotype prediction and patient readmission prediction. Lei et al. [2020] combine KG embeddings with deep learning to classify diseases. Knowledge representation is primarily achieved through the TransE model. The method layer is chiefly implemented by artificial NN, gated recurrent unit, and ResNet. Shen et al. [2019] present HPO2Vec+, a framework to enrich KG embeddings for the HP by incorporating biomedical knowledge bases, such as OMIM, DECIPHER,

and Orphanet. As an evaluation, HPO2Vec+ embeddings are used for a relation prediction task with four operators to obtain the pair representation and six ML models. Chen et al. [2020a] develop a novel approach that generates gene embeddings using DL2Vec over a KG where genes are described with respect to their associated phenotypes, functions of the gene products and anatomical location of gene expression. A pointwise learning-to-rank model is then used to predict associations between genes and diseases. DL2Vec extends OWL2Vec by incorporating more complex forms of axioms, such as the complexity of the axioms in a cross-species phenotype ontology. Patel et al. [2021] use node2vec to generate embeddings for genes and phenotypes corresponding to HP classes. Embeddings are fed to supervised classifiers, including LR, RF, XGB, LightGBM, NN. The experiments indicated that the LightGBM model ranked among the top-performing classifiers. Carvalho et al. [2023] present a novel approach that enriches electronic health records to build a KG and generate patient vector representations that are then processed by ML models for intensive care unit readmission prediction. RDF2Vec, OPA2Vec, and TransE are used to generate the embeddings. For readmission prediction, the problem is formulated as a binary classification problem, and four classical ML methods are used (Naive Bayes, LR, RF, SVM). Ye et al. [2023] present a new approach for syndrome differentiation that uses two types of data: chinese electronic record data and TCMKG, a KG about traditional Chinese medicine. Four KG embedding methods - TransE, DistMult, ComplEx, ConvKB - are used to generate vector representations of each KG entity. The text representations obtained with BERT and KG entity embeddings are fused using the FMT-KNR method to predict the nature of diseases through multilabel classification. Biswas et al. [2019] adopt ComplEx embedding model to predict disease pairs as a link prediction problem. Besides predicting disease associations from the benchmark dataset, a Markov clustering-based method is applied over a disease-protein-protein-interaction network to generate newly generated co-morbid disease pairs.

Other biomedical tasks include the prediction of subcellular localization of proteins or kinase–substrate interaction prediction. Cheng et al. [2018] propose pLoc-mGneg, a new approach to predict the subcellular localization of bacteria Gram-negative proteins. Proteins are represented according to the amino acid composition and GO information. To make the prediction, pLoc-mGneg adopts the multi-label Gaussian kernel regression classifier. Gavali et al. [2022] adopt a novel embedding method to learn representations of kinases and substrates over a phosphoproteomic KG. The novel embedding method, TripleWalk, is inspired by DeepWalk, but instead of sampling one node at a time, it samples one triple. These representations are then used as input to an RF classifier to predict interactions. Kinase–substrate interaction prediction is cast as a binary classification problem, and for each pair kinase-substrate, the respective embeddings are combined using the Hadamard operator. Ma et al. [2023b] present DKM, a novel approach that uses a medical KG (BCM-KG) for organ failure prediction in intensive care unit patients. The first step of DKM is to encode entities and relations in the BCM-KG to a low-dimensional space using the graph embedding model TransE. The second step is to employ

a temporal convolutional NN to predict organ failure.

In summary, several types of KG embedding models have been successfully applied to several biomedical applications. Despite the promising results, there are some challenges that need to be tackled, namely the knowledge evolution [Mohamed et al., 2021]. Biological knowledge evolves every day with the ongoing discovery of new entities and the introduction of novel associations between biological entities. However, KG embedding models can solely generate embeddings for entities that existed in the KG at the time the model is training. Another particularly relevant challenge is the lack of explainability since most KG embedding methods operate as opaque models. The embedding spaces they generate are complex, making it difficult for users to understand how specific relationships and entities are encoded.

3.2 Knowledge Graph Semantic Similarity-based Approaches

Semantic similarity can be used as a semantic representation or directly as features when the learning task takes as input pairs of entities for several biomedical tasks.

Some approaches use KG embedding methods to compute a graph embedding semantic similarity. Abdelaziz et al. [2017] present an extension of Tiresias, a similarity-based framework for predicting drug interactions. Tiresias uses local and global similarity-based features to measure the similarity between two drugs. Global features are derived from KG embeddings cosine similarity. Local features include chemical–protein interactome profile-based similarity, mechanism of action-based similarity, physiological effect-based similarity, pathways-based similarity, side effect-based similarity, metabolizing enzyme-based similarity, chemical structure similarity, among others. The resulting features are used for a logistic regression model to predict potential drug interactions. In Zong et al. [2017], DeepWalk is used to generate vector representations of drugs and targets using a heterogeneous network generated from biomedical datasets. The cosine similarity between drugs and targets is then calculated, and a rule-based inference method is applied to discover the drug-target associations. More recently, Alshahrani and Hoehndorf [2018] propose SmuDGE that generates disease and gene embeddings for GDA prediction. The KG includes gene–phenotype associations, disease–phenotype associations, interactions between genes and the PhenomeNET ontology. Regarding the embedding method, the authors generate a corpus and then use a skip-gram model to generate embeddings for genes and diseases in the graph. The embeddings are then used to compute the cosine similarity or as the input to a NN. SmuDGE is evaluated by predicting candidate genes for each disease and ranking them based on the similarity score or NN prediction score. Smaili et al. [2018b] present OPA2Vec, a novel embedding method that combines asserted and inferred logical axioms in ontologies with annotation axioms to produce vector representations. OPA2Vec is applied to PPI and GDA prediction in two different ways: by computing the cosine similarity between protein embeddings and using it as a prediction score for whether two proteins interact or not; by using the

embeddings themselves as an input for a NN. Daluwatumulle et al. [2022] adopt a novel approach based on KG embedding methods to predict candidate drugs for diseases. TransE, DistMult, ComplEx, HolE, ConvE, and ConvKB are employed to generate embeddings. Then, the cosine similarity between the embeddings of the pair is used to find drug-disease candidates. Furthermore, more drug candidates are extracted through link prediction. Finally, the concatenated drug and disease embeddings of the triples are fed to several classifiers (Naive Bayes, k-Nearest Neighbors, SVM, RF, DT, XGB, LR).

However, most of the approaches that use semantic similarity still focus on taxonomic semantic similarity measures. Several works propose new semantic similarity measure to predict relations between biomedical entities. Wu et al. [2006] present a new protein semantic similarity measure by comparing the relative specificity of pairs of GO terms assigned to them in similarity within GO. Only the cellular component and biological process subgraphs and their respective annotations are used in this study. Jain and Bader [2010] proposed an algorithm, Topological Clustering Semantic Similarity, that uses the taxonomic semantic similarity between GO terms annotated to proteins to distinguish true from false protein interactions. The central idea is to find subsets of GO terms defining similar concepts and score gene products belonging to a similar subset higher than if they belong to different sets. Liu et al. [2018b] propose a method that incorporates enrichment of GO terms by a gene pair in computing the taxonomic semantic similarity. This GO enrichment is incorporated by querying gene pair in the computation of IC of a GO term. The enrichment of a GO term by the pair of genes depends on whether the term is annotated by one gene or by both genes in the pair.

Other methods integrate semantic similarity with ML algorithms. In Kastrin et al. [2018], potential drug-drug interactions are also predicted using unsupervised and supervised classification algorithms on several large-scale several KGs (e.g., DrugBank, KEGG). The input features are topological and semantic measures, including anatomical therapeutic chemical classification system similarity, chemical structure-based drug similarity, MeSH-based similarity, and adverse drug effect-based similarity. Kim et al. [2019] propose a model for drug repositioning, in other words, predicting associations between drugs and diseases. Since similar diseases can be treated with similar drugs, they computed drug-drug taxonomic semantic similarity and disease-disease taxonomic semantic similarity. The features of drug and disease similarities are combined into a vector to represent each drug-disease association. Given the combinations of drug-disease similarities, prediction models are constructed using diverse classification algorithms. Clinical trial data validated new indications for 20 existing drugs and 31 herbal compounds. Li et al. [2019] propose a new approach, Xrare, that uses phenotype similarity measures to train a gradient tree boosting approach for predicting the pathogenicity of a variant. The similarity between two sets of phenotypes is measured using a new taxonomic semantic similarity called emission-reception information content that addresses the presence of imprecision and noise in phenotype annotations. Gilvary et al. [2020] propose CATNIP, a drug repurposing approach that is based

on computing the similarities between all possible pairs of drugs with known indications and then training a binary classifier (XGB, SVM or LR) to predict if a drug pair shares or not an indication. Three types of similarity are computed: structure similarity, network distance, and target semantic similarity computed using the Jaccard index between the targets listed for both drugs. Mukherjee et al. [2021] present DiGePred, a RF approach designed to identify candidate disease genes using features extracted from biological networks, genomics, evolutionary history, and functional annotations. Six features for each disease-gene are considered, namely the phenotype similarity corresponding to the overlap between the HP annotations of the two genes using a Jaccard similarity metric. Wang et al. [2022a] propose MLCDA, a framework that integrates several data sources, including ontologies, for circRNA-disease associations prediction. The representation of the genes is computed using similarities over the primary sequence information of circRNAs. The representation of the diseases is computed using semantic similarity measures over the disease ontology. MLCDA then employs principal component analysis to predict association rating scores for circRNA-disease pairs. This prediction is achieved by projecting the pairs into a potential space through inductive matrix completion.

Given the popularity of GO, several approaches explore the similarity over the GO KG using ML algorithms for a wide variety of biomedical tasks, namely PPI prediction. Zhang and Tang [2016] propose a GO-based method to predict PPIs by integrating different similarity measures. For each GO aspect (biological process, cellular component, and molecular function), five semantic similarity measures calculate the similarity score. In the end, a feature vector is obtained to characterize a protein pair by concatenating three feature vectors derived from the three GO aspects. The feature vectors are then used as input to train binary SVM. Chen et al. [2019] introduce an ensemble learning approach for PPI prediction that integrates multiple learning algorithms and different protein-pair representations: GO-based features, network-based features, and sequence-based physicochemical features. A GO-based feature is defined as one GO-term cluster indexed by the lowest common ancestor. Network-based features are derived from the constructed PPI network, where two proteins are linked based on the Resnik semantic similarity of their GO terms. In Wang et al. [2019c], gene expression similarities and semantic similarity computed over the GO are incorporated to measure relatedness between a pair of genes and predict PPIs. The problem is formulated as a LR problem, and SVM is used as a regressor.

GO KG has also been used to describe target proteins of drugs. Olayan et al. [2018] present a novel method - DDR - that first computes similarity measures between drugs and between target proteins and then applies a RF model to predict drug-target interactions. Multiple similarity measures between drugs and between target proteins are used, namely the protein similarity based on functional annotation using GO. Lee et al. [2019] develop a deep-learning model to predict the pharmacological effects of drug interactions. This model uses an autoencoder that receives as input similarity values between drug pairs, followed by a deep feed-forward network

that predicts the drug-drug interaction type. Three similarity measures are used: structural similarity, target gene similarity, and GO term similarity. The target gene similarity is based on the distance between the pairwise combination of the target genes in the functional interaction network. The GO term similarity is calculated in the same way.

But the uses of GO KG do not end with drug-drug interaction prediction. Asif et al. [2018] use GO-based gene functional similarities as input to supervised ML methods to predict GDA. The semantic similarity is computed using Resnik [Resnik, 1995], Wang [Wang et al., 2007], and Relevance [Schlicker et al., 2006] combined with the maximum strategy. The proposed pipeline is assessed using Autism Spectrum Disorder candidate genes.

Semantic similarity measures still play a crucial role in biomedical applications. Nonetheless, they face some limitations, namely the prerequisite that the two entities for which the similarity is being calculated need to be described in the same semantic space. However, biomedical applications often involve data from multiple sources, but integrating them into semantic similarity measures can be challenging.

3.3 End-to-End Approaches

In recent years, a popular avenue has been the integration of end-to-end approaches, which take KGs as input and produce predictions as output for several biomedical tasks, namely drug-drug interaction prediction. Lin et al. [2020] present KGNN, a end-to-end framework that explores KGs for relation prediction. KGNN encodes the features and its neighborhood structures between entity pairs in KG to predict the interaction value based on those encodings. Yu et al. [2021] address the challenge of effectively use biomedical KG for drug-drug interaction prediction. They present a new approach, SumGNN, that uses subgraph summarization in the KG around drug pairs to extract useful information. Then, a multi-channel neural encoding is used to make multi-typed drug-drug interaction predictions. Several KG embedding methods (e.g., DeepWalk, node2vec) and several GNN architectures (e.g., Graph Attention Network, Decagon) are used as baselines. Su et al. [2022b] propose a novel framework, KG2ECapsule, that exploits the type of drug association based on biomedical KGs in an end-to-end fashion. KG2ECapsule consists of four main components: (i) negative triplets sampling, which constructs the negative dataset by considering the likelihood of an entity appearing in either the head or tail position; (ii) graph-to-embedding layer, which propagates embeddings iteratively from an entity's neighbors and the relations connecting them; (iii) capsule layer, which specializes the entity representation under the given relational space and predict whether interactions occur between pairs of entities.

End-to-end approaches have also been proposed to predict associations between drugs and potential targets and diseases. Gao et al. [2018] propose an end-to-end NN model that uses chemical structures, amino acid sequences, and GO annotations to predict drug-target interactions.

The authors employ a long short-term memory recurrent NN and GCNN to learn proteins and drug representations. Then, a siamese network receives the two multi-layer networks as input and outputs the similarity between the input pair. Finally, the attention-based vector representations are used by a sigmoid function to make a prediction. Ge et al. [2021] proposes an integrative drug repositioning framework that builds a KG containing biomedical entities (drugs, human targets and virus targets) and their relations based on the known chemical structures, protein sequences and relations derived from publicly available databases. A deep learning-based method learns and updates the feature representation of each KG entity to predict the potential drug candidates against a specific coronavirus. Che et al. [2021] introduce Att-GCN-DDI, a GCNN-based model for predicting new drugs for diseases using a drug KG. The drug KG is built by integrating different knowledge bases and contains five types of entities (drugs, genes, diseases, channels and side effects). GCNN with an attention mechanism extracts features from the KG by employing matrix operations to reconstruct the prediction matrix. Gao et al. [2022a] present the KG-Predict framework for predicting new drug indications. The KG-Predict architecture includes two types of layers: a stack of CompGCN layers to learn vector representations of entities and relations; InteractE layers that concatenate the representations for each triple Drug-Treat-Disease and use a ranking function to generate higher scores for true triples and lower scores for false triples. Saadat et al. [2022] develop a novel approach based on GCNN for drug recommendation. The recommendation system module KGCNN receives the KG and the user-drug matrix as input. The biomedical KG also includes sentiment analysis extracted from public reviews. Krix et al. [2023] present MultiGML, which integrates biomedical KGs and other biological data sources into an end-to-end approach for predicting drug-related adverse events. In MultiGML, a multi-modal embedding layer is employed to generate a low-dimensional representation of entities that are the input for an encoder. Subsequently, a bilinear decoder is employed to compute a probability score indicating the likelihood of a relation between a drug and an adverse effect.

Prediction of disease or GDAs are other biomedical tasks that have been addressed. Bourgeais et al. [2021] present Deep GONet, a deep learning model that integrates GO for phenotype prediction (clinical diagnosis, prognosis, and drug response) based on gene expression profile of a patient. The hidden layers of Deep GONet represent the structure of GO since each hidden layer represents a GO level. Lan et al. [2022a] present KGANCDA that uses a graph attention NN to predict associations between circRNA and diseases. The KGs are constructed by integrating different kinds of biological association information, including circRNA, disease, lncRNA and miRNA. The attention network receives the KG as input and learns a representation for each entity by distinguishing the importance of information from neighbors. A multilayer perceptron is then employed on the representations of each pair of circRNA-disease associations, and the prediction scores are obtained. Lan et al. [2022b] present DRGCNCDA based on a transformer architecture for cirRNA-disease association prediction. The encoder is a GCNN and generates

the vector representations of circRNA and disease. The decoder is the DistMult factorization and is used as scoring function to predict potential circRNA-disease associations. Gao et al. [2022b] propose GenePredict-KG, the same methodology proposed in Gao et al. [2022a], but this time applied to GDA prediction.

Synthetic lethality prediction is directly related to GDA prediction. Synthetic lethality describes gene pairs where a mutation in either gene alone does not impact cell viability, but mutations in both genes together result in cell death. Wang et al. [2021c] propose KG4SL, a synthetic lethality prediction approach that uses a KG as input for a GNN model. The KG, designated as SynLethKG, includes several kinds of biomedical entities (e.g., diseases, genes, compounds, etc) and ontology classes (e.g., GO classes). The framework of KG4SL includes (i) generating a gene-specific weighted subgraph for each gene pair; (ii) incorporating an aggregation layer to update the representation of a specific gene by combining the representations of its neighbors in the weighted subgraph; (iii) computing a score for each pair using a normalized inner product derived from their learned representations. KG4SL is compared with several KG embedding methods (node2vec, deepWalk) and other GNN architectures (Graph Attention Network, GCNN, GraphSage). Zhu et al. [2023] use the same KG, but present a novel approach, SLGNN. SLGNN also uses GNNs to perform KG message aggregation and obtain a vector representation for each gene. The inner product of the representation for the gene pair is used as the probability of a synthetic lethality interaction. Wu [2023] also use a GNN to generate gene pair representations for predicting synthetic lethality. The difference is that a local subgraph is constructed and summarised before applying the GNN. The goal is to remove the entities that are not important to the target gene in the KG.

Prediction of single gene loss-of-function phenotypes and patient readmission risk prediction are other relevant biomedical tasks. Kulmanov and Hoehndorf [2020] propose DeepPheno that uses the GO annotations for gene products and the HP annotations for diseases to predict the phenotypes which result from the loss of function of a single gene. DeepPheno uses a fully connected NN followed by a hierarchical classification layer to encode the GO structure into the NN. DeepPheno is evaluated on the phenotype annotations from the HP. The results showed that the DeepPheno model performs best using experimental and electronically inferred GO annotations and gene expression values. The predicted phenotype annotations are also used to compute the similarity between genes and diseases and prioritize candidate genes to predict GDAs. Lu et al. [2021] present a GCNN-based approach for intensive care unit patient readmission risk prediction. The first step of this approach is constructing a multiview graph for clinical notes using external biomedical KGs. GCNNs are then used for representation learning.

In summary, deep learning architectures have been increasingly used in biomedical applications and perform well in various tasks. However, the predictive performance of GNN-based approaches is sensitive to their hyperparameters. Slight adjustments to these parameters can have considerable influence over the accuracy of predictions [Gonzales et al., 2022]. While a

common practice involves employing an exhausting brute-force parameter search to find the optimal parameters of GNN models, it is a time-consuming and computationally expensive approach. Furthermore, GNN still need to tackle the challenges of biomedical data complexity and interpretability.

3.4 Explainable Artificial Intelligence Approaches

There has been a growing emphasis on achieving explainability in biomedical tasks in recent years. Among the collection of 74 papers selected in Table 3.1, 15 propose approaches that are explainable. However, these approaches employ different strategies summarized in Table 3.2.

Gilvary et al. [2020] and Mukherjee et al. [2021] use semantic similarity-based semantic representations. Semantic similarity is explainable in its essence since it reflects the similarity of the entities according to the domain represented by the KG. In Gilvary et al. [2020], the drug similarity features are analysed to investigate the possible mechanisms behind the candidate drugs for each disease. In particular, the authors present two use cases: adrenergic uptake inhibitors applied to Parkinson disease and kinase inhibitors applied to diabetes. In Mukherjee et al. [2021], the features' importance of the DiGePred RF models is computed using the Gini impurity approach and the permutation approach.

Sosa et al. [2019], Hu et al. [2021a], Wang et al. [2021a], Zhang et al. [2021b], Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan [2022], Gavali et al. [2022], Ma et al. [2023b], Ma et al. [2023a], and Bang et al. [2023] have in common their usage of embeddings as semantic representations. Although very popular, embeddings are not explainable. Therefore, these approaches present post-hoc explainability approaches. Sosa et al. [2019] generate paths connecting drugs and disease pairs using three meta-paths (drug-disease-gene-disease, drug-disease-drug-disease and drug-gene-gene-disease). Then the distribution of path is analysed to identify specific patterns in the KG, which help explain how the model infers link predictions. The web server DLGLinker proposed in Hu et al. [2021a] includes a network visualization tool for graphical exploration of the associations between disease and genes and other biological factors in the KG, which played a role in their classification. Wang et al. [2021a] propose an explainable method for adverse drug reaction prediction by finding every path between the drug and the adverse effect. These paths are shown for the adverse drug reactions related to the drug osimertinib. Similarly, Ma et al. [2023a] employ a reinforcement learning model to identify the paths on the KG from drug nodes to disease nodes. The DrugMechDB obtains expert-verified paths as ground-truth data to evaluate explanations. Ma et al. [2023b] also explore paths between entities using a depth-first search method to generate organ failure prediction explications. In Zhang et al. [2021b], the discovery of patterns based on semantic relations explains why a particular drug is associated with a particular target. For this approach, a human expert is needed to sort out the noise in semantic relations. Using this discovery pattern approach, five promising drugs are

discovered. Gavali et al. [2022] quantify the changes in predictive performance when the KG is changed. This is accomplished through two types of experiments: one involves removing a specific set of triples while maintaining all other KG triples, and the other involves preserving triples related to a particular subset of the KG while eliminating all unrelated triples. Bang et al. [2023] use the teleport-guided random walks to explore the local neighborhood and other semantically relevant regions. The path analysis allows for an explanation of the biological mechanisms of drugs and diseases.

Bresso et al. [2021] adopt a different strategy to generate the representations that are then fed to a binary classifier for adverse drug reactions prediction. Three kinds of features are extracted from the KG: paths, path patterns, and neighbors. All the features are binary. For example, if a path p is found from the drug d, d is associated with the feature p in the output matrix. The features are constrained in order to avoid a combinatorial explosion of the number of features. Regarding the ML, transparent models based on DTs are employed. The explainability of the proposed approach is not only achieved by using interpretable features in the form of paths, path patterns or simple neighbors but also by employing transparent ML methods. A user study with experts showed that the most significant features are relevant for adverse drug reaction mechanisms.

Bourgeais et al. [2021], Yu et al. [2021], Krix et al. [2023], and Zhu et al. [2023] are end-to-end approaches and explainable. Deep GONet [Bourgeais et al., 2021] is a self-explainable deep NN that provides three levels of explanation: the disease level, the subtype of disease level, and the patient level. The model interpretation at the disease level involves clustering samples according to their neurons' activation. The model interpretation at the subtype of disease level and patient level refers to pointing out the main biological functions (GO classes) used for predictions and quantifying their contribution. SumGNN [Yu et al., 2021] provides model explainability by generating reasoning paths for each drug-drug interaction prediction. MultiGML [Krix et al., 2023] builds on the integrated gradients method that calculates the integrated gradients for each predicted link between a drug and a side effect to understand the importance of the individual features. The interpretability of the results obtained by SLGNN [Zhu et al., 2023] is achieved by highlighting the top KG relations and their weights for each gene pair prediction.

Several additional papers address the importance of explainability to some extent. However, they do not go into depth and demonstrate the explainability of the proposed approaches [Lei et al., 2020; Kulmanov and Hoehndorf, 2020; Fan et al., 2020; Zhang et al., 2021e; Wang et al., 2021c; Krämer et al., 2021; Chen et al., 2021b; Binder et al., 2022; Su et al., 2022b; Soman et al., 2023; Wu, 2023].

Reference	Type	Level of scope
Gilvary et al. [2020]	Transparent	Local
Mukherjee et al. [2021]	Post-hoc	Global
Sosa et al. [2019]	Post-hoc	Global
Hu et al. [2021a]	Post-hoc	Local
Wang et al. [2021a]	Post-hoc	Local
Zhang et al. [2021b]	Post-hoc	Local
Gavali et al. [2022]	Post-hoc	Global
Ma et al. [2023b]	Post-hoc	Local
Ma et al. [2023a]	Post-hoc	Local
Bang et al. [2023]	Post-hoc	Local
Bresso et al. [2021]	Transparent	Global
Bourgeais et al. [2021]	Transparent	Local
Yu et al. [2021]	Post-hoc	Local
Krix et al. [2023]	Post-hoc	Local
Zhu et al. [2023]	Post-hoc	Local

Table 3.2: Categorization of papers included in the literature review that employ explainable methods. Explainable approaches can be classified into two types: transparent models by design or post-hoc explainability techniques. Regarding the scope level, explanations can be categorized into local and global.

3.5 Limitations of the Related Work

This chapter provides a comprehensive overview of approaches that use ML and biomedical KGs as background knowledge to address a wide variety of biomedical tasks. The approaches employ KG embedding methods, semantic similarity measures, and GNN architectures, as detailed in previous sections. KG embeddings and semantic similarity are used to represent semantic representations that are then given as input to classical ML algorithms. KG embeddings are also used directly in the context of biomedical tasks cast as link prediction tasks. On the other hand, GNN architectures are integrated into end-to-end approaches, incorporating KGs as direct inputs. However, the discussion of the overall limitations of these approaches is missing. There are also some surveys in the literature that elucidate on the limitations associated with KG embeddings [Alshahrani et al., 2021; Su et al., 2018; Mohamed et al., 2021; Kulmanov et al., 2021] and GNN [Zhang et al., 2021d] in biomedical applications.

Su et al. [2018] and Mohamed et al. [2021] highlight as key limitations of KG embedding approaches the lack of interpretability, the inability to keep up the evolution of knowledge, the difficulty to use the semantics underlying ontologies and the hyperparameter sensitivity. While these concerns are associated with KG embeddings, they are equally applicable to GNN-based approaches. Efforts have been made to address the lack of explainability, with various approaches providing post-hoc techniques or using transparent models. However, there is still a long way to go, especially concerning the evaluation of the explanations. In the context of this Ph.D., it is also important to acknowledge a limitation regarding the utilization of the whole KG, ignoring the different KG semantic aspects. Although a few works do not use the entire KG, but rather a subgraph, the selection of the subgraph is always manual. Although this concern is particularly relevant to relation prediction tasks, it persists even in end-to-end approaches. Treating the KG as a whole may introduce noise, potentially impacting the accuracy of predictions.

Part II

Methodologies

Chapter 4

Explainable Similarity-based Semantic Representations for Relation Prediction

The potential of AI as a tool for scientific discovery in the biomedical domain has long been recognized, with ML, pattern mining and reasoning playing roles in several steps of the scientific process [Mjolsness and DeCoste, 2001]. One of AI's most promising and successful applications is its ability to predict PPIs. Proteins often interact with each other to carry out vital physiological functions. Through the integration of ML algorithms, AI predictive models have enabled researchers to identify potential protein interactions with implications for drug discovery and personalized medicine [Zhang and Tang, 2016; Chen et al., 2019; Zhang et al., 2021e; Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan, 2022].

Although AI plays an important role in the scientific process, its application in science requires explainability, a fundamental piece to transform AI into a scientific tool that is able to uncover new knowledge, to understand the mechanisms that underlie the natural phenomena that are being predicted and to distinguish between meaningful predictions and spurious correlations [Barredo Arrieta et al., 2020]. However, the vast majority of scientific projects that employ AI are not concerned with explainability [Roscher et al., 2020]. In the biomedical domain, both the complexity of the data and the natural phenomena under study highlight the necessity of domain knowledge to support explainability [Holzinger et al., 2017]. Explainable AI is gaining traction as a potential solution to ensure that algorithms and their predictions can be human-understandable. A knowledge-enabled explainable AI system includes a representation of the domain knowledge in the field of application, which is explored to generate

user-comprehensible and context-aware explanations of the mechanistic functioning of the AI system and the knowledge used [Chari et al., 2020].

Biomedical ontologies express knowledge about a domain and allow the description of complex biological phenomena that are not easily captured in mathematical form [Staab and Studer, 2010]. As such, they provide the scaffolding for comparing biological entities at a higher level of complexity by comparing the ontology classes with which they are annotated. Measuring the semantic similarity between biomedical entities through their ontology annotations has become a cornerstone bioinformatics application in PPI prediction [Zhang and Tang, 2016; Chen et al., 2019; Wang et al., 2019c], GDA identification [Hoehndorf et al., 2011; Asif et al., 2018; Mukherjee et al., 2021], and drug-drug interaction prediction [Abdelaziz et al., 2017; Kastrin et al., 2018; Lee et al., 2019]. Semantic similarity has been combined with ML approaches in different supervised and unsupervised learning tasks, but in recent years, a spate of novel KG embeddings and deep learning-based approaches have been employed over the same tasks with success [Kulmanov et al., 2021; Chen et al., 2019; Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan, 2022]. However, in some applications, classical semantic similarity measures still outperform KG embeddings [Sousa et al., 2021]. One advantage of employing semantic similarity-based features over KG embeddings is that similarity assessment is a natural explanatory mechanism [Wang et al., 2019a], whereas KG embeddings are opaque vectorial representations [Palmonari and Minervini, 2020].

Although a semantic similarity score as a feature is explainable in its essence, since it reflects the similarity of the entities according to the domain represented by the ontology, it is a very compact explanation, often reduced to a single numerical score or at most to one score per ontology root of a general-purpose KG. This similarity score tells us if two proteins are similar but not why they are similar. For instance, in GO, it is common to measure the semantic similarity of annotated gene products according to its three branches: biological process, cellular component and molecular function. However, providing a very general explanation, such as the fact that both proteins have a higher biological process similarity, would not enhance the ML model's reliability or elucidate the biological phenomena. The hypothesis is that by measuring the semantic similarity between entities targeting specific subgraphs of the ontology, it is possible to increase the explainability of semantic similarity-based features without substantial performance sacrifices. These subgraphs capture different *semantic aspects*, i.e., different perspectives of the representation of ontology-annotated entities.

In this chapter, two key challenges are tackled. Firstly, the lack of explainability in well-known KG-based representations, such as KG embeddings. Secondly, the general-purpose nature of KGs that leads to KG-based representations of entities lacking meaningful interpretations. This chapter presents KGsim2vec, a novel method to generate explainable vector representations by representing entity pairs in a KG through aspect-oriented semantic similarity features. This technique explores the rich semantics of the ontology to identify the semantic aspects and com-

pute the similarity for each aspect. Then the similarities are given as features for an ML model for relation prediction. Given a prediction, KGsim2vec explains it by using either the ML model or post-hoc techniques. Additionally, a novel approach is proposed to evaluate explanation quality, which combines the number of features in an explanation with their informativeness as measured in the KG. KGsim2vec is evaluated in PPI prediction using the GO KG. Since biomedical data resources share, reuse and import data from each other routinely, there is a high potential for data leakage in biomedical ML applications. Therefore, the potential data leakage in the prediction of PPIs is investigated by comparing the performance of models trained and tested on the same versions of data versus training on archived data and predicting only for newly discovered protein interactions.

Within the scope of the thesis and its RQs, this chapter explores semantic similarity-based semantic representations (RQ1) and establishes a definition of class-based semantic aspects as subgraphs of the ontology at the same depth (RQ2). The assessment of the proposed semantic representation's impact on performance and explainability is conducted in the context of PPI prediction (RQ3). Additionally, the evaluation includes a comparative analysis of these representations against other KG embedding methods and GNNs (RQ1). The experiments reveal that the proposed semantic representation improves explainability by producing global models that capture biological phenomena and elucidate data biases.

Main contributions of the chapter:

- KGsim2vec, a novel method for generating explainable representations by representing entity pairs in a KG through semantic aspect-oriented semantic similarity features. The semantic aspects are defined subgraphs of the ontology at a specified depth provided as input.

- A novel approach for evaluating the quality of explanations, which considers both their size and informativeness.

- An evaluation framework for PPI prediction, which demonstrate the effectiveness of KGsim2vec in producing useful explanations for relation prediction.

- An investigation of potential data leakage in the PPI prediction using the GO KG by comparing the performance of models trained and tested on the same versions of data versus training on archived data and predicting only for newly discovered protein interactions.

- The code is available at `https://github.com/liseda-lab/ExplainablePPI`.

Papers supporting the chapter:

- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2024). Explaining protein–protein interactions with knowledge graph-based semantic similarity. Computers in Biology and Medicine, 170, 108076.*[1](Appendix A)
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2021). evoKGsim+: a framework for tailoring knowledge graph-based similarity for supervised learning. In Proceedings of Extended Semantic Web Conference - Poster and Demo Track, pages 141–146. Springer.*[2] (Appendix G)
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2021). Is there data leakage in protein-protein interaction prediction using knowledge graphs? In Proceedings of International Semantic Web Conference - Poster Demo Track.*[2] (Appendix F)
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2022). Explaining protein-protein interaction predictions with genetic programming. In EvoStar Late-breaking abstracts, page 30.*[2] (Appendix E)

4.1 Problem Formulation

The focus of KGsim2vec is on ontology-rich KGs (Definition 1), where ontologies are used to describe individual instances, while the instances themselves are usually flat with no connections between them. As a result, these KGs have two distinct types of nodes: nodes corresponding to individual entities and nodes corresponding to ontology classes. These KGs also contain two types of edges: one type that connects ontology classes to each other and another type that links individual entities to the classes that describe them. For example, in the GO KG, an edge between a protein (individual entity) and a GO class (ontology class) indicates that a particular protein P performs a specific function F described in the GO. At the same time, an edge between two GO classes, $F1$ and $F2$ can represent the fact that one GO class is a subclass of the other one. KG semantic similarity measures compute the similarity between two entities by comparing the ontology classes which are connected to each entity considering the structure of the ontology itself.

The goal is to learn a relation between two KG entities when the relation itself is not explicitly defined in the KG. To tackle this prediction task, recent approaches employ KG embedding methods. These methods generate vector representations for each entity, which are then combined to be used as input features for ML methods [Kulmanov et al., 2021; Chen et al., 2019;

[1]This chapter reproduces the methodology and results presented in this paper.
[2]This chapter provides a summarized version of this paper.

Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan, 2022]. However, these vector representations are non-explainable since each dimension does not represent any specific meaning. Furthermore, these approaches rely on creating a representation of each entity using the whole KG, ignoring the different semantic aspects of the KG. A semantic aspect is a perspective of the KG entities, and it can be represented as a subgraph (Definition 6). For example, the three branches of the GO (biological process, cellular component and molecular function) can represent three semantic aspects. Entities are characterized according to various semantic aspects, but only a few of them might be relevant for predicting a particular relationship. A prior investigation [Sousa et al., 2020] demonstrated that not all branches of the GO are equally important for PPI prediction.

4.2 KGsim2vec

KGsim2vec is a novel method to generate explainable vector representations of entity pairs in a KG to support learning with minimal losses in performance when compared to opaque models. The proposed method computes the explainable vector representations, then applies ML algorithms to generate predictive models, and finally generates explanations (represented in Figure 4.1).

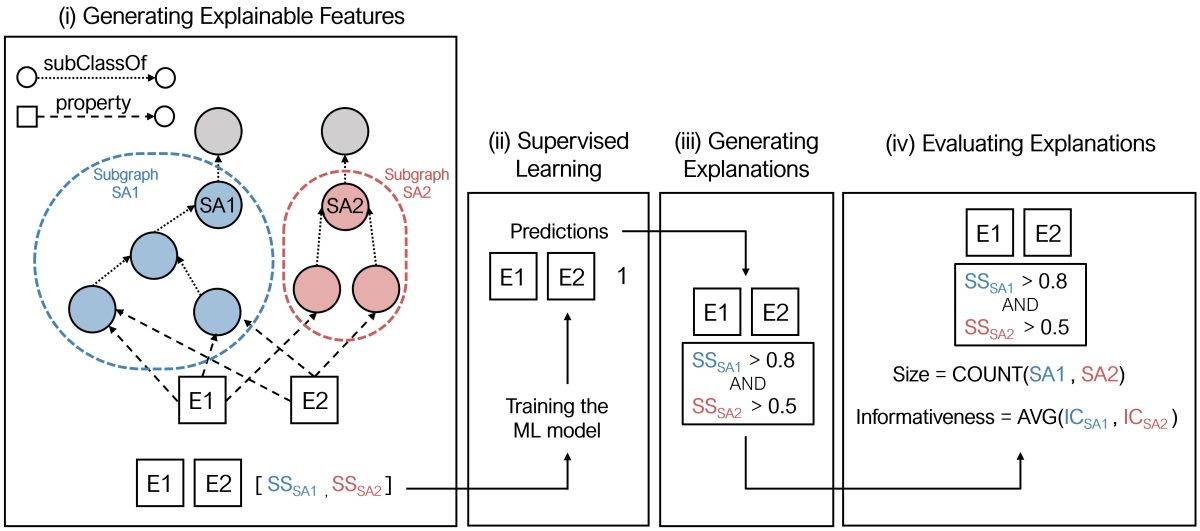


Figure 4.1: Overview of KGsim2vec with the main steps: (i) generating explainable features (ii) supervised learning (iii) generating explanations (iv) evaluating explanations.

The first step is generating explainable features. To do that, the KG is transformed into an RDF graph, which facilitates the subsequent processing. Then, the proposed approach extracts

the KG semantic aspects to compute the semantic similarity and generate a pair representation. The second step is concerned with employing supervised learning methods to learn a relation prediction model taking as input the pair representation. The last steps correspond to generating and evaluating explanations for the predictions.

4.2.1 Generating Explainable Features

The proposed novel method, KGsim2vec, generates an explainable vector representation of entity pairs in a KG described according to the same ontology. The representation is based on the semantic similarity between the entities according to different semantic aspects of the ontology, i.e., subgraphs of the ontology at the same depth (Algorithm 1).

Algorithm 1 KGsim2vec algorithm. MICA stands for most informative common ancestor.

$\alpha \leftarrow minimum\_feature\_number$
$\beta \leftarrow minimum\_height$
$\gamma \leftarrow minimum\_coverage$
function GET SA($classes$)
    if $\beta > 0$ **then**
        $classes \leftarrow$ FILTER CLASSES BY HEIGHT($classes$)
    if $\gamma > 0$ **then**
        $classes \leftarrow$ FILTER CLASSES BY COVERAGE($classes$)
    if $len(classes) \geq \alpha$ **then**
        return $classes$
    else
        $new\_classes \leftarrow \emptyset$
        for c in $classes$ **do**
            $new\_classes$.append(GET SUBCLASSES(c))
        return GET SA($new\_classes$)

function GET SS SCORE($entity_1$,$entity_2$,$subgraph$)
    $a_1 \leftarrow$ GET ANNOTATIONS($entity_1, subgraph$)
    $a_2 \leftarrow$ GET ANNOTATIONS($entity_2, subgraph$)
    return max(GET IC(GET MICA(c_1, c_2)): $c_1 \in a_1, c_2 \in a_2$

function GET EXPLAINABLE VECTORS($entity\_pairs, ontology$)
    $vectors \leftarrow \emptyset$ ▷ dictionary to hold explainable vectors for each entity pair
    $root \leftarrow$ GET ROOT($ontology$)
    $semantic\_aspects \leftarrow$ GET SA($root$)
    for s in $semantic\_aspects$ **do**
        $sg \leftarrow$ GET SUBGRAPH(S) ▷ extracts an ontology subgraph rooted in s
        for e_1, e_2 in $entity\_pairs$ **do**
            $vectors[e_1, e_2]$.append(GET SS SCORE(e_1,e_2,sg))
    return $vectors$

To extract the semantic aspects, a breadth-first search on the ontology graph is performed to find the depth (i.e. distance to the root(s)) at which the number of classes is greater than α and retrieve those classes. This parameter can be set to manipulate the size and, consequently, the level of detail afforded by the explainable vectors. Other criteria can also be explored to filter the subgraphs. In addition to depth, a minimum height (β) – i.e., distance to a leaf class – is used to remove subgraphs of insufficient depth, as well as a minimum coverage (γ) – i.e., percentage of entities annotated in the semantic aspects – to remove subgraphs that are seldom used to describe the entities.

Once the semantic aspects have been identified, the semantic similarity between each pair of entities are computed according to each semantic aspect. KGsim2vec employs the maximum pairwise similarity between all classes that annotate each entity. To measure class similarity, KGsim2vec computed the IC of the most informative common ancestor between the classes. KGsim2vec employs IC_{Seco} [Seco et al., 2004], a structure-based approach based on the number of direct and indirect descendants.

4.2.2 Supervised Learning

After obtaining the vector representations, ML algorithms are used to learn relation prediction models. Representative tree-based ML algorithms are used: two interpretable models, DTs and GP, and two opaque models, RF and XGB.

DTs [Denison et al., 1998] meet the characteristics of transparent models (algorithmic transparency, decomposability and simulatability) and are a familiar representation [Barredo Arrieta et al., 2020]. GP [Koza, 1992] is an evolutionary computation technique inspired by Darwinian natural selection and Mendelian genetics and can return interpretable models by combining features, operators and numerical values [Mei et al., 2022]. However, the size of the models can also influence interpretability. While learning over complex data, DTs and GP models may grow very large, increasing the cognitive effort required to interpret the solutions. To tackle these challenges, models that take into consideration their depth during training are generated:

- DT6 where the maximum depth that trees are allowed to reach is 6;

- GP6x with a fitness function that penalizes models with a depth greater than 6, and using only interpretable operators (i.e., maximum, minimum, addition and subtraction, since operators such as multiplication and division have a less straightforward meaning for interpretability).

Regarding RF and XGB, they are ensemble models that combine the decisions from multiple DTs. They are classified as opaque models and require post-hoc techniques.

4.2.3 Generating Explanations

For interpretable models, the explanation is the model itself. For example, a DT is constructed by beginning with the root node that contains the whole learning sample and then splitting a node into two child nodes repeatedly. Each DTs can be converted into a set of decision rules with the form: IF condition 1 AND condition 2 AND condition 3 AND ... THEN outcome, where the number of conditions is the number of decision nodes from root to leaf. The GP models can also be converted to a mathematical formula easily interpretable by reading their trees depth-first.

However, for the opaque models, this is not possible. Therefore, a surrogate model is added to produce local models to explain individual predictions. Two of the most well-known post-hoc explainability methods are used:Local Interpretable Model-Agnostic Explanations (LIME) [Ribeiro et al., 2016] and Local Rule-Based Explanations (LORE) [Guidotti et al., 2018a].

LIME can explain the predictions of any classifier or regressor by approximating it with an interpretable model that is locally faithful to the ML model. This algorithm starts by randomly generating instances in the neighborhood of the instances to be explained, weighs them by the proximity to the instance, and then infers linear models as comprehensible local predictors. The LIME output is a list of weighted features in the format `(feature f > x, y)`, where `feature f > x` frames the value of feature `f` for that data sample and `y` is the contribution of feature `f` to the prediction of a data sample. The number of features that appear in the LIME explanation is given as a parameter. In the experiments, two values for this parameter are tested: 3 (LIME 3feat) and 8 (LIME 8feat).

Another similar approach to LIME is LORE. This method first generates a synthetic neighborhood on which it learns a local interpretable predictor. A local explanation is then extracted. The LORE output is more straightforward since it is a single decision rule which characterizes the conditions concerning the features' values for the decision of the opaque model. These rules are built by generating a set of neighbors of the data sample through a genetic algorithm and then extracting from such a set a DT.

4.2.4 Evaluating Explanations

To evaluate the explanations, two aspects are considered: size and informativeness. Since each feature represents the similarity for a specific ontology semantic aspect rooted in a specific ontology class, the specificity of each feature is measured according to the IC of the class (IC_{Seco}). The higher the IC value, the more informative this feature will be. The informativeness of an explanation is the average of the IC of the explanation features. A good explanation would then be composed of a few features to be easily understood (cognitive studies indicate humans are able to hold 7 ± 2 objects in short-term memory [Miller, 1956]), but those few features should be as informative as possible.

4.3 Results and Discussion

KGsim2vec targets relation prediction tasks cast as a classification task that takes as input entity pairs and a KG back-boned by an ontology. The ontology is structured as a directed acyclic graph, where each class is linked to its ancestor through subclass relations. As a result, each class is more specific than its ancestors. Furthermore, these relations are transitive, indicating that they inherit all the ancestor classes up to the root. KGsim2vec is evaluated on PPI prediction using the GO KG. The data used are described in the following sections.

4.3.1 Data

The understanding of biological processes relies on the study of PPI. However, experimental detection of PPIs is time-consuming and laborious. Despite proteins being annotated with GO (either for experimental evidence or automatically generated), only a limited number of protein interaction sites have been experimentally validated in current databases.

The target relations to predict are obtained from STRING [Szklarczyk et al., 2021]. This database is one of the largest available PPI databases that integrates physical interactions and functional associations between proteins collected from several sources. All interaction evidence is benchmarked and scored to estimate the confidence on whether a proposed association is biologically meaningful given all the contributing evidence. The following criteria are considered to select protein pairs:

- each protein must be annotated with the GO;
- protein interactions must be extracted from curated databases or experimentally determined (as opposed to computationally determined);
- interactions must have a confidence score above 0.950 to retain only high confidence interaction.

The PPI dataset contains 23571 interacting protein pairs and an equal number of negative pairs that have been generated through random negative sampling from the same pool of proteins.

The GO KG describes proteins and is built by integrating the GO [Consortium, 2021] and protein annotation data [Huntley et al., 2015] (see section 2.1.1 for more details about GO KG). The GO KG allows measuring the similarity between gene products by comparing the set of concepts they are annotated with. To avoid the potential for data circularity, the *"protein-containing_complex"* branch and the corresponding GO annotations are removed from the GO KG.

Data Leakage in Protein-Protein Interaction Prediction using Gene Ontology Knowledge Graph

In biomedical applications, such as PPI, data leakage can also be an issue since it is not uncommon for multiple databases and resources to reuse the same sources of information. Leakage occurs when information about the target of a data mining problem that should not be legitimately available to mine from is introduced [Kaufman et al., 2012], and it can lead to an overestimation of the model's performance. The GO KG, composed of the GO [Consortium, 2021] and GO annotations [Huntley et al., 2015] that link proteins to GO classes, is continuously evolving as more data become available [Tomczak et al., 2018]. The majority of GO annotations are inferred from electronic annotations, which means they are based on the automated processing of other data sources. This could result in the same information that is used to support a PPI in a database (e.g. STRING [Szklarczyk et al., 2021]) to also be used to establish a GO annotation for the proteins. Therefore, the potential data leakage is investigated by comparing the performance of models trained on archived data and predicting exclusively for recently found protein interactions with models trained and tested on the same versions of data.

The hypothesis is that if this type of data leakage is typical, then the performance of GO-based PPI prediction methods would be artificially increased. To test this hypothesis, PPI prediction models trained on older GO data and PPI interactions and tested on previously unknown interactions captured in more recent versions of STRING are compared with models trained and tested on the same version. Furthermore, training the models on labeled examples from the past simulates more closely real-world applications.

The first step is using historical data to build the PPI datasets. Several PPI datasets are built using three archived versions of the STRING database (v9.1, v10, and v10.5) and the current version (v11). For the current version, three datasets are created, each excluding protein pairs present in each of the older versions (see Table 4.1). Then the GO KG and the protein pairs are used to predict interactions using several ML algorithms. Regarding the GO KG, archived versions of the GO and GO annotations in 2015, 2017 and 2019 from the Gene Ontology Data Archive[3] are obtained.

Two types of experiments are conducted: (i) ***Same version***, where the model is trained with randomly chosen 10000 protein interacting pairs from the archived STRING version and tested with the remaining pairs; (ii) ***Future version***, where the model is trained with randomly chosen 10000 protein pairs from the archived STRING version and tested on data from the current STRING version (excluding interactions present in the archived version). The same randomly chosen 10000 protein pairs are used in both settings.

Since three archived versions are used, the *Future version* experiments also allow measuring the impact of using increasingly older versions of STRING and GO in training. The results

[3]http://release.geneontology.org/

Table 4.1: Number of positive pairs in each version of the STRING database.

STRING Version	**Date**	**Number of positive pairs**
v9.1	04/2015	12 681
v10	05/2017	26 863
v10.5	01/2019	31 384
v11 (excluding pairs in v9.1)	10/2020	41 227
v11 (excluding pairs in v10)	10/2020	31 642
v11 (excluding pairs in v10.5)	10/2020	23 571

do not support a clear indication of data bias. While for the 2019 version, it is always slightly easier to predict future PPIs, this is reversed in the 2017 version, and varies between methods for the 2015 version, so no clear trend is discernible. The weighted average F-measure for the *Same version* experiments is 0.844, while it is 0.845 for the *Future version* (see Figure 4.2). In addition to not detecting data leakage, the results also indicate that the relation between the functions of a protein and its interactions does not fundamentally change over time. Even for more recently discovered interactions that can be biologically different, protein functions are still a good predictor of PPIs.

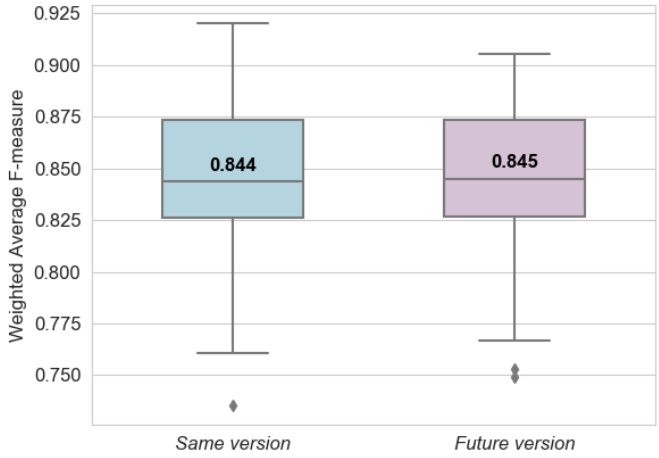


Figure 4.2: Weighted average F-measure boxplot using the *Same version* and the *Future version* to test.

4.3.2 Preliminary Results

evoKGsim+ precedes KGsim2vec by computing the semantic similarity for the different KG semantic aspects defined as subgraphs rooted in classes. However, the semantic aspects correspond to subgraphs when there are multiple roots or the subgraphs at a distance of one of the root when a single root exists. KGsim2vec is a generalization of evoKGsim+, where the depth of the classes that root the subgraphs is not fixed but rather dependent on several parameters given as input.

The evoKGsim+ framework[4] is able to: (1) compute semantic similarity-based representations of KG individuals according to different semantic aspects and using different similarity approaches; (2) employ GP to learn a suitable representation targeted to a supervised learning task by combining the different semantic aspects; and (3) evaluate the outcome of (2) against a set of static representations emulating experts. This framework is independent of the specific implementation of KG-based similarity and the GP parameters employed to evolve the representations. evoKGsim+ supports 10 different KG-based similarity measures: 6 taxonomic similarity measures, derived by combining one of two information content approaches (IC_{Seco} and IC_{Resnik}) with one of three set similarity measures ($Resnik_{Max}$, $Resnik_{BMA}$, and SimGIC [Pesquita et al., 2008]); 4 measures based on cosine similarity over embeddings generated from TransE [Bordes et al., 2013], distMult [Yang et al., 2015], RDF2Vec [Ristoski and Paulheim, 2016a] and Owl2Vec [Chen et al., 2021a].

A key aspect of the evaluation approach is to compare evoKGsim+, that is able to evolve a combination of semantic aspects, to static combinations established *a priori*. This allows comparing the proposed methodology to a scenario where semantic aspects are selected and combined by experts before the prediction task. Five static combinations are used as baselines: biological process, molecular function, and cellular component single aspects, and the average and maximum of the single aspect scores. To establish the performance of the static baselines, the prediction of PPI is formulated as a classification problem where a semantic similarity score for a protein pair exceeding a certain threshold (semantic similarity cutoff) indicates a positive interaction. The semantic similarity threshold is chosen after evaluating the weighted average F-measure at different threshold intervals and selecting the maximum. This emulates the best choice that a human expert could theoretically select.

Table 4.2 presents the results obtained using different similarity-based semantic representations for the baselines and evoKGsim+. For evaluating the quality of a predicted classification, the weighted average F-measure is used for stratified 10-fold cross-validation.

evoKGsim+ with taxonomic similarity always achieves the best performance compared to the baseline semantic representations. Regarding the KG embedding approaches, TransE has performed worse than the other embedding methods. These differences are not unexpected since

[4]https://github.com/liseda-lab/evoKGsim

Table 4.2: Median of weighted F-measure for the baselines (biological function, cellular component, molecular function, Average, and Maximum) and evoKGsim+ 10-fold cross-validation. BP stands for biological function, CC stands for cellular component, and MF stands for molecular function.

Similarity Measure	**Baselines**					**evoKGsim+**
	BP	CC	MF	Avg	Max	
ResnikMax + IC_{Seco}	0.760	0.713	0.646	0.749	0.743	**0.765**
ResnikMax + IC_{Resnik}	0.750	0.717	0.653	0.766	0.774	**0.776**
ResnikBMA + IC_{Seco}	0.753	0.715	0.643	0.771	0.744	**0.777**
ResnikBMA + IC_{Resnik}	0.753	0.714	0.648	0.777	0.772	**0.782**
SimGIC + IC_{Seco}	0.736	0.682	0.642	0.729	0.701	**0.746**
SimGIC + IC_{Resnik}	0.739	0.704	0.651	0.750	0.734	**0.758**
TransE	0.501	**0.534**	0.502	0.519	0.521	0.521
distMult	0.704	0.599	0.498	0.670	0.668	**0.712**
RDF2Vec	0.675	0.654	0.631	0.684	0.668	**0.685**
Owl2vec	0.678	0.662	0.621	**0.693**	0.686	**0.693**

the goal is to learn which aspects of a KG are more relevant to the learning task, and most of the information to be processed is represented in the ontology portion of the KG, where taxonomic relations play an important role. Therefore, translational distance approaches that emphasize local neighbourhoods are less suitable than semantic matching methods, like disMult, or methods that capture longer-distance relations, such as path-based approaches (RDF2Vec and Owl2Vec).

When comparing the two types of semantic similarity, evoKGsim+ with taxonomic similarity achieves a better performance than evoKGsim with embedding similarity. Although embeddings consider all types of relations, the hypothesis is that taxonomic similarity can take into account class specificity that may give it the advantage over embedding similarity in more accurately estimating similarity.

4.3.3 Performance Evaluation

The predictive performance of KGsim2vec is evaluated against popular KG embedding approaches and GNN. The KG embeddings were generated using the whole KG and three KG embedding approaches: RDF2Vec [Ristoski and Paulheim, 2016a], OWL2Vec* [Chen et al., 2021a] and GO2vec [Zhong et al., 2019] (an application of node2vec to the GO). Then, the vectors representing each protein are combined using the Hadamard operator and given as input to the ML approaches. For the GNNs, the framework proposed in Lin et al. [2020] is used. This framework encodes each protein's semantic features based on their neighbourhood, and the in-

teraction is predicted based on the learned embeddings. Each model was evaluated using 10-fold cross-validation and, for each fold, the weighted average F-measure was computed. This metric accounts for class imbalance by computing the F-measure for each class and then calculating the average of all computed F-measures, weighted by the number of instances of each class.

Table 4.3 reports the median weighted average F-measure and the interquartile range of the 10 weighted average F-measure values for different ML algorithms using KGsim2vec explanainable representation method or the KG embeddings. The proposed method was applied with a straightforward set of parameters ($\alpha = 10$, $\beta = 0$ and $\gamma = 0$). Statistically significant differences are determined using pairwise non-parametric Kruskal-Wallis tests at $p < 0.01$.

Table 4.3: Weighted average F-measure medians (M) and interquartile range (IQR) using KGsim2vec or the embeddings coupled with different ML approaches (RF, XGB, DT, DT6, GP, GP6x), as well as a GNN. The best result for each ML approach is in bold. KGsim2vec performance values are italicized/underlined when improvements are statistically significant with p-value < 0.01 for the Kruskal-Wallis test against the other methods.

	RF		**XGB**		**DT**		**DT6**		**GP**		**GP6x**	
	M	IQR	M	IQR	M	IQR	M	IQR	M	IQR	M	IQR
KGsim2vec	***0.919***	0.005	0.915	0.004	***0.899***	0.003	***0.906***	0.002	***0.866***	0.005	***0.866***	0.006
RDF2Vec	0.904	0.004	**0.917**	0.009	0.783	0.007	0.747	0.006	0.756	0.005	0.776	0.021
OWL2Vec*	0.861	0.002	0.873	0.004	0.710	0.011	0.683	0.007	0.656	0.035	0.693	0.021
GO2Vec	0.881	0.007	0.904	0.002	0.715	0.005	0.687	0.016	0.728	0.016	0.749	0.011

GNN	
M	IQR
0.815	0.007

The results in Table 4.3 show that KGsim2vec outperforms RDF2Vec when combined with all ML methods except XGB. These differences are statistically significant.

4.3.4 Explanations Evaluation

KG embeddings are, of course, non-explainable, since each feature does not have a particular meaning. As such, this evaluation focuses on comparing the explanations generated by applying the proposed method both with interpretable ML approaches (DT6 and GP6x) and surrogate methods over non-interpretable ones (LIME and LORE over RF and XGB).

Figure 4.3 shows that the explanations generated by DT6 and LORE are the smallest and least informative, whereas GP6x finds a compromise between the number of features and their

informativeness. LIME's performance is dependent on defining the number of features to consider: when using fewer features, informativeness drops below that obtained by GP6x, and approximates DT6 when the same explanation size is considered, but for a similar size to GP6X (8 features), its explanations are the most informative.

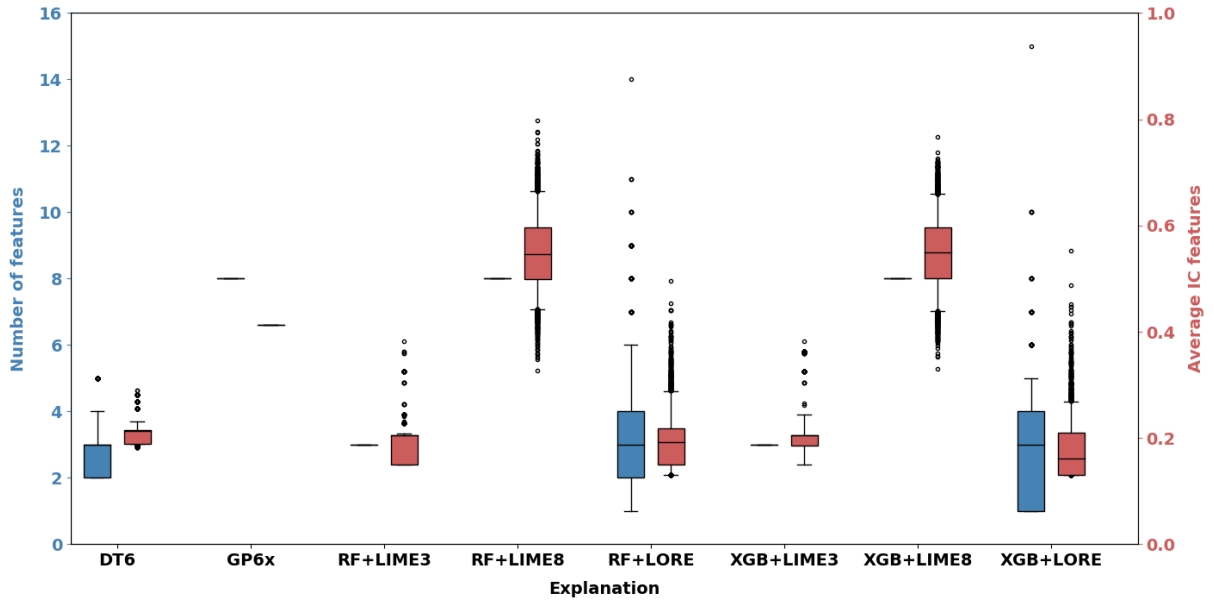


Figure 4.3: Size and informativeness of the explanations obtained for the first partition samples.

4.3.5 Explanations by Example

Explanations by example consider the extraction of representative data examples related to the result generated by a specific model, enabling a better understanding of the model itself.

Tables 4.4 to 4.7 present explanations for four protein pairs chosen randomly from the PPI dataset representing each of the four possible outcomes: a true positive, a false positive, a true negative, and a false negative. The tables present a chart with semantic similarity measured for the most relevant semantic aspects, a short description of the interaction status and the generated explanations with their size and informativeness. Since in the four selected examples, XGB and RF agreed on the predictions resulting in equivalent explanations, only RF is shown. However, two runs were performed for LORE since its explanations for the same instance and ML model can vary between runs due to the stochastic neighbor generating strategy it employs.

40S Ribosomal Protein S12 and 40S Ribosomal Protein S10

40S ribosomal protein S12[5] and 40S ribosomal protein S10[6] make up the first pair (Table 4.4). The eukaryotic small ribosomal subunits (40S) play a central role in protein translation since they contain the decoding centre where mRNA codons are recognized by complementary anticodons of tRNAs bearing amino acid residues for protein synthesis. Multiple binding sites characterize this subunit.

These proteins have 12 direct annotations in common, namely two specific biological process classes: *"nuclear-transcribed_mRNA _catabolic_process, nonsense-mediated_decay"* class (defined as a nonsense-mediated decay pathway that prevents the translation of mRNAs into potentially harmful proteins); *"SRP-dependent_cotranslational_ protein_targeting_to_ membrane"* class (SRP is a cytosolic particle that transiently binds to the endoplasmic reticulum and it is essential for the targeting of proteins to a membrane in translation). For all analyzed models, the high similarity computed for the *"metabolic_process"* and *"cellular_process"* semantic aspects always appears in the explanations. These explanations are in agreement with the fact that the two proteins participate in the same metabolic processes.

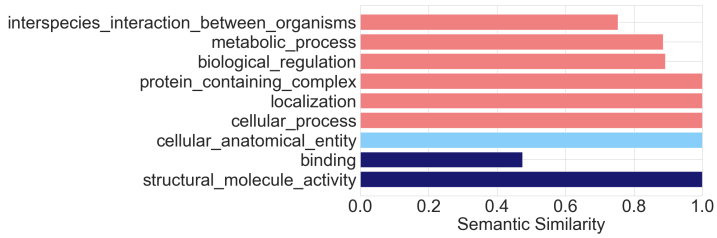
Neuroblast Differentiation-associated Protein AHNAK and Protein S100-A10

The neuroblast differentiation-associated protein AHNAK[7] and the protein S100-A10[8] constitute the second pair (Table 4.5). Neuroblast differentiation-associated protein AHNAK is a sizeable structural scaffold protein that may play a role in such diverse processes as blood-brain barrier formation, cell structure and migration, cardiac calcium channel regulation, and tumour metastasis. Protein S100-A10 is an integral part of cellular structural scaffolding that interacts with plasma membrane proteins through its association with annexin II.

The protein pair share four direct annotations since they both have the same function (*"protein_binding"*) and are localized in the same cellular components, namely cytoplasm, extracellular exosome and membrane raft. Although the proteins share some semantic annotations, they are very general. Therefore, all the analyzed ML methods fail to predict this protein pair interaction. However, according to the literature, they are likely involved in the mediated organization of the actin cytoskeleton [Hayes et al., 2004]. Both proteins are poorly described under the GO, which may explain why the ML models fail.

[5]https://www.uniprot.org/uniprot/P25398
[6]https://www.uniprot.org/uniprot/P46783
[7]https://www.uniprot.org/uniprot/Q09666
[8]https://www.uniprot.org/uniprot/P60903

Table 4.4: Explanations of ML models for the 40S ribosomal protein S12 – 40S ribosomal protein S10 positive pair.

40S ribosomal protein S12 – 40S ribosomal protein S10			
			40S ribosomal protein S12 and 40S ribosomal protein S10 are components of the 40S ribosomal subunit that plays a central role in protein translation and is characterized by multiple binding sites.
Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (+)	$SS_{\text{metabolic\_process}} > 0.5167$ AND $SS_{\text{cellular\_process}} > 0.8032$	2	0.161
GP6x (+)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}) \geq 0.5$	9	0.423
LIME 3feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4694), (SS_{\text{metabolic\_process}} > 0.61, 0.2611), (SS_{\text{biological\_regulation}} > 0.57, 0.1752)$	3	0.151
LIME 8feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4800), (SS_{\text{metabolic\_process}} > 0.61, 0.2656), (SS_{\text{pigmentation}} \leq 0, -0.2017), (SS_{\text{biological\_regulation}} > 0.57, 0.1929), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1490), (SS_{\text{protein\_folding\_chaperone}} \leq 0, -0.1440), (SS_{\text{molecular\_carrier\_activity}} \leq 0, 0.1256), (SS_{\text{multi-organism\_process}} \leq 0, -0.1233)$	8	0.466
LORE 1 (+)	$SS_{\text{biological\_regulation}} > -0.1206$ AND $SS_{\text{response\_to\_stimulus}} \leq 0.1863$ AND $SS_{\text{biological\_adhesion}} \leq 0.0056$ AND $SS_{\text{cellular\_process}} > 0.5715$	4	0.278
LORE 2 (+)	$SS_{\text{metabolic\_process}} > 0.7834$	1	0.190

Proline Protein and Guanine Nucleotide-binding Protein

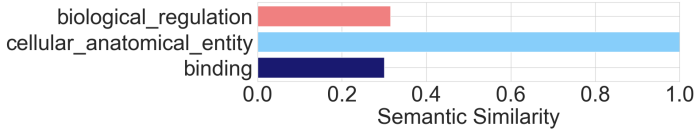
The third example pair is composed of proline protein[9] and a guanine nucleotide-binding protein[10] (Table 4.6). The proline-rich protein 5-like associates with the mTORC2 complex that regulates cellular processes, including survival and organization of the cytoskeleton. The guanine nucleotide-binding protein-like 3 is a GTPase binding nuclear protein that has been reported to be involved in various biological processes, including cell proliferation, cellular senescence and tumorigenesis.

Although the two proteins have annotations for the three GO domains, they have only one direct annotation in common: the molecular function class *"protein_binding"*. Furthermore,

[9]https://www.uniprot.org/uniprot/Q6MZQ0
[10]https://www.uniprot.org/uniprot/Q9NVN8

Table 4.5: Explanations of ML models for the S100-A10 – neuroblast differentiation-associated protein positive pair.

S100-A10 protein – neuroblast differentiation-associated protein



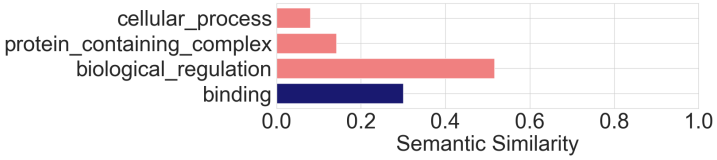
Protein S100-A10 is an integral part of cellular structural scaffolding that works together with neuroblast differentiation-associated protein AHNAK, a membrane-associated protein, in the development of the intracellular membrane.

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (−)	if $SS_{\text{binding}} \leq 0.4693$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.1162$ AND $SS_{\text{cellular\_process}} \leq 0.0399$ AND $SS_{\text{biological\_regulation}} \leq 0.5484$	4	0.216
GP6x (−)	$\max(SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}, SS_{\text{signaling}}, SS_{\text{translation\_regulator\_activity}}) < 0.5$	9	0.442
LIME 3feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3510), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1522), (SS_{\text{metabolic\_process}} \leq 0, -0.1211)$	3	0.206
LIME 8feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3128), (SS_{\text{molecular\_carrier\_activity}} \leq 0, -0.2570), (SS_{\text{detoxification}} \leq 0, -0.1822), (SS_{\text{intraspecies\_interaction\_between\_organisms}} \leq 0, -0.1779), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1487), (SS_{\text{metabolic\_process}} \leq 0, -0.1299), (SS_{\text{molecular\_adaptor\_activity}} \leq 0, -0.1126), (SS_{\text{growth}} \leq 0, 0.0456)$	8	0.479
LORE 1 (−)	$SS_{\text{cellular\_process}} \leq 0.6046$ AND $SS_{\text{metabolic\_process}} \leq 0.5021$ AND $SS_{\text{biological\_regulation}} \leq 0.5970$ AND $SS_{\text{binding}} \leq 0.4632$	4	0.189
LORE 2 (−)	$SS_{\text{cellular\_process}} \leq 0.4916$ AND $SS_{\text{metabolic\_process}} \leq 0.5396$ AND $SS_{\text{biological\_regulation}} \leq 0.5419$ AND $SS_{\text{binding}} \leq 0.4449$ AND $SS_{\text{multicellular\_organismal\_process}} \leq 0.0522$ AND $SS_{\text{localization}} \leq 0.2706$	6	0.235

they only have in common that they are both involved in the negative regulation of the protein modification process. In summary, these two proteins have binding functions, but they do not participate in the same biological processes, translating into low similarity values for several semantic aspects and explaining why ML algorithms do not predict interaction.

Table 4.6: Explanations of ML models for the Proline-rich 5-like – Guanine nucleotide-binding 3-like negative pair.

Proline-rich 5-like – Guanine nucleotide-binding 3-like



The proline-rich protein 5-like associates with the mTORC2 complex that regulates the organization of the cytoskeleton. In opposition, guanine nucleotide-binding protein-like 3 is a GTPase-binding nuclear protein.

Model (pred.)	Explanation	Size	IC
DT6 (−)	$SS_{\text{binding}} \leq 0.4693$ AND $0.0399 < SS_{\text{cellular\_process}} \leq 0.6145$ AND $SS_{\text{biological\_regulation}} \leq 0.5484$	3	0.189
GP6x (−)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}, SS_{\text{translation\_regulator\_activity}}) < 0.5$	10	0.454
LIME 3feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3192), (SS_{\text{cellular\_anatomical\_entity}} \leq 0.53, -0.1842), (SS_{\text{metabolic\_process}} \leq 0, -0.1227)$	3	0.206
LIME 8feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3109), (SS_{\text{molecular\_carrier\_activity}} \leq 0, 0.2230), (SS_{\text{cellular\_anatomical\_entity}} \leq 0.53, -0.1621), (SS_{\text{molecular\_adaptor\_activity}} \leq 0, -0.1319), (SS_{\text{metabolic\_process}} \leq 0, -0.1259), (SS_{\text{biomineralization}} \leq 0, -0.1256), (SS_{\text{structural\_molecule\_activity}} \leq 0, -0.1197), (SS_{\text{multicellular\_organismal\_process}} \leq 0, -0.1090)$	8	0.464
LORE 1 (−)	$SS_{\text{cellular\_process}} \leq 0.4285$ AND $SS_{\text{metabolic\_process}} \leq 0.7601$ AND $SS_{\text{biological\_regulation}} \leq 0.7260$	3	0.151
LORE 2 (−)	$SS_{\text{cellular\_process}} \leq 0.6524$ AND $SS_{\text{metabolic\_process}} \leq 0.6210$ AND $SS_{\text{biological\_regulation}} \leq 0.8630$	3	0.151

Protransforming Growth Factor alpha and Disks Large Homolog 2

The last pair is composed by transforming growth factor α[11] and disks large homolog 2[12] (Table 4.7) and correspond to a false positive. Transforming growth factor-α is a mitogenic polypeptide that acts synergistically with protransforming growth factor-β to promote anchorage-independent cell proliferation. Disks large homolog 2 is a member of the membrane-associated guanylate kinase (MAGUK) family and forms a heterodimer with a related family member that may interact at postsynaptic sites to form a postsynaptic protein scaffold of excitatory synapses.

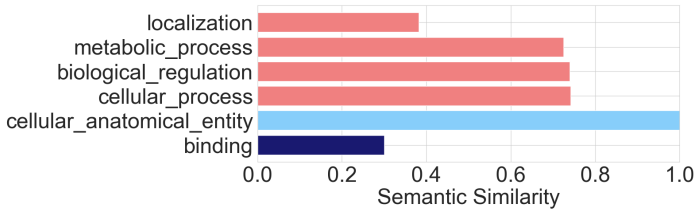
Regarding GO annotations, the two proteins are localized in the basolateral plasma mem-

[11] https://www.uniprot.org/uniprot/P01135
[12] https://www.uniprot.org/uniprot/Q15700

brane and participate in MAPK cascade, which contributes to a high semantic similarity for several semantic aspects. All the models wrongly predict an interaction. These predictions are justified by high similarity values for the most relevant semantic aspects. Curiously, although no information on their interaction was found in the literature, transforming growth factor-β is regulated by disk large homolog 5, and both proteins activate the MAPK cascade [Sezaki et al., 2013]. This led us to think that this is not a true negative pair but rather an unknown interaction that was mistakenly used as a negative example via random negative sampling.

Table 4.7: Explanations of ML models for protransforming growth factor α – Disks large homolog 2 negative pair.

Protransforming growth factor α – Disks large homolog 2



There is no evidence of interaction between these proteins, but there is evidence of an interaction between highly similar proteins: Transforming growth factor-β is regulated by discs large homolog 5, and both proteins activate the MAPK cascade [Sezaki et al., 2013].

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (+)	$SS_{\text{cellular\_anatomical\_entity}} > 0.7589$ AND $SS_{\text{metabolic\_process}} > 0.6023$ AND $0.6145 < SS_{\text{cellular\_process}} \leq 0.7531$	3	0.206
GP6x (+)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}) \geq 0.5$	9	0.423
LIME 3feat (+)	$(SS_{\text{metabolic\_process}} > 0.61, 0.2603), (SS_{\text{biological\_regulation}} > 0.57, 0.1788), (0.49 < SS_{\text{cellular\_process}} \leq 0.84, 0.1702)$	3	0.151
LIME 8feat (+)	$(SS_{\text{metabolic\_process}} > 0.61, 0.2552), (SS_{\text{biological\_regulation}} > 0.57, 0.1876), (SS_{\text{protein\_tag}} \leq 0, -0.1700), (0.49 < SS_{\text{cellular\_process}} \leq 0.84, 0.1639), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1626), (SS_{\text{translation\_regulator\_activity}} \leq 0, 0.1258), (SS_{\text{intraspecies\_interaction\_between\_organisms}} \leq 0, -0.0989), (SS_{\text{biomineralization}} \leq 0, 0.0486)$	8	0.494
LORE 1 (+)	$SS_{\text{cellular\_process}} > 0.5126$	1	0.131
LORE 2 (+)	$SS_{\text{cellular\_anatomical\_entity}} > 0.8126$ AND $SS_{\text{molecular\_adaptor\_activity}} \leq 0.0179$ AND $SS_{\text{cellular\_process}} > 0.5870$ AND $SS_{\text{response\_to\_stimulus}} \leq 0$ AND $SS_{\text{localization}} > 0.3259$ AND $SS_{\text{multicellular\_organismal\_process}} \leq 0.0254$	6	0.326

4.3.6 Frequent Rules Analysis

Table 4.8 shows the most frequent DT6 rules across different models and the number of pairs explained by those rules. Since DTs do not always learn the same cutoff values, after individual models, the cutoff values were rounded to one decimal place to ensure that similar rules with slight variations in cutoff values are treated as the same. Once the cutoff values are standardized, the frequency of each DT rule is computed across the different models. Table 4.8 shows the rules that appear in at least half of the models.

Many rules are clearly supported by existing scientific knowledge, for instance, rule 2 indicates that high values in $SS_{\text{metabolic\_process}}$ and $SS_{\text{cellular\_process}}$ imply an interaction, which makes sense. Proteins participating in the same metabolic or cellular process are very likely to interact. For somewhat lower $SS_{\text{cellular\_process}}$ values, a positive interaction now requires a high *biological_regulation* (rule 3). Another rule in compliance with biological knowledge is rule 1, which takes into account the low values of $SS_{\text{metabolic\_process}}$, $SS_{\text{cellular\_process}}$, and $SS_{\text{biological\_regulation}}$ to indicate that the interaction between the two proteins is not likely. However, some of these general rules do not reflect biology but likely capture the incidental characteristics of the underlying data. For instance, rule 17 classifies positive interactions as those with very low similarity scores in several features, which is probably an attempt to classify poorly annotated proteins (on average, each protein of the dataset has around 23 annotations). This hypothesis justifies that, although the rules appear in most models, the number of protein pairs for which those rules apply is low. In contrast, rules applied to a higher number of pairs seem to be capturing the natural phenomenon.

Although it is expected that the most frequent rules could encode biological information relevant to the PPI predictions, the results seem to show that the model is also learning a phenomenon of functional annotation. The interaction between a pair can be predicted even if it has low similarity values due to the poorly annotated proteins. These results reinforce the need for interpretability and explainability to understand what is actually being learned.

4.3.7 Ablation Studies

Ablation studies using β and γ to filter out leaf classes ($\beta = 1$ or classes that annotated less than 1% of the proteins ($\gamma = 0.01$) are employed. Table 4.9 shows that there are no significant differences in performance when employing these filters and effectively reducing the number of semantic aspects considered.

The impact on explanation size and informativeness is shown in Figures 4.4 and 4.5. The informativeness of the longer explanations given by GP6x and LIME decreases for both variants, indicating that although performance is not modified by filtering out leaf classes or low annotations classes, this can have an impact on explainability.

Table 4.8: Analysis of the most frequent rules across different DT6 models and using the similarity for 51 semantic aspects as input.

	Rule	**Pred.**	**#Models**	**#Pairs**
1	IF $SS_{\text{binding}} \leq 0.5$ AND $0.0 < SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	−	10	17216
2	IF $SS_{\text{metabolic\_process}} > 0.6$ AND $SS_{\text{cellular\_process}} > 0.8$	+	5	4837
3	IF $SS_{\text{metabolic\_process}} \leq 0.6$ AND $SS_{\text{cellular\_process}} > 0.9$	+	5	2359
4	IF $SS_{\text{cellular\_anatomical\_entity}} > 0.8$ AND $0.6 < SS_{\text{cellular\_process}} \leq 0.8$	+	6	2082
5	IF $SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.0$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	−	10	1403
6	IF $SS_{\text{binding}} \leq 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.7$ AND $SS_{\text{cellular\_process}} \leq 0.5$ AND $0.5 < SS_{\text{biological\_regulation}} \leq 0.9$	−	10	841
7	IF $0.5 < SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.8$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	10	574
8	IF $SS_{\text{binding}} \leq 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.7$ AND $0.5 < SS_{\text{cellular\_process}} \leq 0.6$ AND $0.5 < SS_{\text{biological\_regulation}} \leq 0.9$	+	10	515
9	IF $SS_{\text{binding}} > 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.7$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $0.5 < SS_{\text{biological\_regulation}} \leq 0.9$	+	8	472
10	IF $0.5 < SS_{\text{binding}} \leq 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.9$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	−	6	350
11	IF $SS_{\text{binding}} > 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.9$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	6	268
12	IF $SS_{\text{binding}} \leq 0.1$ AND $SS_{\text{cellular\_anatomical\_entity}} \leq 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.0$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	6	248
13	IF $0.5 < SS_{\text{binding}} \leq 0.5$ AND $0.1 < SS_{\text{cellular\_anatomical\_entity}} \leq 0.8$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	−	10	217
14	IF $SS_{\text{binding}} \leq 0.5$ AND $0.1 < SS_{\text{cellular\_anatomical\_entity}} \leq 0.8$ AND $0.6 < SS_{\text{cellular\_process}} \leq 0.8$ AND $SS_{\text{biological\_regulation}} \leq 0.7$	−	7	210
15	IF $0.1 < SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} \leq 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.0$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	−	6	157
16	IF $SS_{\text{cellular\_anatomical\_entity}} \leq 0.8$ AND $0.6 < SS_{\text{cellular\_process}} \leq 0.8$ AND $SS_{\text{biological\_regulation}} > 0.7$	+	6	101
17	IF $0.5 < SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} \leq 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.1$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	7	93
18	IF $SS_{\text{cellular\_anatomical\_entity}} \leq 0.7$ AND $0.0 < SS_{\text{cellular\_process}} \leq 0.5$ AND $SS_{\text{biological\_regulation}} > 0.7$	−	5	86

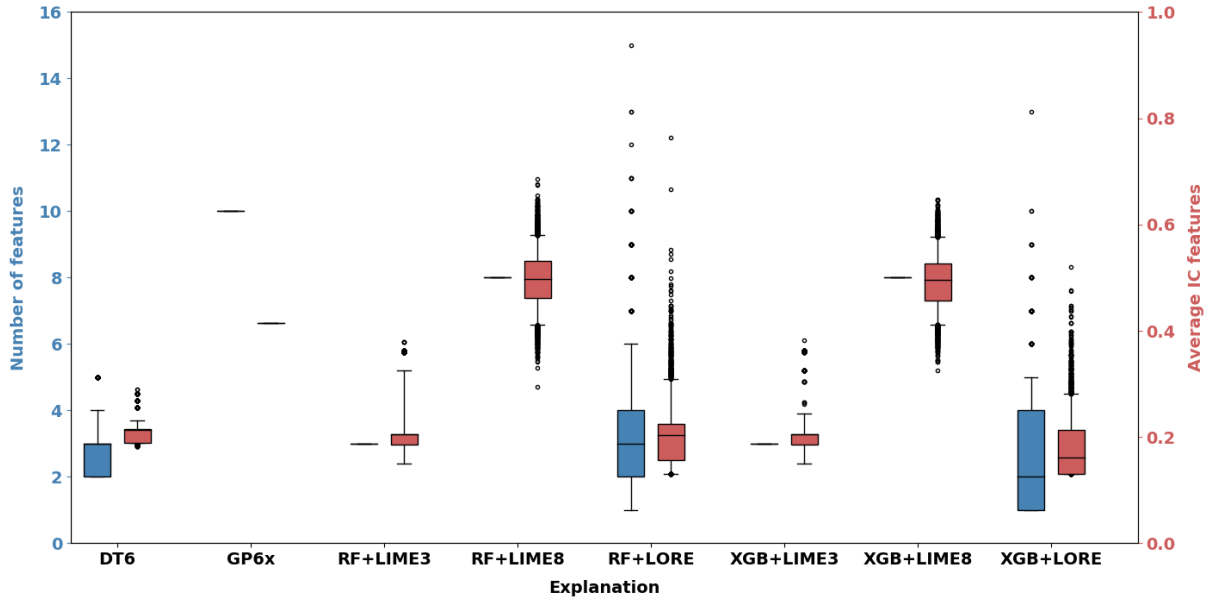


Figure 4.4: Size and informativeness of the explanations obtained for the first partition samples with $\beta = 1$.

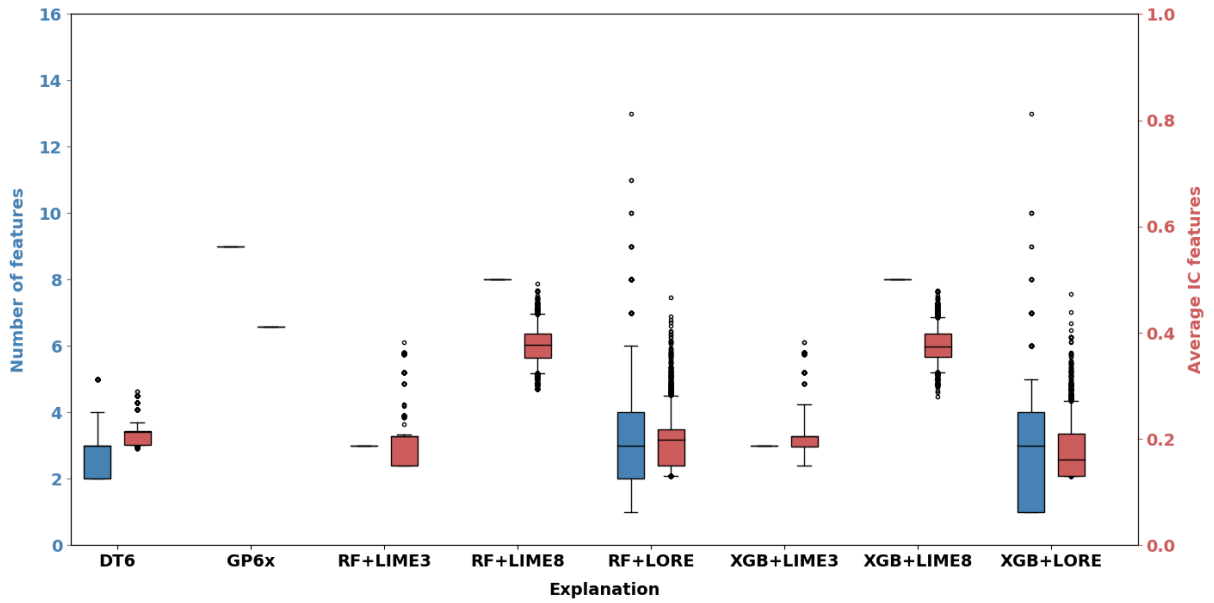


Figure 4.5: Size and informativeness of the explanations obtained for the first partition samples with $\gamma = 0.01$.

Table 4.9: Weighted average F-measure medians (M) and interquartile range (IQR) using different parameters for KGsim2vec.

α, β, γ	**SAs**	**RF**		**XGB**		**DT**		**DT6**		**GP**		**GP6x**	
		M	IQR	M	IQR	M	IQR	M	IQR	M	IQR	M	IQR
$10, 0, 0$	51	0.919	0.005	0.915	0.004	0.899	0.003	0.906	0.002	0.866	0.005	0.866	0.006
$10, 1, 0$	42	0.920	0.004	0.915	0.004	0.899	0.003	0.906	0.002	0.869	0.008	0.866	0.004
$10, 0, 0.01$	24	0.919	0.005	0.915	0.004	0.899	0.002	0.906	0.002	0.878	0.019	0.867	0.005

4.4 Conclusions

Explainability is crucial to support the adoption of ML as a scientific tool that helps understand natural phenomena and drives hypotheses. In the biomedical domain, the abundance of data described with ontologies and integrated into KGs affords a unique opportunity to explore domain knowledge to improve the explainability of ML applications. However, most state-of-the-art ML approaches based on KGs employ KG embeddings, which are not explainable. When the prediction target is finding a relation between two entities represented in the KG, similarity presents itself as a natural explanatory mechanism. Ontologies and KGs can support similarity computation, and measuring similarity between entity pairs according to different aspects represented in the KG opens the door to elucidate relevant aspects behind relations between entities.

The proposed novel method, KGsim2vec, generates vector representations of entity pairs in a KG that can be used to explain relations between them. The explanations are based on computing semantic similarity according to different aspects represented in the KG. The quality of an explanation combines the standard approach based on the number of aspects in the informativeness with their informativeness as measured by the IC captured by each aspect. KGsim2vec is evaluated on PPI prediction, a very relevant application of KG-based supervised learning in the biomedical domain. The experimental results have shown that KGsim2vec can outperform opaque representations given by KG embeddings or end-to-end deep learning approaches. In addition, it generates interesting explanations that capture biological phenomena and gaps in the current knowledge.

Regarding the data leakage investigation between the GO KG and the STRING database in the task of PPI prediction, the results are not able to detect an influence of data leakage, indicating that if this problem exists, its magnitude is not affecting the performance of KG-based PPI predictions.

Chapter 5

Explainable Embedding-based Semantic Representations for Relation Prediction

In recent years, KG embedding methods [Wang et al., 2017] have become increasingly popular to bridge the gap between the complex representations a KG affords and the vectorial representations most ML methods take as input since they map KGs into low-dimensional spaces preserving syntactic and structural properties. KG embeddings are popularly employed in link prediction via a scoring function or as features for supervised learning [Portisch et al., 2022]. However, this represents a significant trade-off: KG embeddings sacrifice the full and rich interpretability offered by KGs, especially when structured by rich ontologies, for the more simple to process latent representations [Palmonari and Minervini, 2020]. The effectiveness and usefulness of KG embeddings approach hinges on the crucial assumption that KG embeddings serve as semantically meaningful representations of the underlying entities. To validate such an assumption, KG embedding methods would need to be explainable (i.e., they would need to afford a human-understandable description of the logic, behavior or factors that influence the representation learning process), but in the vast majority of cases they are not. This is a fundamental requirement to ensure the scientific validity of KG embeddings, or any AI method, as a tool that can be used to uncover new knowledge, help understand the mechanisms underlying natural phenomena, and distinguish meaningful predictions from spurious correlations [Barredo Arrieta et al., 2020].

This is more challenging on the problem of predicting a relation between KG entities that is not defined in the KG. Predicting relations such as PPI or GDA by exploring KGs and ontologies

has been the focus of extensive research in the biomedical domain. Both algorithmic [Zhang and Tang, 2016; Hoehndorf et al., 2011; Asif et al., 2018] and ML approaches [Kulmanov et al., 2021] have been employed to achieve this with success, with KG embeddings particularly excelling at the task [Chen et al., 2019; Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan, 2022; Alshahrani et al., 2017]. However, understanding the nature of these relations requires discerning which aspects of the KG have the most influence on a prediction. This empowers users not only in assessing the reliability of the model itself but also in potentially elucidating the phenomena underlying the relation. For example, if the goal is to explain the interaction between the proteins *transforming growth factor* α and *discs large homolog 2*, generating an explanation based on the fact that they both perform the very specific function *MAPK cascade* would likely increase trust as well as highlight a relevant aspect for interaction. In contrast, a very general explanation, such as the fact that both proteins are present in the *plasma membrane* would contribute to neither.

This chapter presents SEEK (Shared Explainable Embeddings for Knowledge graphs), a novel method for generating explainable KG embeddings that represent entity pairs for relation prediction. The intuition behind this is that an entity pair can be represented by combining embeddings that represent each of their shared semantic aspects, rather than simply combining their respective embeddings. This technique explores the rich semantics of the ontology to identify the shared semantic aspects between related entities based on computing their disjoint common ancestors. Then, these pair embeddings are used to train a supervised ML model for relation prediction. SEEK is fundamentally different from link prediction methods since it produces representations of pairs of entities based on shared semantic aspects, whereas link prediction methods rely on learning representations of individual KG entities and apply a scoring function to estimate the likelihood of triples. Given a prediction, SEEK explains it by computing the importance of each shared semantic aspect in identifying it. Inspired by Watson et al. [2021] and Rossi et al. [2022a], SEEK considers that an explanation includes two complementary views: the set of semantic aspects that, if absent from an entity pair, would render the model incapable of generating that prediction (i.e., necessary explanations); the set of semantic aspects that, if shared by any entity pair, would prompt the model to produce that prediction (sufficient explanations). Since SEEK explains specific predictions rather than the global mechanism of the model, it consequently falls under the category of local post-hoc explanation methods as proposed by Guidotti et al. [2018b].

Within the scope of the thesis and its RQs, this chapter explores embedding-based semantic representations (RQ1) and establishes a comprehensive definition of a class-based semantic aspect that represents pairs of entities by its shared subgraphs (RQ2). The effectiveness of the proposed semantic representation is evaluated in two different tasks: PPI prediction and GDA prediction (RQ3). The extensive experiments show that the proposed approach produces useful explanations besides improving performance over state-of-the-art embedding methods.

Main contributions of the chapter:

- SEEK, a novel method for generating explainable KG embeddings that represent entity pairs for relation prediction.
- Extensions of popular KG embedding methods implementing SEEK.
- Explanation methods that quantify the importance of specific KG semantic aspects for specific relation predictions.
- Extensive experimental results demonstrate that SEEK is able to produce effective explanations for relation prediction as well as generally improving predictive performance on multiple models and biomedical datasets.
- The code is available at `https://github.com/liseda-lab/seek`.

Paper supporting the chapter:

- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Explainable Representations for Relation Prediction in Knowledge Graphs. In 20th International Conference on Principles of Knowledge Representation and Reasoning.*[1] (Appendix B)

5.1 Problem Formulation

SEEK focuses on ontology-rich KGs (Definition 1) and targets the problem of learning a relation between two KG entities, a pair, when the relation itself is not explicitly defined in the KG, using embeddings as inputs for a supervised ML algorithm. This is a fundamentally distinct task from link prediction, where the training set relations are part of the KG. To tackle this relation prediction task, common approaches typically employ three steps: (1) generate embeddings for each entity in the KG; (2) aggregate the embeddings of each entity in a pair into a single representation; (3) use these aggregated representations as input to a supervised learning algorithm to learn a relation prediction model [Sousa et al., 2021; Celebi et al., 2019]. This generates non-explainable predictions since KG embeddings are, of course, non-explainable, as each dimension does not represent any particular meaning, which poses a serious limitation to the use of KG embeddings in a scientific setting.

Moreover, this particular formulation results in two oversimplifications, which may limit its effectiveness and usefulness. Firstly, it relies on aggregating individual embeddings to represent a pair of entities instead of directly learning an embedding that represents the pair. One should clarify that simply representing the pair as yet another entity on the KG would not be a viable

[1] This chapter reproduces the methodology and results presented in this paper.

solution, as it would limit the applicability of the approach to pairs seen at representation learning time and thus fail to generalize to novel pairs. Secondly, it focuses on creating an overall representation of each entity rather than capturing the different semantic aspects that may contribute to the target relation. In large KGs, it is not uncommon for entities to be described according to multiple semantic aspects, but only a few may be relevant for the prediction of a particular relation. In a previous study [Sousa et al., 2020], it was demonstrated that not all branches of the GO are equally important for predicting PPIs.

The problem is then two-fold: (1) to generate latent representations that represent an entity pair directly and (2) to generate latent representations that are amenable to explanation and can capture the relevant semantic aspects for relation prediction.

5.2 Related Work

KG embeddings are not explainable, and there is no widely accepted methodology to effectively explain the predictions of KG embeddings [Palmonari and Minervini, 2020]. CRIAGE [Pezeshkpour et al., 2019] and Kelpie [Rossi et al., 2022b] have made striding efforts towards explaining link prediction based on KG embeddings by identifying the fact to add into or remove from the KG that affects the prediction for a target fact. Betz et al. [2022] also propose a post hoc method that uses adversarial attacks on KG embedding models to identify triples that serve as logical explanations for specific predictions. These works differ fundamentally from ours by focusing on single facts about each entity, whereas SEEK focuses on shared aspects between entity pairs. Additionally, all of these works face the computational challenge posed by having to retrain the KG model after removing a single fact to explain each prediction, and devise heuristic approaches to minimize this aspect. SEEK does not require retraining the model. Instead, SEEK generates explanations by identifying shared semantic aspects and making predictions with the trained model. ExCut [Gad-Elrab et al., 2020] is another approach that uses KG embeddings to identify clusters of entities and then combines it with rule-mining methods to learn interpretable labels.

5.3 SEEK

SEEK is a novel approach that generates explainable vector representations of KG entity pairs to support relation prediction tasks with minimal loss in performance. Code and online tool are available at `https://github.com/liseda-lab/seek`.

Figure 5.1 shows an overview of the SEEK approach. In the first step, the KG is transformed into an RDF graph, which facilitates the subsequent processing. Representations for each ontology class are then learned using a KG embedding method. Notably, SEEK is agnostic to the specific KG embedding method employed and can accommodate a broad range of techniques.

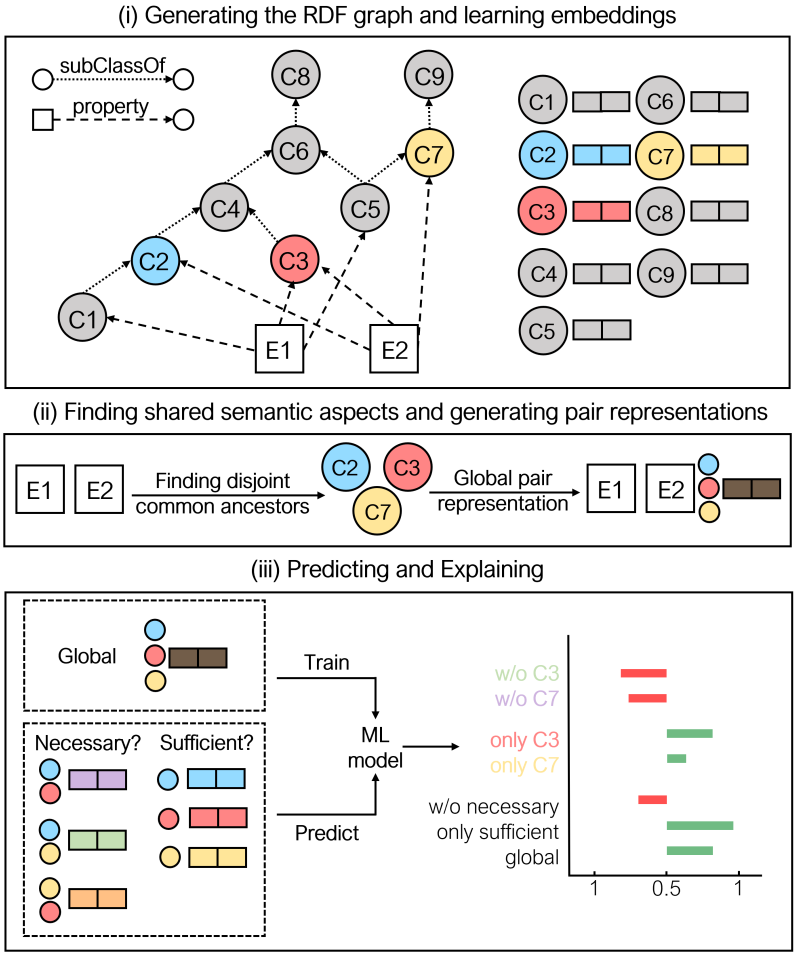


Figure 5.1: Overview of SEEK with the main steps: (i) generating the RDF graph and learning embeddings (ii) finding shared semantic aspects and generating pair representations (iii) predicting and explaining.

The second step is concerned with identifying the shared semantic aspects between the entities of the pair, which are determined by computing the disjoint common ancestors of all classes related to them. The identification of these semantic aspects is essential for the subsequent generation of accurate and meaningful explanations. Having identified the relevant semantic aspects, the final representations of entity pairs are then generated by aggregating the embeddings of the shared semantic aspects.

In the third and final step, supervised learning methods are employed to learn a relation prediction model taking as input the pair embeddings. This model is then used to generate explanations by adopting a perturbation-inspired approach where the contribution of each se-

mantic aspect to the final prediction is assessed in terms of its sufficiency and necessity. The necessary explanations provide insights into the semantic aspects that are necessary for a particular decision to be made, while the sufficient explanations reveal the aspects that are sufficient to support a particular decision. These explanations enable a more thorough understanding of predicted relations and which KG aspects influence it and can be invaluable in identifying potential biases or errors.

5.3.1 Generating the RDF Graph and Learning Embeddings

Ontology-rich KGs are typically defined in OWL. However, the majority of graph processing and analysis tools require RDF graphs. Therefore, the initial step is to convert the KG into an RDF graph following the guidelines provided by the W3C[2]. The conversion process involves transforming simple axioms directly into RDF triples, such as subsumption axioms or data and annotation properties associated with an entity. Multiple triples are created for more complex axioms involving class expressions, which usually require blank nodes. The relations between entities and the ontology classes describing them are usually stored in annotation files in the biomedical domain. These annotations are processed into object properties. After conversion, the nodes in the RDF graph represent ontology classes or individuals, and the edges represent named relations. Finally, a KG embedding method is employed to learn latent representations of all the ontology classes in the KG.

5.3.2 Finding Shared Semantic Aspects and Generating Pair Representations

To generate a representation for an entity pair, the concept of semantic aspect (i.e., a subgraph of the KG that captures a specific perspective of the domain) is explored. A pair of KG entities is represented by the set of semantic aspects they share, unfolding their relationship into different dimensions each based on a shared aspect. The shared semantic aspects are defined as the set of disjoint common ancestors computed over the set of classes that describe each entity.

Taking as an example two entities e_1 and e_2 and their set of linked classes C_1, C_2. To compute the set of disjoint shared aspects, the disjoint common ancestors of C_1 and C_2 are computed. Following [Couto and Silva, 2011], a_1 and a_2 are disjoint common ancestors of a class c if $c \sqsubseteq a_1$, $c \sqsubseteq a_2$, $a_1 \not\sqsubseteq a_2$ and $a_2 \not\sqsubseteq a_1$. First, C_a, the set of common ancestors between the two sets C_1 and C_2, is computed and then filter this set to include only the disjoint common ancestors, each of which represents a shared semantic aspect. The shared semantic aspects of two sets only include indirect common ancestors if they do not subsume other common ancestors. Considering the example in Figure 5.2, the shared semantic aspects of proteins P1 and P2 correspond to *calcium ion binding* and *cellular anatomical entity*.

[2]https://www.w3.org/TR/owl2-mapping-to-rdf/

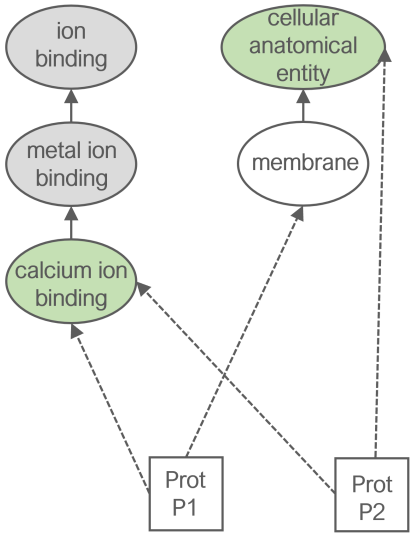


Figure 5.2: A GO KG subgraph to represent the shared semantic aspects of two entities. Green classes represent the disjoint common ancestors of proteins P1 and P2. Grey classes represent the remaining common ancestors.

To represent an entity pair, the embeddings of each class in the shared semantic aspects set are selected and then are aggregated using simple operators such as the Hadamard product, the sum, the average or the L1-norm.

5.3.3 Predicting and Explaining

After obtaining the vector representations, supervised ML algorithms are used to learn relation prediction models and ultimately produce explanations for predicted relations. First, the model is trained using the global representation of the pair, generated by aggregating all shared semantic aspect embeddings. Then, for each target prediction, multiple representations are generated that differ by the presence or absence of a semantic aspect. To understand which semantic aspects are necessary for the prediction, SEEK generates representations that remove each aspect in turn (see Algorithm 2), whereas to understand which aspects are sufficient for the prediction, SEEK generates representations that include a single aspect (see Algorithm 3). A semantic aspect is considered necessary for a prediction if the predicted class changes when it is removed. Likewise, a semantic aspect is considered sufficient for a prediction if the predicted class does not change when it is the only aspect considered.

An explanation is defined as the set of the most relevant shared semantic aspects identified as necessary or sufficient. A necessary explanation is a shared semantic aspect that, when removed from the pair representation, causes the classifier to change its prediction. A sufficient

explanation is a shared semantic aspect that, when used alone to represent a pair, causes the classifier to maintain its prediction. A relation may be explained by multiple necessary and sufficient explanations.

This approach is similar to how saliency XAI methods inject perturbations in the feature space to capture the importance of features. However, it addresses a significant challenge that perturbation or modification-based methods face, including those that aim to explain KG embeddings [Pezeshkpour et al., 2019; Rossi et al., 2022b], which is the need to relearn representations after performing the modification to the data. SEEK avoids this hurdle since it is based on composite representations of ontology classes, which are easy to modify and do not require retraining since the ontology itself is never altered, so the learned class embeddings remain fixed.

The final explanation can be represented as a chart where sufficient and necessary shared semantic aspects are presented alongside their impact on the prediction. In Figure 5.1, both C3 and C7 are necessary to support the prediction since, without either of them, the prediction value changes when compared to the prediction obtained for the global representation. C3 is also a sufficient aspect since it can single-handedly produce a prediction that agrees with the global one. The explanation can be further enriched with the prediction of the global approach, a prediction made with all sufficient shared semantic aspects, and a prediction made without any of the necessary shared semantic aspects, all predictions, including their respective likelihood.

Algorithm 2 Generation of necessary explanations

Input:the entity pair (e_1, e_2);
the KG embedding model K;
the relation prediction model M;
Output: the set of disjoint shared aspects that are necessary for explaining the prediction

$N \leftarrow empty$
$D \leftarrow$ GET DISJOINT SHARED ASPECTS$((e_1, e_2))$
$E \leftarrow$ GET EMBEDDINGS(K, D)
$v \leftarrow$ AGGREGATE(E)
$p \leftarrow$ PREDICT(M, v)
for $d \in D$ **do**
  $e' \leftarrow E.delete(d)$
  $v' \leftarrow$ AGGREGATE(e')
  $p' \leftarrow$ PREDICT(M, v')
  if $p \neq p'$ **then**
    $N.append(d)$
return N

Algorithm 3 Generation of sufficient explanations

Input:the entity pair (e_1, e_2);
the KG embedding model K;
the relation prediction model M;
Output: the set of disjoint shared aspects that are sufficient for explaining the prediction

$S \leftarrow empty$
$D \leftarrow$ GET DISJOINT SHARED ASPECTS$((e_1, e_2))$
$E \leftarrow$ GET EMBEDDINGS(K, d)
$v \leftarrow$ AGGREGATE(E)
$p \leftarrow$ PREDICT(M, v)
for $d \in D$ **do**
  $v' \leftarrow$ GET EMBEDDING(K, d)
  $p' \leftarrow$ PREDICT(M, v')
  if $p == p'$ **then**
    $S.append(d)$
return S

5.4 Evaluation

SEEK is evaluated on two biomedical relation prediction tasks: predicting PPIs and predicting GDAs. Predicting PPIs is a crucial task in molecular biology [Li et al., 2021; Hu et al., 2021b], and several KG embedding-based methods have been employed to tackle it [Kulmanov et al., 2019; Smaili et al., 2018b; Kulmanov et al., 2021, 2019; Xiong et al., 2022]. Due to the high costs and challenges involved in experimentally determining PPI, computational methods can be used to identify protein pairs that are likely to interact, which are subsequently validated through experimental assays rendering the process more efficient. Likewise, predicting the relation between genes and diseases is essential to understanding disease mechanisms and identifying potential biomarkers or therapeutic targets [Eilbeck et al., 2017]. Once again, computational approaches to identify the most promising associations to be further validated are commonly employed, with recent approaches applying KG embedding methods [Alshahrani et al., 2017; Smaili et al., 2018b; Nunes et al., 2023]. However, opaque methods such as KG embeddings are unable to provide explanations behind each prediction. Explanatory mechanisms can elucidate the potential mechanisms behind the predicted relation, which can be helpful to determine the type of experimental procedure that should be applied to confirm the predicted relation but also to identify data biases that can result in misclassification and should be grounds to discard the candidate pair.

Both relation prediction tasks cast as a classification task that takes as input entity pairs and a KG back-boned by an ontology. Ontologies are arranged in a directed acyclic graph, where

ontology classes are connected by subclass relations such that each class is more specific than its ancestor. Moreover, these relationships are transitive, meaning they inherit all ancestors to the root.

5.4.1 Data

Both tasks are grounded on ontology-rich KGs, where PPI employs the GO and GDA is based on the HP. Additionally, prior studies have shown that different branches of these ontologies have varying impacts on achieving precise predictions [Sousa et al., 2020].

Table 5.1: Statistics for each task (PPI and GDA) regarding the number of classes, nodes, and edges. Positive and negative pairs correspond to the number of positive and negative relations.

	PPI	**GDA**
Ontology classes	50422	15656
Literals and blank nodes	462874	443489
Instances	6738	4523
Annotations	349500	160009
Positive Pairs	23571	8189
Negative Pairs	23571	8189

For PPI prediction, the target relations to predict are obtained from the STRING database [Szklarczyk et al., 2021], one of the largest PPI databases that integrate physical interactions and functional associations between proteins from various sources. The protein pairs are filtered to include only pairs that met the following criteria: (i) each protein must be annotated with the GO, (ii) interactions must be extracted from curated databases or experimentally determined, and (iii) interactions must have a confidence score above 0.950. The PPI dataset contains 23571 interacting protein pairs as well as 23571 negative pairs derived from random negative sampling of the same set of proteins. The GO KG is used to describe proteins and is built by integrating the GO [Consortium, 2021] and protein annotation data [Huntley et al., 2015] (see section 2.1 for more details about GO KG). Table 5.1 describes the statistics about PPI data.

For GDA, the target relations to predict are obtained from DisGeNET [Piñero et al., 2019]. The approach in Nunes et al. [2023] is followed, which excludes associations whose sources are used to create some of the ontology annotations. Moreover, each gene and disease must have at least one HP annotation. This resulted in a balanced dataset with a total of 16378 gene-disease pairs. Regarding the KG, the HP KG comprising the HP [Köhler et al., 2020] and HP annotation data to describe genes and diseases is employed (see section 2.1.1 for more details about HP KG). The statistics about GDA data are also shown in Table 5.1.

5.4.2 Models

SEEK is independent of the KG embedding method and of the supervised ML algorithm. For the experiments, five representative KG embeddings are implemented covering translational, semantic matching and random walk-based methods: RDF2Vec [Ristoski and Paulheim, 2016a], OWL2Vec* [Chen et al., 2021a], TransE [Bordes et al., 2013], TransH [Wang et al., 2014] and distMult [Yang et al., 2015].

To generate a pair representation, the average is used as the aggregation, which ensures that the values of each dimension remain within the distribution. In the case of necessary explanations, removing one similar semantic aspect will result in a very similar aggregated representation, revealing that the semantic brings little novel information for the prediction (since a similar semantic aspect is still considered). In the case of sufficient explanations, semantic aspects are evaluated independently.

As supervised ML algorithms, two ensemble methods (RF [Breiman, 2001] and XGB [Chen and Guestrin, 2016]) and a NN-based method (MLP [Rumelhart et al., 1986]) are employed.

5.5 Results and Discussion

5.5.1 Performance Evaluation

To assess the proposed method, the relation prediction performance of SEEK pair representations is compared against the state-of-the-art approach of entity vector aggregation using representative KG embedding methods, supervised ML algorithms and the Hadamard operator. SEEK is not compared to other KG embedding explanation methods such as Kelpie or CRIAGE because they learn embeddings that target link prediction, whereas SEEK learns embeddings to serve as features for supervised ML. The predictive performance of the proposed approach is evaluated against the baselines for each task using 10-fold cross-validation. For each partition, the precision (Pr), recall (Re) and weighted average F1-score (F1) are computed, and the median of the obtained scores (Table 5.2) and the statistical significance of the observed differences are reported.

The results demonstrate that SEEK outperforms the baseline in all cases but one for PPI prediction, while achieving similar or improved scores for GDA. Curiously, the performance of translational methods shows a marked improvement when using SEEK, likely due to the fact that these methods struggle with learning entity representations, but not ontology class representations.

To better understand the differences between the pair representations obtained using the baselines and the ones obtained using SEEK, the RDF2Vec embeddings are plotted using t-distributed stochastic neighbor embedding [Van der Maaten and Hinton, 2008], a nonlinear dimensionality reduction technique that is particularly well-suited for visualizing high-dimensional

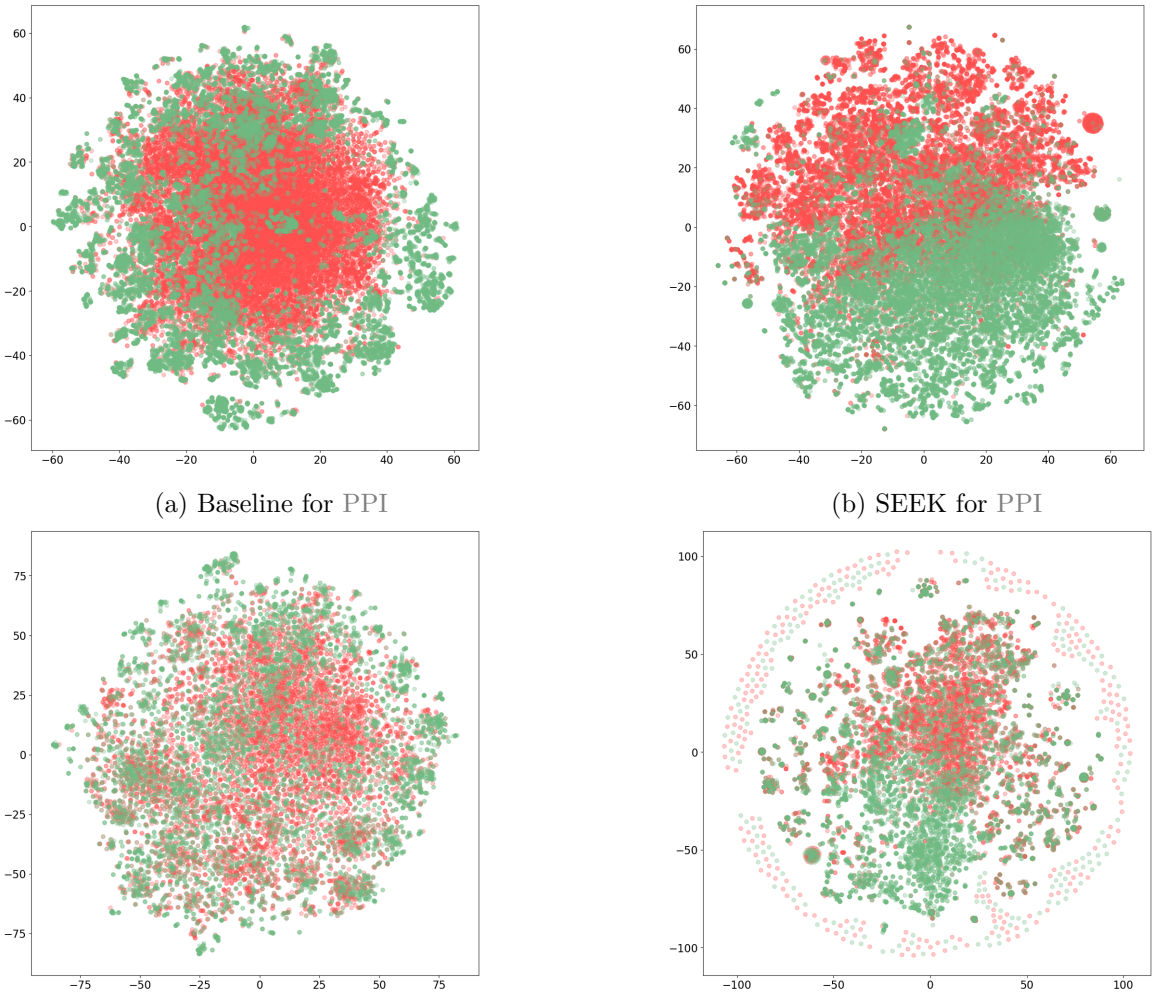
Table 5.2: Medians of precision, recall, and weighted average F1-score (Pr, Re, F1) comparing the approach SEEK to the baseline when coupled with different supervised ML methods (XGB, RF, and MLP) for PPI and GDA prediction. SEEK performance values are underlined when improvements are statistically significant with p-value < 0.05 for the Wilcoxon test against the baselines.

		PPI Prediction						**GDA Prediction**					
		Baseline			**SEEK**			**Baseline**			**SEEK**		
		Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
RDF2Vec	XGB	0.905	0.917	0.910	0.920	0.910	0.915	0.736	0.708	0.724	0.772	0.626	0.719
	RF	0.921	0.881	0.902	0.922	0.892	0.910	0.783	0.673	0.740	0.787	0.625	0.723
	MLP	0.897	0.907	0.902	0.908	0.924	0.917	0.700	0.705	0.696	0.730	0.645	0.703
OWL2Vec*	XGB	0.890	0.881	0.888	0.933	0.925	0.929	0.700	0.664	0.688	0.780	0.647	0.728
	RF	0.913	0.832	0.875	0.922	0.915	0.919	0.730	0.618	0.690	0.780	0.662	0.737
	MLP	0.872	0.865	0.869	0.934	0.923	0.931	0.648	0.676	0.650	0.749	0.642	0.720
distMult	XGB	0.897	0.905	0.902	0.914	0.910	0.912	0.718	0.668	0.704	0.764	0.649	0.722
	RF	0.904	0.860	0.884	0.910	0.897	0.905	0.745	0.636	0.706	0.766	0.637	0.716
	MLP	0.894	0.894	0.896	0.881	0.895	0.888	0.731	0.681	0.715	0.768	0.589	0.698
TransE	XGB	0.642	0.613	0.638	0.914	0.912	0.913	0.526	0.509	0.524	0.755	0.650	0.721
	RF	0.590	0.542	0.583	0.908	0.900	0.905	0.505	0.474	0.502	0.765	0.640	0.719
	MLP	0.250	0.500	0.333	0.882	0.899	0.890	0.500	1.000	0.333	0.779	0.555	0.694
TransH	XGB	0.642	0.614	0.637	0.921	0.918	0.919	0.511	0.493	0.510	0.767	0.651	0.726
	RF	0.586	0.551	0.579	0.912	0.908	0.910	0.500	0.453	0.494	0.770	0.642	0.720
	MLP	0.250	0.500	0.333	0.915	0.920	0.920	0.000	0.000	0.333	0.735	0.665	0.711

data (Figure 5.3). These plots show that SEEK pair representations decrease the overlap between positive and negative pairs and, thus, SEEK is likely to be capturing more meaningful representations.

5.5.2 Effectiveness of Explanations

The effectiveness of the explanations is measured based on how predictive performance varies under two scenarios: when pairs are represented without the *necessary* shared semantic aspects; when pairs are represented by *sufficient* shared semantic aspects only. Table 5.3 presents the results obtained for the PPI and GDA tasks using the two best performing KG embedding methods.



(a) Baseline for PPI

(b) SEEK for PPI

(c) Baseline for GDA

(d) SEEK for GDA

Figure 5.3: t-distributed stochastic neighbor embedding plots comparing SEEK to the baseline using RDF2Vec. Positive pairs are in green and negative pairs are in red.

In the necessary scenario, the necessary explanations are extracted for all correctly predicted relations and produce an ablated representation that does not include any of the necessary shared semantic aspects. The performance variation, in terms of precision (Pr), recall (Re), and F1-score (F1), is measured as the difference in predictive performance between the global representation and the ablated representation. The more negative Δ Pr, Δ Re or Δ F1 are, the more effective are the necessary explanations.

In the sufficient scenario, the sufficient explanations are extracted for all incorrectly predicted relations and produce an ablated representation that only includes the sufficient shared

semantic aspects. The performance variation is also measured as the difference in predictive performance between the global representation and the ablated representation, but in this case, the performance of the global representation is actually zero for all scores since this is only applied to incorrectly predicted relations. A higher Δ value indicates increased effectiveness.

Table 5.3: Explanation effectiveness measured based on the precision (Pr), recall (Re) and weighted average F1-score (F1) variation for PPI and GDA prediction.

		PPI Prediction						**GDA Prediction**					
		RDF2Vec			**OWL2Vec***			**RDF2Vec**			**OWL2Vec***		
		MLP	XGB	RF	MLP	XGB	RF	MLP	XGB	RF	MLP	XGB	RF
w/o necessary	ΔPr	-0.157	-0.109	-0.106	-0.095	-0.099	-0.089	-0.291	-0.296	-0.326	-0.265	-0.332	-0.269
	ΔRe	-0.137	-0.120	-0.153	-0.145	-0.131	-0.129	-0.329	-0.220	-0.277	-0.353	-0.208	-0.329
	ΔF1	-0.148	-0.113	-0.125	-0.117	-0.113	-0.107	-0.264	-0.225	-0.273	-0.270	-0.256	-0.260
only sufficient	ΔPr	0.932	1.000	0.973	0.981	1.000	0.988	0.957	0.969	0.893	0.921	0.986	0.917
	ΔRe	0.959	1.000	0.888	0.927	1.000	0.942	0.737	0.905	0.779	0.777	0.993	0.869
	ΔF1	0.950	1.000	0.945	0.954	1.000	0.967	0.898	0.964	0.896	0.885	0.993	0.925

5.5.3 Explanation Length

The lengths of the explanations, as measured by the number of shared semantic aspects that compose them, are presented in Tables 5.4 and 5.4. In both tasks, the length of necessary explanations is markedly lower than the length of sufficient explanations, highlighting that for many relations, there are no necessary shared semantic aspects. When comparing the shown results to the original number of shared semantic aspects, 9.1 ($\pm$ 6.5) for PPI and 8.5 ($\pm$ 11.0) for GDA, it is possible to verify that sufficient explanations amount to roughly 30% of shared semantic aspects. These sizes are congruent with the number of objects (7 ± 2) humans are able to hold in short-term memory according to cognitive studies [Miller, 1956].

5.5.4 Examples of Explanations

Table 5.5 presents explanations for four protein pairs chosen randomly from the PPI dataset. Each pair represents each of the four possible outcomes: a true positive, a false positive, a true negative, and a false negative.

The first analysed pair consists of paxillin[3] and integrin α-4[4]. There is strong evidence for their interaction [Han et al., 2001] since integrin α-4 binds tightly to paxillin, leading to

[3]https://www.uniprot.org/uniprot/P49023
[4]https://www.uniprot.org/uniprot/P13612

Table 5.4: Explanation average length (Avg) and standard deviation (Std) for PPI prediction and GDA prediction.

		PPI Prediction				**GDA Prediction**			
		RDF2Vec		OWL2Vec*		RDF2Vec		OWL2Vec*	
		Avg	Std	Avg	Std	Avg	Std	Avg	Std
sufficient	MLP	5.6	3.9	5.3	3.5	5.6	7.1	5.5	7.8
	XGB	6.2	3.9	6.3	4.1	6.0	8.3	6.0	9.4
	RF	5.6	3.7	5.9	3.7	5.6	7.7	5.7	8.6
necessary	MLP	0.4	1.1	0.3	1.0	0.6	1.5	0.6	1.1
	XGB	0.4	1.1	0.3	1.0	0.5	1.3	0.5	1.2
	RF	0.4	1.3	0.3	1.1	0.7	1.7	0.7	1.4

increased cell migration and an altered cytoskeletal organization that results in reduced cell spreading. SEEK explanations identify several aspects that are necessary and/or sufficient to explain the interaction and that strongly correlate with the known evidence: focal adhesion, substrate adhesion-dependent cell spreading, cell migration and integrin binding.

The proteins Pulmonary surfactant-associated protein B[5] and granulocyte-macrophage colony-stimulating factor receptor subunit α[6] make up the second pair. Although the proteins share some necessary and/or sufficient semantic aspects, they are very general; therefore, the model does not predict the interaction. However, according to the literature, they are likely involved in the same pulmonary disease [Trapnell et al., 2003]. Both proteins are poorly described under the GO, which can explain why the relation prediction model fails.

The third pair includes the proline-rich 5-like protein[7] and the guanine nucleotide-binding 3-like protein[8]. The model predicts this as a negative pair, and the explanations confirm this, with the removal of the necessary shared semantic aspect resulting in a positive prediction. No interaction is known between these two proteins.

The transforming growth factor α[9] and the discs large homolog 2[10] compose the last pair and correspond to a false positive. The explanations highlight their participation in the MAPK cascade (central signaling pathways that regulate a wide variety of stimulated cellular processes, including proliferation, differentiation, apoptosis and stress response) as well as their co-location

[5]https://www.uniprot.org/uniprot/P07988
[6]https://www.uniprot.org/uniprot/P15509
[7]https://www.uniprot.org/uniprot/Q6MZQ0
[8]https://www.uniprot.org/uniprot/Q9NVN8
[9]https://www.uniprot.org/uniprot/P01135
[10]https://www.uniprot.org/uniprot/Q15700

in the basolateral plasma membrane. It is intriguing to note that although there is no known interaction between these proteins, there is evidence of an interaction between highly similar proteins: transforming growth factor-β is regulated by discs large homolog 5, and both proteins activate the MAPK cascade [Sezaki et al., 2013]. The hypothesis is that this is not, in fact, a true negative pair but a still unknown PPI erroneously used as a negative example through random negative sampling.

5.6 Experimental Validation

After the SEEK evaluation, a collaboration with the biochemistry department of the Faculty of Sciences of the University of Lisbon was initiated. The goal was to validate specific cases, through co-immunoprecipitation assay [Canato et al., 2018; Santos et al., 2020], where SEEK had predicted the interaction between two proteins with great confidence, which was not in the STRING database but the SEEK explanation was scientifically sound. This collaboration allows the experimental confirmation of novel interactions, showing the reliability of SEEK's predictions in identifying previously unknown PPIs.

Co-immunoprecipitation is a widely used experimental technique in biochemistry to validate PPIs. The first step involves selecting an antibody that specifically recognizes one of the proteins involved in the target interaction. This antibody is immobilized on a solid support, such as A/G agarose beads. The second step consists of lysing cells or tissues expressing the proteins of interest to release all cellular components, including proteins. The lysate is then incubated with the antibody-bound beads, allowing the antibody to bind to its target protein. In the third step, the antibody-protein complexes are isolated, and the beads are washed to remove contaminants. The final step involves separating the antibody from the immunoprecipitated proteins. The immunoprecipitated proteins are analyzed using Western blotting techniques to confirm the presence of the other protein in the target interaction.

Unfortunately, validation through the co-immunoprecipitation assay depends on several factors, including whether the protein is expressed sufficiently for the antibodies to bind. The experiments were successful for only two of the four pairs chosen for validation. For the remaining pairs, the antibodies failed to immunoprecipitate the desired protein. In the two cases where immunoprecipitation was successful, the pair consisting of RAS guanyl-releasing protein 1 (RASGRP1) and G-protein coupled receptor 183 (GPR183) showed a clear interaction. However, in the other case, the bands and molecular mass of the target proteins were ambiguous, making the interaction unclear.

Taking a closer look at the experiment for the pair GPR183[11]-RASGRP1[12], GPR183 was chosen for the pull-down. GPR183 is a G-protein coupled receptor and RASGRP1 functions

[11] https://www.uniprot.org/uniprot/P32249
[12] https://www.uniprot.org/uniprot/O95267

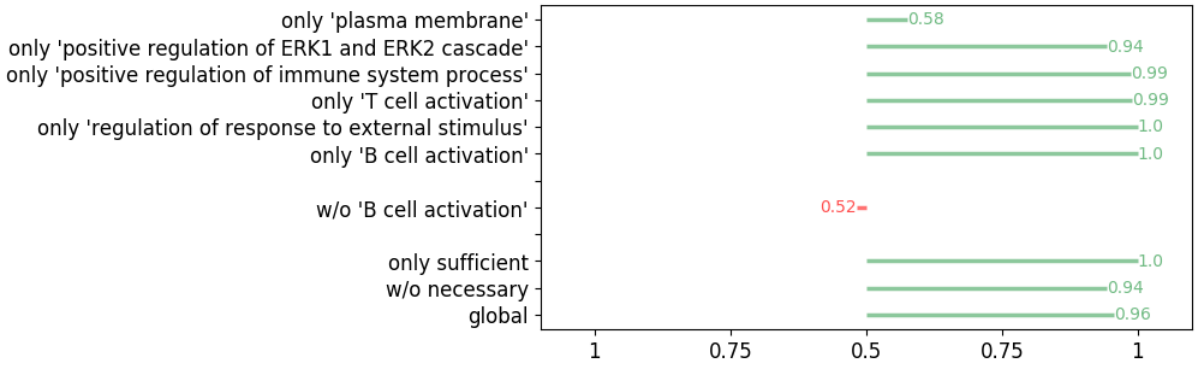


Figure 5.4: Bar chart using different sets of disjoint common ancestors to represent the GPR183-RASGRP1 pair. Each bar represents the likelihood returned by the ML model of the predicted class being correct (red for class 0 and green for class 1).

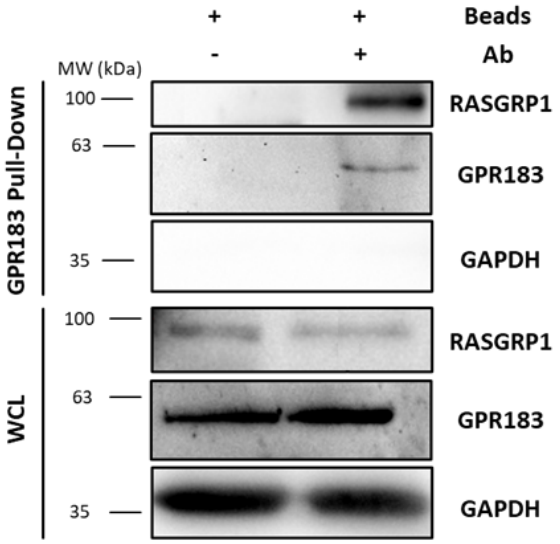


Figure 5.5: Wertern blot showing an interaction between GPR183 and RASGRP1. Co-immunoprecipitation was performed using HELA cells. Cell lysates incubated with non-conjugated beads were used as a control. RASGRP1 was detected by western blot after co-immunoprecipitation of GPR183 (anti-GPR183 antibody cross-linked to protein A/G agarose beads). GAPDH was used as the loading control.

as a guanine-nucleotide exchange factor that activates Ras and is regulated by both calcium and diacylglycerol. For the validation, GPR183 was immunoprecipitated from HeLa cells and lysates incubated only with beads were used as a negative control. After immunoprecipitation, a western blot (Figure 5.5) was used to detect either GPR183 or RASGRP1. Results show that GPR183 is immunoprecipitated. Furthermore, RASGRP1 was also detected after GPR183 immunoprecipitation, validating the GPR183-RASGRP1 interaction. The SEEK explanation is shown in Figure 5.5.

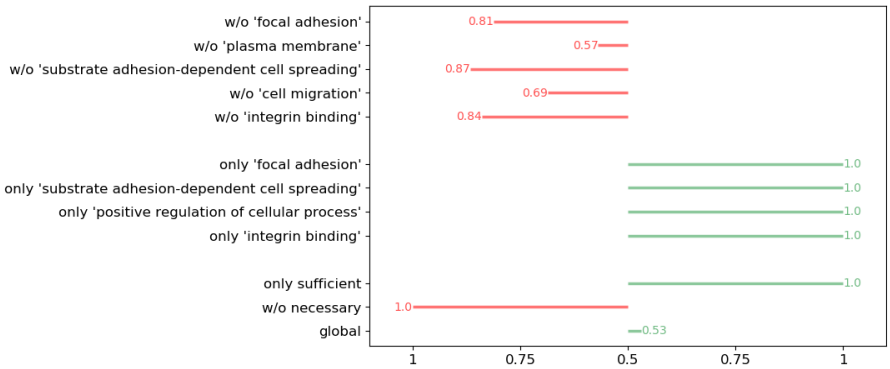
5.7 Conclusions

Existing KG embedding methods are not explainable, which hinders their application in complex and critical domains. This is especially challenging in relation prediction, where understanding which KG semantic aspects are more relevant for a relationship between two KG entities can provide insightful knowledge about its mechanisms and help distinguish meaningful predictions from spurious correlations.

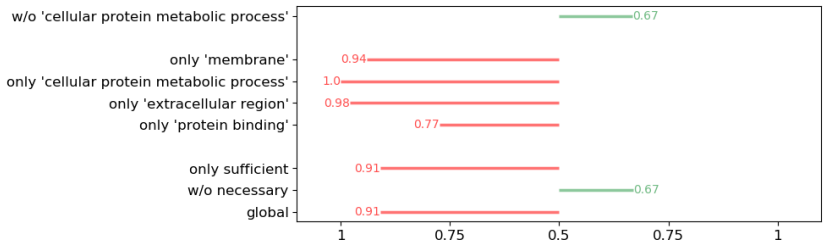
SEEK is a novel approach for learning and explaining representations of KG entity pairs based on their shared semantic space for relation prediction. Its explanatory mechanism is based on generating perturbed representations to identify the relevant semantic aspects of the KG that explain a relation. Since it does not require retraining of representations, it is particularly efficient. SEEK is evaluated on PPI prediction and GDA prediction, two complex and core tasks in the biomedical domain. SEEK clearly outperforms state-of-the-art learning representation methods in performance while generating explanations that can identify critical factors driving biological phenomena.

Table 5.5: Explanations of PPI prediction models for four randomly selected pairs. For each pair, a bar chart is provided using different sets of disjoint common ancestors to represent the pair. On the x-axis, each bar represents the likelihood returned by the MLP model of the predicted class being correct. Classes are represented by colors (red for class 0 and green for class 1).

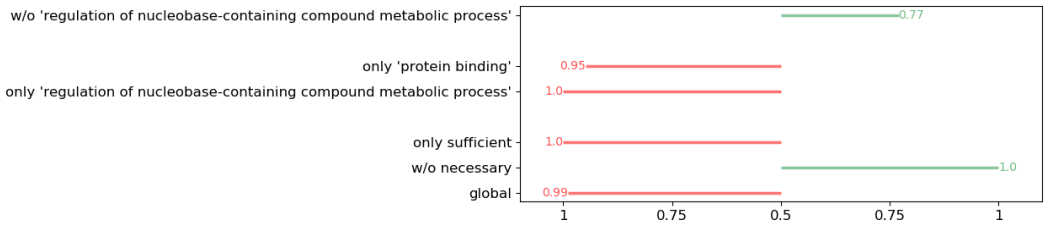
Paxillin – Integrin α-4 (True positive)



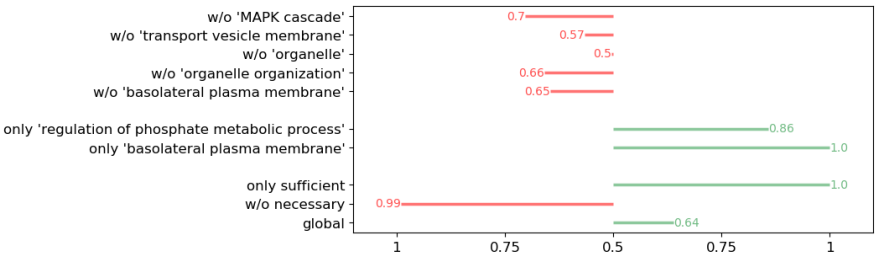
Pulmonary surfactant-associated protein B – Granulocyte-macrophage colony-stimulating factor receptor subunit α (False negative)



Proline-rich 5-like – Guanine nucleotide-binding 3-like (True Negative)



Transforming growth factor α – Discs large homolog 2 (False Positive)



Chapter 6

Embedding-based Semantic Representations with Negative Statements for Relation Prediction

Regardless of their domain, the vast majority of KG facts are represented as positive statements, e.g. (*hemoglobin*, *hasFunction*, *oxygen transport*). Under a Closed World Assumption, negative statements are not required since any missing fact can be assumed as a negative. However, real-world KGs reside under the Open World Assumption, where non-stated negative facts are formally indistinguishable from missing or unknown facts, which can have important implications across a variety of tasks. The importance of negative statements is increasingly recognized [Arnaout et al., 2021a; Flouris et al., 2006]. For example, in the biomedical domain, the knowledge that a patient does not exhibit a given symptom or a protein does not perform a specific function is crucial for both clinical decision-making and biomedical insight. While ontologies are able to express negation and the enrichment of KGs with interesting negative statements is gaining traction, existing KG embedding methods are not able to adequately utilize them [Kulmanov et al., 2021], which ultimately results in less accurate representations of entities.

This chapter presents TrueWalks, to the best of available information, the first-ever approach that can incorporate negative statements into the KG embedding learning process. This is fundamentally different from other KG embedding methods, which produce negative statements by negative random sampling strategies to train representations that bring the representations of nodes that are linked closer, while distancing them from the negative examples. TrueWalks uses explicit negative statements to produce entity representations that consider both existing and lacking attributes. For example, for the negative statement (*Bruce Willis*, *NOT birthPlace*, *U.S.*),

the proposed representation would be able to capture the similarity between Bruce Willis and Ryan Gosling, since neither was born in the U.S (see Figure 6.1). The explicit declaration of negative statements such as these is an important aspect of more accurate representations, especially when they capture unexpected negative statements (i.e., most people would expect that both actors are U.S. born). Using TrueWalks, Bruce Willis and Ryan Gosling would be similar not just because they are both actors but also because neither was born in the U.S.

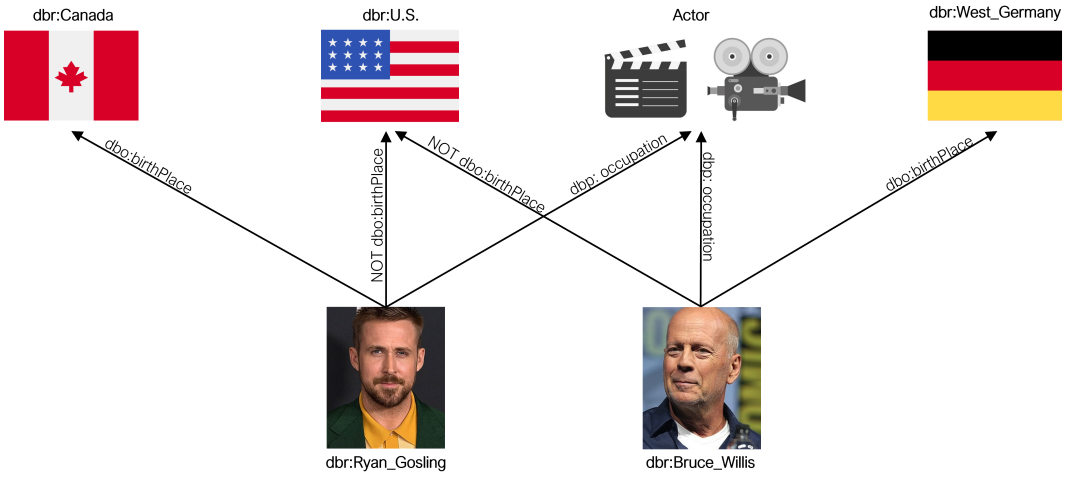


Figure 6.1: A DBPedia example motivating the negative statements problem. The author of Bruce Willis' picture is Gage Skidmore.

TrueWalks generates walks that can distinguish between positive and negative statements and consider the semantic implications of negation in KGs that are rich in ontological information, particularly concerning inheritance. This is of particular importance for applications in the biomedical domain, where the inadequacy of embedding approaches regarding negative statements has been identified as a crucial limitation [Kulmanov et al., 2021]. The resulting embeddings can be employed to determine semantic similarity or as features for relation prediction. The effectiveness of TrueWalks approach is evaluated in two different tasks: PPI prediction and GDA prediction.

Within the scope of the thesis and its RQs, this chapter also explores embedding-based semantic representations (RQ1), and it proposes a definition of property-based semantic aspects where entities are represented through both positive and negative aspects (RQ2). The efficacy of the proposed semantic representation is evaluated in two distinct tasks: PPI prediction and GDA prediction (RQ3). The experimental results show that the proposed approach improves performance over state-of-the-art embedding methods and widely used semantic similarity measures.

Main contributions of the chapter:

- TrueWalks, a novel method to generate random walks on KGs that are aware of negative statements and results in the first KG embedding approach that considers negative statements.
- Extensions of popular path-based KG embedding methods implementing the TrueWalks approach.
- Existing KGs enriched with negative statements and benchmark datasets for two popular biomedical KG applications: PPI prediction and GDA prediction.
- Experimental results that demonstrate the superior performance of TrueWalks when compared to state-of-the-art KG embedding methods.
- The code is available at https://github.com/liseda-lab/TrueWalks and the datasets are available at https://doi.org/10.5281/zenodo.7709195.

Papers supporting the chapter:

- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Biomedical Knowledge Graph Embeddings with Negative Statements. In International Semantic Web Conference.*[1] (Appendix C)
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Benchmark datasets for biomedical knowledge graphs with negative statements. In Workshop on Semantic Web solutions for large-scale biomedical data analytics at Extended Semantic Web Conference.*[2] (Appendix D)

6.1 Problem Formulation

TrueWalks addresses the task of learning a relation between two KG entities (which can belong to the same or different KGs) when the relation itself is not encoded in the KG. Two distinct approaches are employed: (1) using the KG embeddings of each entity as features for a ML algorithm and (2) comparing the KG embeddings directly through a similarity metric.

This work targets ontology-rich KGs (Definition 1) that use an ontology to provide rich descriptions of real-world entities instead of focusing on describing relations between entities themselves. These KGs are common in the biomedical domain. As a result, the KG's richness lies in the TBox, with a comparatively less complex ABox, since entities have no links between

[1] This chapter reproduces the methodology and results presented in this paper.
[2] This chapter provides a summarized version of this paper.

them. This work focuses on OWL [Grau et al., 2008] ontologies since biomedical ontologies are typically developed in OWL or have an OWL version.

Biomedical entities in a KG are typically described through positive statements that link them to an ontology. For instance, to state that a protein P performs a function F described under the GO, a KG can declare the axiom $P \sqsubseteq \exists hasFunction.F$. However, the knowledge that a given protein does not perform a function can also be relevant, especially to declare that a given protein does not have an activity typical of its homologs [Gaudet and Dessimoz, 2017]. Likewise, the knowledge that a given disease does not exhibit a particular phenotype is also decisive in understanding the relations between diseases and genes [Liu and Zhu, 2021]. This work considers the definition of grounded negative statements proposed by Arnaout et al. [2021a] as $\neg(s, p, o)$ which is satisfied if $(s, p, o) \notin KG$ and expressed as a *NegativeObjectPropertyAssertion*[3]. Similar to what was done in Arnaout et al. [2021a], there is no negative object property assertion for every missing triple. Negative statements are only included if there is clear evidence that a triple does not exist in the domain being captured. Taking the protein example, negative object property assertions only exist when it has been demonstrated that a protein does not perform a particular function.

An essential difference between a positive and a negative statement of this kind is related to the implied inheritance of properties exhibited by the superclasses or subclasses of the assigned class. Considering that $(P_1, hasFunction, F_1)$ and $(F_1, subClassOf, F_2)$. This implies that $(P_1, hasFunction, F_2)$, since an individual with a class assignment also belongs to all superclasses of the given class, e.g., a protein that performs *iron ion binding* also performs *metal ion binding*

[3]https://www.w3.org/TR/owl2-syntax/#Negative_Object_Property_Assertions

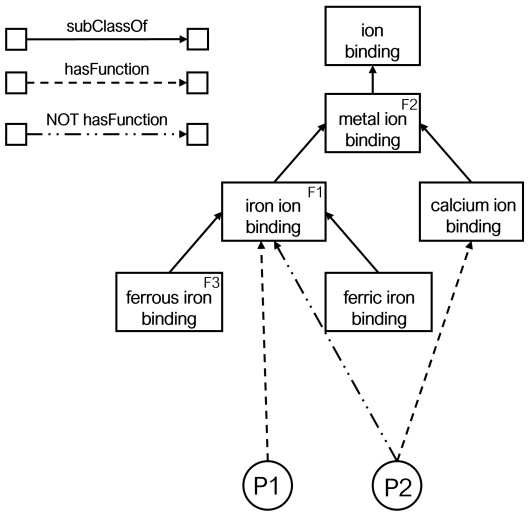


Figure 6.2: A GO KG subgraph motivating the reverse inheritance problem.

(see Figure 6.2). This implication is easily captured by directed walk generation methods that explore the declared subclass axioms in an OWL ontology. However, when there is a negative statement, such as $\neg(P_2, \mathit{hasFunction}, F_1)$, it does not imply that $\neg(P_2, \mathit{hasFunction}, F_2)$. There are no guarantees that a protein that does not perform *iron ion binding* also does not perform *metal ion binding*, since it can very well, for instance, perform *calcium ion binding*. However, for $(F_3, \mathit{subClassOf}, F_1)$ the negative statement $\neg(P_2, \mathit{hasFunction}, F_1)$ implies that $\neg(P_2, \mathit{hasFunction}, F_3)$, as a protein that does not perform *iron ion binding* also does not perform *ferric iron binding* nor *ferrous iron binding*. Therefore, it is necessary to declare that protein P_1 performs both F_1 and F_2, but that P_2 does not perform F_1 and F_3. Since OWL ontologies typically declare subclass axioms, there is no opportunity for typical KG embedding methods to explore the reverse paths that would more accurately represent a negative statement.

The problem is then two-fold: how can the *reverse inheritance* implied by negative statements be adequately explored by walk-based KG embedding methods, and how can these methods distinguish between negative and positive statements.

6.2 Related Work

Approaches to enrich existing KGs with interesting negative statements have been proposed both for general-purpose KGs such as Wikidata [Arnaout et al., 2021b] and for domain-specific ones such as the GO [Fu et al., 2016; Warwick Vesztrocy and Dessimoz, 2020]. Exploring negative statements has been demonstrated to improve the performance of various applications. Arnaout et al. [2021a] developed a method to enrich Wikidata with interesting negative statements and its usage improved the performance on entity summarization and decision-making tasks. Warwick Vesztrocy and Dessimoz [2020] have designed a method to enrich the GO [Consortium, 2021] with relevant negative statements indicating that a protein does not perform a given function. This work demonstrated that a balance between positive and negative annotations supports a more reasonable evaluation of protein function prediction methods. Similarly, Fu et al. [2016] enriched the GO with negative statements and demonstrated an associated increase in protein function prediction performance. The relevance of negative annotations has also been recognized in the prediction of gene-phenotype associations in the context of the Human-Phenotype Ontology (HP) [Köhler et al., 2020], but the topic remains unexplored [Liu and Zhu, 2021]. It should be highlighted that KG embedding methods have not been employed to explore negative statements in any of these approaches.

6.3 TrueWalks

An overview of TrueWalks is shown in Figure 6.3. The first step is the transformation of the KG into an RDF Graph. Next, the proposed random walk generation strategy that is aware of

positive and negative statements is applied to the graph to produce a set of entity sequences. The positive and negative entity walks are fed to neural language models to learn a dual latent representation of the entities. TrueWalks has two variants: one that employs the classical skip-gram model to learn the embeddings (TrueWalks); and one that employs a variation of skip-gram that is aware of the order of entities in the walk (TrueWalksOA, i.e. order-aware).

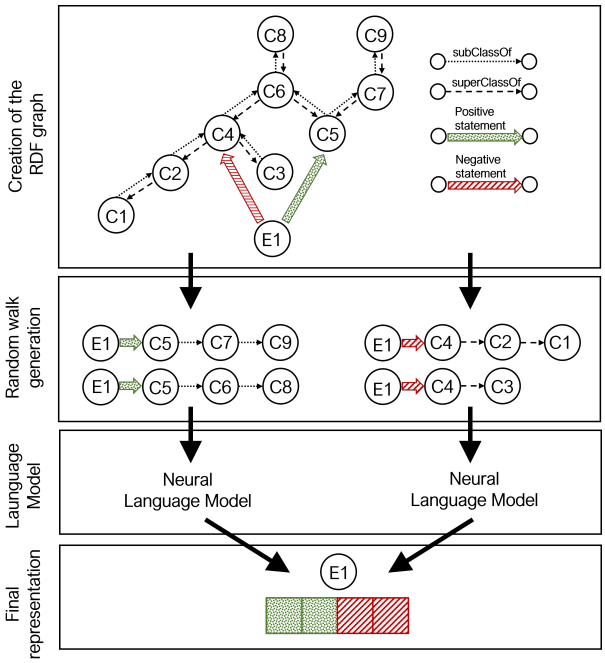


Figure 6.3: Overview of the TrueWalks method with the four main steps: (i) creation of the RDF graph, (ii) random walk generation with negative statements; (iii) neural language models, and (iv) final representation.

6.3.1 Creation of the RDF Graph

The first step is the conversion of an ontology-rich KG into an RDF graph. This is a directed, labeled graph, where the edges represent the named relations between two resources or entities, represented by the graph nodes[4]. The transformation is performed according to the *OWL to RDF Graph Mapping* guidelines defined by the W3C[5]. Simple axioms can be directly transformed into RDF triples, such as subsumption axioms for atomic entities or data and annotation properties associated with an entity. Axioms involving complex class expressions are transformed into multiple triples, which typically require blank nodes.

[4]https://www.w3.org/RDF/
[5]https://www.w3.org/TR/owl2-mapping-to-rdf/

Considering the following existential restriction of the class *obo:GO_0034708 (methyltransferase complex)* that encodes the fact that a methyltransferase complex is part of at least one intracellular anatomical structure:

ObjectSomeValuesFrom(obo:BFO_0000050 (part of),
obo:GO_0005622 (intracellular anatomical structure))

Its conversion to RDF results in three triples:

(obo:GO_0034708, rdfs:subClassOf, _:x)
(_:x, owl:someValuesFrom,obo:GO_0005622)
(_:x, owl:onProperty,obo:BFO_0000050)

where _:x denotes a blank node.

6.3.2 Random Walk Generation with Negative Statements

The next step is to generate the graph walks that will make up the corpus (see Algorithm 4). For a given graph $G = (V, E)$ where E is the set of edges and V is the set of vertices, for each vertex $v_r \in V_r$, where V_r is the subset of individuals targeted for representation learning, TrueWalks generates up to w graph walks of maximum depth d rooted in vertex v_r. A depth-first search algorithm is employed, extending on the basic approach in Ristoski and Paulheim [2016a]. In the first iteration, TrueWalks finds either a positive or negative statement. From then on, walks are biased: a positive statement implies that whenever a subclass edge is found it is traversed from subclass to superclass, whereas a negative statement results in a traversal of subclass edges in the opposite direction (see also Figure 6.3). This generates paths that follow the pattern $v_r \rightarrow e_{1i} \rightarrow v_{1i} \rightarrow e_{2i}$. The set of walks is split in two, negative statement walks and positive statement walks. This will allow the learning of separate latent representations, one that captures the positive aspect and one that captures the negative aspect.

An important aspect of the proposed approach is that, since OWL is converted into an RDF graph for walk-based KG embedding methods, a negative statement declared using a simple object property assertion (e.g. *notHasFunction*) could result in the less accurate path: *Protein P* $\rightarrow$ *notHasFunction* $\rightarrow$ *iron ion binding* $\rightarrow$ *subClassOf* $\rightarrow$ *ion binding*. Moreover, random walks directly over the *NegativeObjectPropertyAssertion*, since it is decomposed into multiple triples using blank nodes, would also result in inaccurate paths. However, the proposed algorithm produces more accurate paths, e.g.: *Protein P* $\rightarrow$ *notHasFunction* $\rightarrow$ *iron ion binding* $\rightarrow$ *superClassOf* $\rightarrow$ *ferric iron binding* by adequately processing the *NegativeObjectPropertyAssertion*.

6.3.3 Neural Language Models

Two alternative approaches are employed to learn a latent representation of the individuals in the KG. For the first approach, the skip-gram model [Mikolov et al., 2013] is used, which predicts

the context (neighbor entities) based on a target word or, in the context of this work, a target entity.

Let $f : E \rightarrow \mathbb{R}^d$ be the mapping function from entities to the target latent representations, where d is the number of dimensions of the representation (f is then a matrix $|E| \times d$). Given a context window c, and a sequence of entities $e_1, e_2, e_3, ..., e_L$, the objective of the skip-gram model is to maximize the average log probability p:

$$\frac{1}{L} \sum_{l=1}^{L} \log p(e_{l+c}|e_l) \tag{6.1}$$

where $p(e_{l+c}|e_l)$ is calculated using the softmax function:

$$p(e_{l+c}|e_l) = \frac{\exp(f(e_{l+c}) \cdot f(e_l))}{\sum_{e=1}^{E} \exp(f(e) \cdot f(e_l))} \tag{6.2}$$

where $f(e)$ is the vector of the entity e.

To improve computation time, TrueWalks employs a negative sampling approach based in Mikolov et al. [2013] that minimizes the number of comparisons required to distinguish the target entity by taking samples from a noise distribution using logistic regression, where there are k negative samples for each entity.

The second approach is the structured skip-gram model [Ling et al., 2015], a variation of skip-gram that is sensitive to the order of words or, in the context of this work, entities in the graph walks. The critical distinction of this approach is that, instead of using a single matrix f, it creates $c \times 2$ matrices, $f_{-c}, ..., f_{-2}, f_{-1}, f_1, ..., f_c$, each dedicated to predicting a specific relative position to the entity. To make a prediction $p(e_{l+c}|e_l)$, the method selects the appropriate matrix f_l.

The neural language models are applied separately to the positive and negative walks, producing two representations for each entity.

6.3.4 Final Representations

The two representations of each entity need to be combined to produce a final representation. Different vector operations can, in principle, be employed, such as the Hadamard product or the L1-norm. However, especially since these vectors will be employed as inputs for ML methods, it is better to create a feature space that allows the distinction between the negative and positive representations, motivating the use of a simple concatenation of vectors.

6.4 Evaluation

While there have been attempts to enhance current KGs with interesting negative statements, no benchmark datasets have been established to evaluate learning tasks over those KGs, to the

Table 6.1: Statistics for the RDF representation of each ontology (GO and HP) regarding classes, nodes, edges.

	GO	**HP**
Classes	50918	17060
Literals and blank nodes	532373	442246
Edges	1425102	1082859

best of available information. In this work, existing biomedical KGs are enriched with negative statements and a collection of datasets for different biomedical tasks of relation prediction is proposed. Two successful biomedical ontologies are enriched: GO, which covers distinct semantic aspects of gene products' function, and HP, which describes the universe of concepts related to phenotypic abnormalities found in human hereditary diseases. Regarding the datasets, they are grouped according to the task: PPI prediction, GDA prediction. These two tasks have significant implications for understanding the underlying mechanisms of biological processes and disease states.

Each benchmark dataset comprises several pairs of biomedical entities (or instances) that can be of the same type (protein-protein) or distinct types (gene-disease) with the respective label (1 for the positive pairs and zero for the negative pairs). Tables 6.1 and 6.2 show the KGs' and datasets' statistics for each task. Since for GDA prediction, the target relation happens between two types of instances (genes and diseases), the instance numbers in Table 6.2 appear separately. Moreover, in the case of PPI prediction, the GO KG that has been subjected to a negative statement enrichment approach is exclusively employed. However, when it comes to GDA prediction, it relies on the HP KG, which lacks a negative statement enrichment approach, resulting in a significant imbalance between the number of positive and negative statements.

Both tasks are modeled as relation prediction tasks. For PPI prediction, TrueWalks embeddings are employed both as features for a supervised learning algorithm and directly for similarity-based prediction. For GDA prediction, since embeddings for genes and diseases are learned over two different KGs, the evaluation focus only on supervised learning. RF algorithm is employed across all classification experiments with the same parameters.

To build these datasets, three main steps are adopted. The first one consists of enriching the KGs. The KG is constructed using the owlready2 package[6], which parses the ontology file in OWL format and processes the annotation file. The annotation file contains positive and negative statements used to describe entities. The guidelines established by the W3C[7]

[6]https://owlready2.readthedocs.io/en/v0.37/
[7]https://www.w3.org/TR/owl2-mapping-to-rdf/

Table 6.2: Statistics for each task's dataset regarding the number of instances, pairs, positive and negative statements.

	PPI prediction	**GDA prediction**
Instances	440	174 + 107
Positive Pairs	1024	107
Negative Pairs	1024	107
Positive statements	7364	14828
Negative statements	8579	9191

```
<owl:NamedIndividual rdf:about="http://purl.obolibrary.org/obo/GO_0048268">
    <rdf:type rdf:resource="http://purl.obolibrary.org/obo/GO_0048268"/>
</owl:NamedIndividual>
<rdf:Description>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#NegativePropertyAssertion"/>
    <owl:sourceIndividual rdf:resource="http://purl.obolibrary.org/obo/GO_0048268"/>
    <owl:assertionProperty rdf:resource="http://purl.obolibrary.org/obo#has_function"/>
    <owl:targetIndividual rdf:resource="https://www.uniprot.org/uniprotkb/Q9BY11"/>
</rdf:Description>
```

Figure 6.4: Example of how the negative statements are defined in the OWL file.

are used to define the negative statements as negative object property assertions[8]. To do so, metamodeling is used, and each ontology class is represented as a class and an individual. This situation translates into using the same internationalized resource identifier. Then, a negative object property assertion is used to state that the individual representing a biomedical entity is not connected by the object property expression to the individual representing an ontology class, as depicted in Figure 6.4. The second step consists of extracting pairs of entities from bioinformatic databases. The third step involves selecting the pairs containing KG entities that are well described with positive and negative statements.

The following subsections describe in more detail the KGs as the characteristics of each task.

6.4.1 Biomedical Knowledge Graphs

Two KGs back-boned by biomedical ontologies are used: the GO KG and the HP KG. Table 6.1 shows the statistics for each ontology.

The GO KG is built by integrating three sources: the GO[9] itself, the GO Annotation

[8]https://www.w3.org/TR/owl2-syntax/#Negative_Object_Property_Assertions

[9]The GO was downloaded on September 2021. It is available at http://release.geneontology.org/2021-09-01/ontology/index.html.

data[10] [Consortium, 2021], and negative GO associations produced in Warwick Vesztrocy and Dessimoz [2020][11]. A GO annotation associates a gene product with a GO class that describes it. The GO annotations corresponding to positive statements from the GOA database for human species. For each gene product P and each of its association statements to a function F in GOA, the assertion $(P, hasFunction, F)$ is added. The negative GO associations produced in Warwick Vesztrocy and Dessimoz [2020] are added. These negative associations were derived from expert-curated annotations of protein families on phylogenetic trees. For each gene product P and each of its association statements to a function F in the negative GO associations dataset, a negative object property assertion is added. To do so, metamodeling (more specifically, punning [12]) is used. Each ontology class is represented as both a class and an individual. This situation translates into using the same internationalized resource identifier. Then, a negative object property assertion is used to state that the individual representing a biomedical entity is not connected by the object property expression to the individual representing an ontology class.

HP [Köhler et al., 2020] describes phenotypic abnormalities found in human hereditary diseases. HP annotations can link diseases to HP classes through both positive and negative statements. The construction of HP KG[13] is similar to that of the GO KG. A negative annotation from HP that includes NOT indicates that a disease does not cause that phenotype, so they are included as negative statements.

6.4.2 Protein-Protein Interaction Prediction Dataset

Predicting PPIs is a fundamental task in molecular biology that can explore both sequence and functional information [Hu et al., 2021b]. Given the high cost of experimentally determining PPI, computational methods have been proposed as a solution to the problem of finding protein pairs that are likely to interact and thus provide a selection of good candidates for experimental analysis. In recent years, a number of approaches for PPI prediction based on functional information as described by the GO have been proposed [Zhang and Tang, 2016; Kulmanov et al., 2019; Smaili et al., 2018b; Sousa et al., 2020; Kulmanov et al., 2021]. Therefore, the GO KG is used to describe the proteins of the dataset.

The positive examples are extracted from the STRING [Szklarczyk et al., 2021] database. The selection of protein pairs was based on the following criteria: (i) protein interactions must be extracted from curated databases or experimentally determined (as opposed to computationally

[10]The GO positive annotations were downloaded on January 2021. It is available at `http://release.geneontology.org/2021-01-01/annotations/index.html`.

[11]The negative annotations were downloaded from `https://lab.dessimoz.org/20_not`.

[12]https://www.w3.org/TR/owl2-new-features/#F12:_Punning

[13]The HP was downloaded on October 2022, while the HP annotations were downloaded on November 2021. A link to these versions is no longer available.

determined); (ii) interactions must have a confidence score above 0.950 to retain only high confidence interaction; (iii) each protein must have at least one positive GO association and one negative GO association. The PPI dataset contains 440 proteins, 1024 interacting protein pairs, and another 1024 pairs generated by random negative sampling over the same set of proteins.

6.4.3 Gene-Disease Association Prediction Dataset

Predicting the relation between genes and diseases is essential to understand disease mechanisms and identify potential biomarkers or therapeutic targets [Eilbeck et al., 2017]. However, validating these associations in the wet lab is expensive and time-consuming, which fostered the development of computational approaches to identify the most promising associations to be further validated. Many of these explore biomedical ontologies and KGs [Vanunu et al., 2010; Zakeri et al., 2018; Robinson et al., 2014; Asif et al., 2018; Luo et al., 2019] and some recent approaches even apply KG embedding methods such as DeepWalk [Alshahrani et al., 2017] or OPA2Vec [Smaili et al., 2018b; Nunes et al., 2023].

For GDA prediction, the GO KG, the HP KG, and a GDA dataset are used. Two different ontologies are used to describe each type of entity. Diseases are described under the HP and genes under the GO.

The target relations to predict are extracted from DisGeNET [Piñero et al., 2019], adapting the approach described in Nunes et al. [2023] to consider the following criterion: each gene (or disease) must have at least one positive GO (or HP) association and one negative GO (or HP) association. This resulted in 755 genes, 162 diseases, and 107 gene-disease relations. To create a balanced dataset, random negative examples were sampled over the same genes and diseases.

6.5 Results and Discussion

TrueWalks is compared against ten state-of-the-art KG embedding methods: TransE, TransH, TransR, ComplEx, distMult, DeepWalk, node2vec, metapath2vec, OWL2Vec* and RDF2Vec. TransE, TransH and TransR are representative methods of translational models. ComplEx and distMult are semantic matching methods. They represent a bottom-line baseline with well-known KG embedding methods. DeepWalk and node2vec are undirected random walk-based methods, and OWL2Vec* and RDF2Vec are directed walk-based methods. These methods represent a closer approach to ours, providing a potentially stronger baseline. Each method is run with two different KGs, one with only positive statements and one with both positive and negative statements. In this second KG, the negative statements are declared as an object property, so positive and negative statements appear as two distinct relation types. The size of all the embeddings is 200 dimensions across all experiments, with TrueWalks generating two 100-dimensional vectors, one for the positive statement-based representation and one for the

negative, which are concatenated to produce the final 200-dimensional representation.

Table 6.3: Median precision (Pr), recall (Re), and weighted average F1-score (F1) for PPI and GDA prediction. TrueWalks performance values are italicized/underlined when improvements are statistically significant with p-value < 0.05 for the Wilcoxon test against the positive (Pos)/positive and negative (Pos+Neg) variants of other methods. The best results are in bold.

	Method	**PPI Prediction**			**GDA Prediction**		
		Pr	**Re**	**F1**	**Pr**	**Re**	**F1**
Pos	TransE	0.553	0.546	0.554	0.533	0.538	0.531
	TransH	0.566	0.562	0.566	0.556	0.563	0.548
	TransR	0.620	0.607	0.616	0.594	0.600	0.592
	ComplEx	0.680	0.659	0.679	0.597	0.625	0.598
	distMult	0.765	0.737	0.754	0.585	0.600	0.575
	DeepWalk	0.813	0.836	0.822	0.618	0.646	0.629
	node2vec	0.826	0.741	0.794	0.643	0.616	0.644
	metapath2vec	0.562	0.563	0.561	0.554	0.531	0.549
	OWL2Vec*	0.833	0.806	0.823	0.652	0.656	0.646
	RDF2Vec	0.831	0.826	0.828	0.623	0.625	0.615
Pos + Neg	TransE	0.584	0.582	0.585	0.597	0.585	0.586
	TransH	0.573	0.572	0.570	0.563	0.554	0.554
	TransR	0.722	0.678	0.704	0.633	0.625	0.630
	ComplEx	0.750	0.720	0.740	0.549	0.545	0.545
	distMult	0.813	0.740	0.784	0.530	0.523	0.534
	DeepWalk	0.843	0.834	0.841	0.615	0.646	0.630
	node2vec	0.847	0.734	0.798	0.614	0.594	0.621
	metapath2vec	0.557	0.569	0.558	0.527	0.531	0.522
	OWL2Vec*	0.860	0.812	0.840	0.654	0.600	0.645
	RDF2Vec	0.847	**0.844**	0.845	0.625	**0.661**	0.630
TrueWalks		***0.870***	0.817	*0.846*	***0.667***	0.625	***0.661***
TrueWalksOA		*0.868*	0.836	***0.858***	*0.661*	0.616	*0.654*

6.5.1 Relation Prediction using Machine Learning

To predict the relation between a pair of entities e_1 and e_2 using ML, their vector representations are combined using the binary Hadamard operator to represent the pair: $r(e_1, e_2) = v_{e_1} \times v_{e_2}$.

The pair representations are then fed into a RF algorithm for training using Monte Carlo cross-validation [Xu and Liang, 2001]. Monte Carlo cross-validation is a variation of traditional k-fold cross-validation in which the process of dividing the data into training and testing sets (with β being the proportion of the dataset to include in the test split) is repeated M times. The proposed experiments use Monte Carlo cross-validation with $M = 30$ and $\beta = 0.3$. For each run, the predictive performance is evaluated based on recall, precision and weighted average F1-scores. Statistically significant differences between TrueWalks and the other methods are determined using the non-parametric Wilcoxon test at $p < 0.05$.

Table 6.3 reports the median scores for both PPI and GDA prediction. The top half contains the results of the first experiment where state-of-the-art methods using only the positive statements are compared to TrueWalks (at the bottom) which uses both types. The results reveal that the performance of TrueWalks is significantly better than the other methods, improving both precision and F1-score. An improvement in precision, which is not always accompanied by an increase in recall, confirms the hypothesis that embeddings that consider negative statements produce more accurate representations of entities, which allows a better distinction of true positives from false positives.

A second experiment employs a KG with both negative and positive statements for all methods. The proposed method can accurately distinguish between positive statements and negative statements, as discussed in subsection 6.3.2. For the remaining embedding methods, the negative statements are declared as an object property so that these methods distinguish positive and negative statements as two distinct types of relation. This experiment allows testing whether TrueWalks, which takes into account the positive or negative status of a statement, can improve the performance of methods that handle all statements equally regardless of status.

The bottom half of Table 6.3 shows that both variants of TrueWalks improve on precision and F1-score for both tasks when compared with the state-of-the-art methods using both positive and negative statements. This experiment further shows that the added information given by negative statements generally improves the performance of most KG embedding methods. However, no method surpasses TrueWalks, likely due to its ability to consider the semantic implications of inheritance and walk direction, especially when combined with the order-aware model.

Comparing the two variants of TrueWalks demonstrates that order awareness does not improve performance in most cases. However, TrueWalksOA improves on precision and F1-score for all other state-of-the-art methods. These results are not unexpected since the same effect was observed in other order-aware embedding methods [Portisch and Paulheim, 2021].

Regarding the statistical tests, TrueWalks performance values are italicized/underlined in Table 6.3 when improvements over all other methods are statistically significant, except when comparing TrueWalks with OWL2Vec* for GDA, since in this particular case the improvement is not statistically significant.

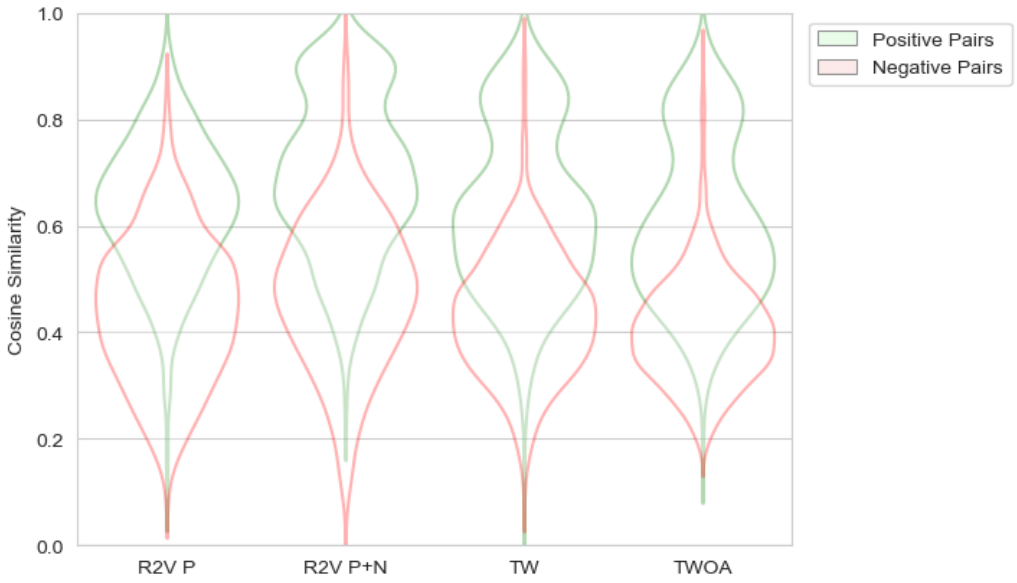


Figure 6.5: Violin plot with embedding similarity obtained with RDF2Vec with positive statements (R2V P), RDF2Vec with both positive and negative statements (R2V P+N), TrueWalks (TW), and TrueWalksOA (TWOA).

6.5.2 Relation Prediction using Semantic Similarity

All methods are also evaluated in PPI prediction using KG embedding-based semantic similarity, computed as the cosine similarity between the vectors of each protein in a pair. Adopting the methodology employed by Kulmanov et al. [2019] and Xiong et al. [2022], for each positive pair e_1 and e_2 in the dataset, the similarity between e_1 and all other entities is computed and the rank of e_2 is identified. The performance was measured using recall at rank 10[14], recall at rank 100, mean rank, and the area under the ROC curve (Table 6.4). Results show that TrueWalksOA achieves the top performance across all metrics, but TrueWalks is bested by RDF2Vec on all metrics except Hits@10, by OWL2Vec* on Hits@100 and by node2vec on Hits@10.

To better understand these results, Figure 6.5 shows the distribution of similarity values for positive and negative pairs. There is a smaller overlap between negative and positive pairs similarities for TrueWalksOA, which indicates that considering both the status of the function assignments and the order of entities in the random walks results in embeddings that are more meaningful semantic representations of proteins. Furthermore, the cosine similarity for nega-

[14]Since the similarity score is computed for all possible pairs to simulate a more realistic scenario where a user is presented with a ranked list of candidate interactions, the task is several degrees more difficult to perform and all KG embedding methods have a recall score of 0 at rank 1. As a result, the results for this metric have been excluded from the analysis.

Table 6.4: Hits@10, Hits@100, mean rank, and area under the receiver operating characteristic curve (AUC ROC) for PPI prediction using cosine similarity obtained with different methods. In bold, the best value for each metric.

	Method	Hits@10	Hits@100	Mean Rank	AUC ROC
Pos	TransE	0.013	0.125	103.934	0.538
	TransH	0.013	0.134	102.703	0.543
	TransR	0.037	0.196	81.916	0.636
	ComplEx	0.080	0.261	64.558	0.689
	distMult	0.112	0.340	46.512	0.803
	DeepWalk	0.125	0.380	35.406	0.847
	node2vec	0.163	0.375	37.275	0.827
	metapath2vec	0.017	0.151	98.445	0.558
	OWL2Vec*	0.152	0.386	33.192	0.860
	RDF2Vec	0.133	0.391	32.419	0.870
Pos + Neg	TransE	0.022	0.161	94.809	0.576
	TransR	0.100	0.274	60.120	0.732
	TransH	0.025	0.174	91.553	0.594
	ComplEx	0.132	0.334	45.268	0.805
	distMult	0.149	0.378	35.351	0.853
	DeepWalk	0.148	0.383	35.365	0.849
	node2vec	**0.166**	0.389	34.305	0.840
	metapath2vec	0.020	0.165	93.374	0.578
	OWL2Vec*	0.160	0.397	32.234	0.869
	RDF2Vec	0.155	0.401	30.281	0.879
TrueWalks		0.161	0.392	32.089	0.869
TrueWalksOA		**0.166**	**0.407**	**28.128**	**0.889**

tive pairs is consistently lower when using both variants of TrueWalks, which supports that the contribution of negative statement-based embeddings is working towards filtering out false positives.

6.6 Conclusions

KG embeddings are increasingly used in biomedical applications such as the prediction of PPIs, GDAs, drug-target relations and drug-drug interactions [Mohamed et al., 2021]. The novel

approach, TrueWalks, was motivated by the fact that existing KG embedding methods are ill-equipped to handle negative statements, despite their recognized importance in biomedical ML tasks [Kulmanov et al., 2021]. TrueWalks incorporates a novel walk-generation method that distinguishes between positive and negative statements and considers the semantic implications of negation in ontology-rich KGs. It generates two separate embeddings, one for each type of statement, enabling a dual representation of entities that can be explored by downstream ML, focusing both on features entities have and those they lack. TrueWalks outperforms representative and state-of-the-art KG embedding approaches in predicting PPIs and GDAs.

TrueWalks is expected to be generalizable to other biomedical applications where negative statements play a decisive role, such as predicting disease-related phenotypes [Xue et al., 2019] or performing differential diagnosis [Köhler et al., 2019].

Algorithm 4 Walk generation for one entity using TrueWalks. The function GET NON VISITED NEIGHBOURS(status) is used to generate the random walks using a depth-first search. It gets the neighbors of a given node that have not yet been visited in previous iterations. If the status is negative (which means that the first step in the walk was made with a negative statement), the neighbors will include all the non-visited neighbors except those connected through subclass statements, and if the status is positive, it will include all the neighbors except those connected through superclass statements.

```
d ← max_depth_walks
w ← max_number_of_walks
ent ← root_entity
function GET TRUEWALKS(ent)
 pos_walks ← GET RANDOM WALKS(ent, positive)
 neg_walks ← GET RANDOM WALKS(ent, negative)
 return pos_walks, neg_walks
function GET RANDOM WALKS(ent, status)
 while len(walks) < w do
  walk ← ent
  depth ← 1
  while depth < d do
   last ← len(walk) == d
   e, v ← GET NEIGHBOR(walk, status, last)
   if e, v == None then
    break
   walk.append(e, v)
   depth + +
  walks.append(walk)
 return walks
function GET NEIGHBOR(walk, status, last)
 n ← GET NON VISITED NEIGHBORS(status)
 if len(n) == 0 & len(walk) > 2 then
  e, v ← walk[−2], walk[−1]
  ADD VISITED NEIGHBORS(e, v, len(walk) − 2, status)
  return None
 e, v ← n[rand()]
 if last then
  ADD VISITED NEIGHBORS(e, v, len(walk), status)
 return e, v
```

Part III

Conclusions

Chapter 7

General Discussion and Conclusions

7.1 General Discussion

This thesis hypothesises that considering the different KG semantic aspects can improve semantic representations. It aims to address three RQs: which are the semantic representations that are more suitable to support supervised learning over KG; how can semantic aspects and ML be explored to improve semantic representations; are the improved semantic representations useful to bioinformatics applications.

The first RQ focuses on identifying the most suitable semantic representations. Two main types of semantic representations are employed throughout the different methodologies: semantic similarity and KG embedding methods. While KG embeddings map entities and relations into a low-dimensional continuous vector space, semantic similarity measures compare entities based on their taxonomic relationships. In Chapter 4, KGsim2vec and its predecessor evoKGsim+ generate semantic similarity-based semantic representations. Namely, evoKGsim+ employs more than ten KG-based semantic similarity measures, including taxonomic semantic similarity measures and KG embedding semantic similarity measures, to generate representations that take into consideration the different semantic aspects. This facilitates comparisons along various axes: comparing different taxonomic measures, comparing different embedding methods to compute similarity, and taxonomic similarity vs. embedding similarity. The results showed that taxonomic semantic similarity measures achieve better results than KG embedding semantic similarity measures. The initial assumption was that embedding similarity could outperform taxonomic similarity since semantic similarity is limited to the taxonomic relations within the ontology. In contrast, embeddings consider all types of relations, and therefore, the embedding representations could be more informative in principle. However, the ability of taxonomic similarity to take into account class specificity may give it the advantage over embedding similarity to estimate similarity more accurately. Besides, taxonomic similarity measures are

usually hand-crafted, providing human-interpretable results for further analysis. In KGsim2vec, the semantic representation is based on a taxonomic semantic similarity measure.

Chapters 5 and 6 introduce representations grounded on embeddings. SEEK is independent of the KG embedding method. Therefore, five representative KG embeddings covering translational, semantic matching and random walk-based methods are employed and compared. The results demonstrated that translational models, such as TransE, performed worse than the other embedding methods. In contrast, path-based methods, such as RDF2vec, achieve better results than the other embedding methods in most experiments. The differences between KG embedding approaches are not unexpected. The methods that emphasise local neighborhoods, such as translational distance approaches, are less suitable since they fail to capture longer-distance relations. This is relevant when most of the information to be processed is represented in the ontology portion of the ontology-rich KG, where taxonomic relations play an essential role. RDF2Vec can capture longer-distance relations, translating into a broader entity representation. In TrueWalks, embeddings are generated using a random walk-based approach. The answer to the first RQ also emerges through evaluating each methodology. For instance, KGsim2vec representations are compared against state-of-the-art KG embedding methods and GNNs.

Regarding the question of how semantic aspects can be explored by ML to improve semantic representations, different approaches are presented that propose distinct definitions of semantic aspects. Two types of semantic aspects are distinguished in this thesis - the class-based and property-based semantic aspects. Chapter 4 and Chapter 5 explore class-based semantic aspects. In Chapter 4, the semantic aspects are defined as subgraphs of the ontology at the same depth. The classes anchoring the subgraphs representing different semantic aspects are defined by three parameters: minimum number of semantic aspects, distance to a leaf class, and minimum coverage. This allows more flexibility, which comes with performance improvements and helps explainability. In Chapter 5, the goal is no longer to generate a representation for each entity but rather a pair representation. Each pair is described concerning the shared semantic aspects. Finally, Chapter 6 explores property-based semantic aspects, where subgraphs are defined by the type of statement describing each entity. An entity is represented by the positive and negative aspects that describe it. The results in all the chapters showed that representations that consider the different semantic aspects, regardless of the definition used, outperform representations considering the entire KG. However, the goal is not solely limited to performance comparison. It extends to evaluating the usage of semantic aspects to increase the explainability of representations for relation prediction. The results also demonstrated that KGsim2vec produces explainable semantic similarity-based semantic representations, while SEEK proposes a new approach to generate explainable embedding-based semantic representations. Figure 7.1 summarizes the definitions of semantic aspects for the different methodologies.

Finally, the third RQ aims to understand whether the improved semantic representations

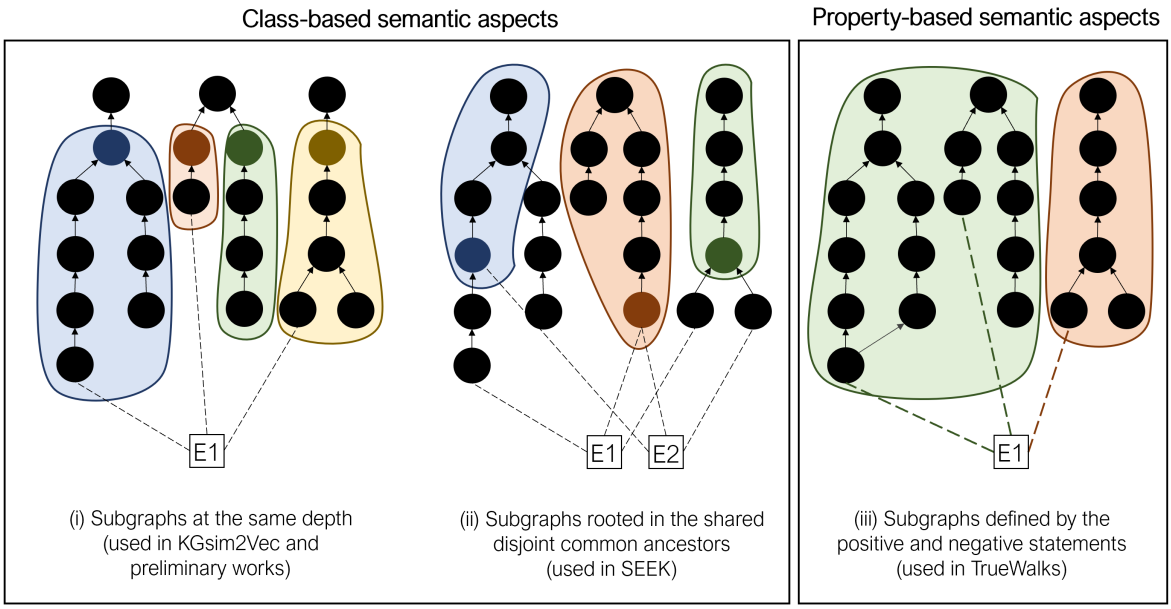


Figure 7.1: Different definitions of semantic aspects for the approaches developed in the context of this Ph.D.

generated by our methodologies are useful for bioinformatics applications. For all methodologies, an evaluation framework focused on biomedical tasks supports the comparative evaluation of representations that consider the whole KG and ML-based semantic representations. Furthermore, extensive effort was invested in constructing benchmark datasets designed to assess the efficacy of different approaches in two biomedical tasks: PPI prediction and GDA prediction. These two tasks are distinct in terms of semantics. In PPI prediction, the goal is to predict a relation between two proteins - entities of the same type described under the same biomedical KG. Conversely, in GDA prediction, the goal shifts towards predicting a relation between a gene and a disease - two distinct entity types described under different KGs. PPI prediction benchmark datasets are generated using the STRING database to evaluate KGsim2vec and SEEK. GDA datasets based on DisGeNET are constructed and used to evaluate SEEK. Additionally, benchmark datasets with negative statements are created to evaluate TrueWalks for PPI, GDA prediction, and disease prediction. It is important to note that all the datasets and associated codes are available online (Github and Zenodo). While these benchmark datasets offered valuable insights about performance, it is essential to acknowledge their limitations, such as challenges in generating accurate negative examples, as discussed in the Chapters 4 and 5. Experimental validation in collaboration with the biochemistry department, described in Chapter 5, was the most robust answer to the third research question. The experimental validation

to test possible new interactions between proteins effectively showed that XAI approaches, such as SEEK, can be used to uncover new knowledge.

In addition to the RQs' answers, the main results of the three methodologies can also be compared to each other, specifically focusing on their predictive performance and explainability. Table 7.1 presents a compilation of the best results for each methodology. Although the three methodologies use the same PPI data and KGs, direct comparisons of these results are not straightforward due to methodological differences. KGsim2vec and SEEK results are derived from 10-fold cross-validation, whereas TrueWalks employ Monte Carlo cross-validation. Furthermore, the results for PPI and GDA are not directly comparable, given the fact that TrueWalks' datasets exclude pairs lacking negative statements, introducing variability in the datasets used for evaluation. TrueWalks models are trained and tested with a limited set of 1024 positive pairs for PPI prediction. In contrast, KGsim2vec and SEEK models trained and tested the ML models with a significantly larger dataset comprising 23,571 positive pairs. This significant dataset size difference could account for the considerable variations in performance observed between these methodologies [Osisanwo et al., 2017].

To conduct a more fair comparison, new experiments for KGsim2vec and SEEK are conducted using the TrueWalks datasets in a Monte-Carlo cross-validation scenario. These experiments are specifically focused on PPI since KGsim2vec focuses on semantic similarity representations that are not suitable for tasks where the entities in the pair are described in different KGs, such as the case of GDA. The results of the new experiments are presented in Table 7.2, showcasing the weighted F-measure scores for the three methodologies using RF for PPI prediction. The analysis results indicate that TrueWalksOA outperforms all other methods, with SEEK following closely behind. However, it is essential to highlight that TrueWalks takes negative statements into account, providing access to more information compared to the other methodologies. It is also interesting to see that SEEK performs better than KGsim2vec. Despite variations in the definition of semantic aspects between these two methodologies, both focus on obtaining a representation of the pair, in contrast to TrueWalks, which follows a paradigm of creating individual entity representations and subsequently combining them to get the pair representation. However, SEEK focuses on embedding-based representations, which appears to confer an advantage in its performance.

Regarding explainability, only KGsim2vec and SEEK address this challenge by generating explainable representations that can be used to explain relation predictions. KGsim2vec focuses on similarity-based representations for various semantic aspects. For transparent models, KGsim2vec explanations are the model itself, whereas, for opaque models, explanations are generated post-hoc techniques such as LIME and LORE. On the other hand, SEEK focuses on embedding-based representations and an explanation is defined as the set of the most relevant shared semantic aspects identified as necessary or sufficient. This explanation can be presented as a chart where sufficient and necessary shared semantic aspects are presented alongside their

Table 7.1: Compilation of the median f-measure for KGsim2vec, SEEK, and TrueWalks using different ML methods for the two biomedical tasks. The utilization of ML methods varied across different methodologies, resulting in empty cells.

Methodology	**PPI Prediction**					**GDA Prediction**		
	GP6x	DT6	RF	XGB	MLP	RF	XGB	MLP
KGsim2vec	0.866	0.906	0.919	0.915	-	-	-	-
SEEK (using RDF2vec)	-	-	0.910	0.915	0.917	0.723	0.719	0.703
SEEK (using OWL2vec*)	-	-	0.919	0.929	0.931	0.737	0.728	0.720
TrueWalks	-	-	0.846	-	-	0.661	-	-
TrueWalksOA	-	-	0.858	-	-	0.654	-	-

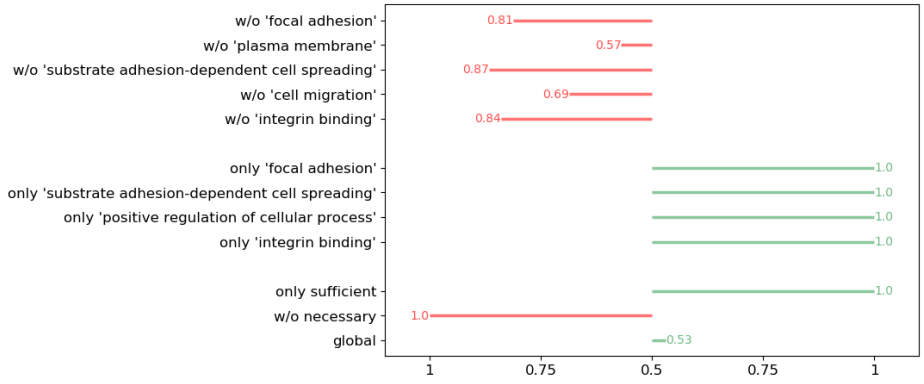
Table 7.2: Median f-measure for KGsim2vec, SEEK, and TrueWalks using RF for PPI prediction.

Methodology	**PPI Prediction**
KGsim2vec	0.850
SEEK (using OWL2vec*)	0.856
TrueWalks	0.846
TrueWalksOA	0.858

impact on the prediction. To compare these two types of explanations, Table 7.3 illustrates the explanations provided by each methodology for a specific protein pair composed of paxillin and integrin α-4. While KGsim2vec, when coupled with transparent methods (such as DT and GP), can provide a global explanation for protein interactions, SEEK explanations are always local and post-hoc. However, the explanation given by SEEK exhibits a higher degree of specificity and a higher alignment with the literature. This is supported by the higher average IC of the ontology classes that appear in the SEEK explanation (0.764 against 0.560 for KGsim2vec). Moreover, according to the literature [Han et al., 2001], integrin α-4 binds tightly to paxillin, leading to increased cell migration and an altered cytoskeletal organization that reduces cell spreading. SEEK explanation captures this by encompassing classes such as "integrin binding" and "focal adhesion."

Table 7.3: Explanations for Paxillin – Integrin α-4

.

Methodology	**Explanation**	**IC**
KGsim2vec (using DT6)	$SS_{\text{metabolic\_process}} \leq 0.6048$ AND $SS_{\text{cellular\_process}} > 0.8535$	0.161
KGsim2vec (using GP6x)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}, SS_{\text{translation\_regulator\_activity}})$	0.454
KGsim2vec (using RF + LIME 8feat)	$(SS_{\text{cellular\_process}} > 0.84, 0.4665)$, $(SS_{\text{intraspecies\_interaction\_between\_organisms}} \leq 0.00, -0.2148)$, $(SS_{\text{molecular\_carrier\_activity}} \leq 0.00, -0.1549)$, $(0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1.00, 0.1463)$, $(SS_{\text{pigmentation}} \leq 0.00, -0.1377)$, $(SS_{\text{metabolic\_process}} \leq 0.00', -0.1253)$, $(SS_{\text{protein\_folding\_chaperone}} \leq 0.00, 0.0864)$, $(SS_{\text{antioxidant\_activity}} \leq 0.00, 0.0843)$	0.560
SEEK		0.764

It is also important to discuss the adaptability of the proposed approaches to the dynamic

nature of KGs. While the presented works may not directly address this aspect, understanding how these methodologies accommodate the evolution of knowledge is crucial. For example, new biomedical entities have been constantly discovered over the years. Methodologies can be broadly categorized into two types: inductive approaches, which learn representations for existing entities in the KG and can make predictions for previously unseen or non-existent entities; and transductive approaches, which learn representations for existing KG entities but are limited to making predictions involving those KG entities. From this perspective, both KGsim2vec and SEEK fall under the inductive category. For KGsim2vec, the representations are based on taxonomic semantic similarity measures, which translate into a mathematical formula that can be applied individually to each protein pair. On the other hand, SEEK represents pairs of entities based on composite representations of ontology classes. Therefore, it is possible to compute a representation for a new pair of proteins just by combining the embeddings of ontology classes they share. The only limitation of SEEK is that the ontology classes shared by the new pairs must already exist in KG. In contrast, TrueWalks is classified as transductive since all the entities must exist in KG. It is not possible to obtain an embedding for an entity that did not exist in the KG at the time of the embedding method's training.

In summary, this thesis demonstrates that generating semantic representations that consider the different KG semantic aspects can achieve more accurate predictions but also generate potentially explainable representations that will help the advancement of biomedical research. As shown, the three RQs were answered successfully, and the hypothesis was confirmed. However, the contributions of this thesis are not only novel methodologies but also a new vision for the ML over KGs. From my point of view, in complex and multi-disciplinary domains, such as the biomedical domain, where a single KG is used to support a wide variety of tasks, it is essential to shift focus from viewing KGs as a whole. To benefit from the full potential promised by KGs for diverse tasks, approaches must capture the different KG semantic aspects. Otherwise, we will be losing fundamental information. That being said, I believe my thesis is a step towards this new vision.

7.2 Research Contributions

This thesis contributed to the advancements in the field of ML over KG for biomedical applications. The chapters of part II are structured around a series of papers that were published during the Ph.D.:

- ***Sousa, R. T.*** *(2020). Evolving meaning for supervised learning in complex biomedical domains using knowledge graphs. In PhD Symposium at Extended Semantic Web Conference, pages 280-290. Springer.*
- ***Sousa, R. T.****, Silva, S., and Pesquita, C. (2021). evoKGsim+: a framework for tailor-*

ing knowledge graph-based similarity for supervised learning. In Extended Semantic Web Conference - Poster and Demo Track, pages 141–146. Springer. (ESWC2021 Best Poster Award)

- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2021). Is there data leakage in protein-protein interaction prediction using knowledge graphs? In International Semantic Web Conference - Poster Demo Track.*
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2022). Explaining protein-protein interaction predictions with genetic programming. In EvoStar - Late-breaking abstracts, page 30.*
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Explainable Representations for Relation Prediction in Knowledge Graphs. In 20th International Conference on Principles of Knowledge Representation and Reasoning.*
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Benchmark datasets for biomedical knowledge graphs with negative statements. In Workshop on Semantic Web solutions for large-scale biomedical data analytics at Extended Semantic Web Conference.*
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Biomedical Knowledge Graph Embeddings with Negative Statements. In International Semantic Web Conference.*
- ***Sousa, R. T.**, Silva, S., and Pesquita, C. (2024). Explaining protein–protein interactions with knowledge graph-based semantic similarity. Computers in Biology and Medicine, 170, 108076.*

7.3 Parallel Contributions

Throughout the Ph.D., some other additional research work was made that, although not directly aligned with the focus of this thesis in considering the different semantic aspects of the KG, made valuable contributions to its overall body of knowledge:

- evoKGsim was mostly developed during the master's degree, but published during the first year of the Ph.D. This approach serves as the foundation for investigating the integration of semantic aspects into the generation of semantic representations. evoKGsim applies GP over a set of semantic similarity features, each based on a semantic aspect of the data, to obtain the best combination for a given supervised learning task. The approach was evaluated on several benchmark datasets for PPI prediction using the GO as the KG to support taxonomic semantic similarity, and it outperformed competing strategies, including manually selected combinations of semantic aspects emulating expert knowledge. evoKGsim was also able to learn species-agnostic models with different combinations of

species for training and testing, effectively addressing the limitations of predicting protein-protein interactions for species with fewer known interactions.

***Sousa, R. T.**, Silva, S., and Pesquita, C. (2020). Evolving knowledge graph similarity for supervised learning in complex biomedical domains. BMC Bioinformatics, 21:1–19.*

- A collection of 21 benchmark datasets that aim at circumventing the difficulties in building benchmarks for large biomedical KGs by exploiting proxies for biomedical entity similarity. These datasets include data from two successful biomedical ontologies, GO and HP, and explore proxy similarities calculated based on protein sequence similarity, protein function family similarity, protein-protein interactions and phenotype-based gene similarity. The 21 benchmark datasets are available online[1] and explore four objective similarities based on protein and gene properties. This resulted in one gene dataset and 16 protein datasets, divided by species, level of annotation completion and objective similarity, and four additional datasets, combining all species' protein pairs in the same objective similarity group. Datasets range from 264 individual proteins and 428 pairs to 27 thousand proteins and 158 thousand pairs.

  *Cardoso, C., **Sousa, R. T.**, Köhler, S., and Pesquita, C. (2020). A collection of benchmark data sets for knowledge graph-based similarity in the biomedical domain. In Extended Semantic Web Conference - Poster and Demo Track, pages 50-55. Springer. (ESWC2020 Best Poster Award)*

  *Cardoso, C., **Sousa, R. T.**, Köhler, S., and Pesquita, C. (2020). A collection of benchmark data sets for knowledge graph-based similarity in the biomedical domain. Database, baaa078.*

- The supervised semantic similarity toolkit that uses supervised ML to tailor aspect-oriented semantic similarity measures to fit a particular view on biological similarity or relatedness. The supervised semantic similarity toolkit tackles the challenge of tailoring semantic similarity to a viewpoint of the domain or a particular use case in an automated fashion. The supervised semantic similarity toolkit[2] learns the similarity between entities represented in KGs optimized towards a specific objective similarity. An objective similarity is a similarity metric that compares two individuals based on an objective representation of a specific property (e.g. two proteins represented by their amino acid sequences can be compared through their sequence similarity score). This tailoring is achieved by considering the similarities for different semantic aspects instead of the static semantic similarity measures that does not consider additional external input or tailoring to a specific objective similarity.

[1] https://github.com/liseda-lab/kgsim-benchmark
[2] https://github.com/liseda-lab/Supervised-SS

The supervised semantic similarity toolkit includes three steps. The first step involves identifying the semantic aspects describing the KG entities. It takes as pre-defined semantic aspects the subgraphs when the KGs have multiple roots or the subgraphs rooted in the classes at a distance of one from the KG root class (such as GO). The next step is representing each instance (i.e., a pair of KG entities) according to static KG-based similarities computed for each semantic aspect. For the computation of KG-based similarities for each semantic aspect, the supervised semantic similarity toolkit employs several KG-based semantic similarity measures. The third step in this approach is to select the objective similarity for which the similarity will be tailored. The last step is employing an ML method to learn a supervised semantic similarity. A supervised regression algorithm computes the aggregation function. Therefore, each regressor receives the similarity values for each semantic aspect as input features (independent variables) and an objective similarity value as the expected output (dependent variable) and returns an aggregated similarity score as the predicted output.

The supervised semantic similarity toolkit is evaluated using 21 benchmark datasets and two different KGs. The objective similarities correspond to widely employed biological similarity metrics - PPI similarity, protein function similarity, protein sequence similarity and phenotype-based gene similarity - and were used to train and test the supervised models. The results show that the proposed supervised similarity model achieves significant improvements over classical taxonomic semantic similarity measures as well as the more recently proposed KG embedding-based measures. Regarding the ML models, both transparent and opaque ML algorithms are evaluated and their performance and interoperability. While the opaque models produced predictions with higher accuracy in the experiments, the supervised similarity obtained using LR and GP still showed improvement over the baseline models and allowed for an insightful analysis. This highlights the need to explore the trade-off between performance and interpretability. Finally, the results also demonstrate that tailoring a semantic similarity measure to the appropriate use case has a marked influence on predictive performance based on semantic similarity measure, as evidenced by the case study on PPI prediction.

***Sousa, R. T.**, Silva, S., and Pesquita, C. (2023). Supervised biomedical semantic similarity. IEEE Access, 11:60635-60645.*

***Sousa, R. T.**, Silva, S., and Pesquita, C. (2022). The supervised semantic similarity toolkit. In Proceedings of Extended Semantic Web Conference - Poster and Demo Track, pages 42–46. Springer.*

***Sousa, R. T.**, Silva, S., and Pesquita, C. (2022c). Towards supervised biomedical semantic similarity. In Workshop on Semantic Web solutions for large-scale biomedical data analytics at Extended Semantic Web Conference.*)

- A novel approach to predict GDAs using semantic representations based on KG embeddings over multiple ontologies, GO and HP, linked by logical definitions and compound ontology mappings. The HP also includes logical definitions that provide a definition of its classes in terms of a composition of classes from different ontologies with complex semantic relations, facilitating interoperability and data integration. Logical definitions can then be explored to bridge domains and contextualize relations between different entities, such as genes and diseases. Regarding the compound ontology mappings, they are generated using AML-Compound [Oliveira and Pesquita, 2018], a variant of the AgreementMakerLight ontology matching system that is able to retrieve relations between ontology classes. The GDA prediction results showed that considering semantically-rich KGs can significantly improve GDA prediction and that different KG embeddings methods benefit more from distinct types of semantic richness. This work paves the way for exploring alternative ontologies that can benefit from multiple perspectives on the data.

  *Nunes, S., **Sousa, R. T.**, Pesquita, C. (2023). Multi-domain knowledge graph embeddings for gene-disease association prediction. Journal of Biomedical Semantics, 14(1), 1-12.*

- A position paper about explaining AI predictions of disease progression with semantic similarity. This work builds on the recently developed Brainteaser Ontology [Bettin et al., 2021] and explores how this ontology coupled with the rich panorama of more general biomedical ontologies can support semantic similarity-based explanations for patient end-stage event predictions that build upon the contextualization of patient data and AI predictions. The underlying idea is that the prediction for one patient can be explained by considering aspect-oriented semantic similarity with other relevant patients based on the most important features used by ML approaches or selected by users. A key aspect is the selection of the most relevant patients to explain an event prediction. Three types of explanatory patients are defined: the most similar patients exhibiting the same outcome; the least similar patients with the same outcome; and the most similar patients exhibiting a different outcome.

  Nunes, S., **Sousa, R. T.***, Serrano, F.*, Branco, R., Soares, D. F., Martins, A. S., Auletta,E., Castanho, E. N., Madeira, S. C., Aidos, H., Pesquita, C. (2022). Explaining artificial intelligence predictions of disease progression with semantic similarity. In CLEF (pp. 1613-0073).*

  *Branco, R., Soares, D. F., Martins, A. S., Auletta, E., Castanho, E. N., Nunes, S., Serrano, F., **Sousa, R. T.**, Pesquita, C., Madeira, S. C., Aidos, H. (2022). Hierarchical modelling for ALS prognosis: predicting the progression towards critical events. In CLEF (pp. 1613-0073).*

- A novel approach - DKI - for domain knowledge injection into GNNs that explores biomed-

ical KGs and KG embeddings to produce meaningful numerical features for biomedical entities. The approach was evaluated using ten different GNN approaches and five different KG embedding methods using two tasks provided by the well-known Open Graph Benchmark [Hu et al., 2020] and the Human Reference Interactome [Luck et al., 2020]. The results showed that domain knowledge injection can amply improve the performance of GNNs over PPI networks, in both node classification and link prediction tasks. The obtained results surpassed all existing entries in the leaderboards of both ogbn-proteins (protein function prediction) and ogbl-ppa (PPI prediction) tasks and demonstrate that powerful node features can be generated by exploring external sources of knowledge.

*Balbi, L., **Sousa, R. T.**, Cotovio, P., Pesquita, C. Injecting domain knowledge into graph neural networks for protein-protein interactions. (submitted at BMC Bioinformatics)*

7.4 Limitations and Future Work

This thesis advances the state-of-the-art by presenting three methodologies - KGsim2vec, SEEK, and TrueWalks - that improve semantic representations of KG entities in terms of performance and explainability by considering the different semantic aspects of the KG. Each methodology is characterized by its definition of semantic aspects, the employed semantic representations, the machine learning techniques used for supervised learning, and the biomedical applications in which it is evaluated. However, some limitations leave room for future research and improvement.

SEEK generates explanations by adopting a perturbation-inspired approach where, for each relation to be explained, multiple representations are generated that differ by the presence or absence of a shared semantic aspect. However, the impact of removing or including more than one shared semantic aspect still needs to be explored. Therefore, the next step is to improve explanations by investigating the minimal set of shared semantic aspects required to explain a relation adequately. Additionally, there is room for improvement in the evaluation process. This thesis aims to investigate how semantic aspects can be explored, and it is essential to clarify that explainability is not a goal by itself. In the context of SEEK, explainability emerges as a consequence of considering semantic aspects, and consequently, the evaluation of SEEK explanations is not the main focus of this work. For future work, SEEK can be evaluated by gathering expert feedback to confirm that SEEK explanations are effective and useful. Another promising avenue for future work lies in creating a user-friendly graphical interface designed specifically to the needs of experts, such as a webpage, to enhance the accessibility and usability of SEEK.

TrueWalks also has some limitations. It uses explicit negative statements to produce entity representations considering both existing and lacking attributes. TrueWalks can capture the similarity between two entities if they share negative or positive statements. However, TrueWalks

do not consider how opposite statements can impact the dissimilarity of entities. A possible direction for future research involves exploring counter-fitting approaches, such as those proposed for language embeddings [Mrksic et al., 2016]. Counter-fitting approaches or language embeddings serve as post-processors that fine-tune pre-trained word embeddings by refining pairwise distances according to the antonymy and synonymy constraints. The idea could then be to use these techniques to push away embeddings of entities described with opposite statements.

Stepping beyond the limitations of each specific methodology, the goal of this Ph.D. project is to propose methodologies especially tailored to address the characteristics of biomedical KGs. Although most evaluations focus on PPI or GDA prediction, many other biomedical tasks could benefit from the proposed methodologies. For example, TrueWalks can be generalizable to other biomedical applications where negative statements play a decisive role, such as predicting disease-related phenotypes or performing differential diagnoses. SEEK and KGsim2vec can also be applied to any biomedical relation prediction task. These encompass, for instance, predicting drug-drug interactions, drug-target associations or drug-disease associations. In fact, the application of the proposed methodologies extends to any scenario where relations need to be predicted, and there is an ontology-rich KG.

It would be impossible to finish this future work section without mentioning the recent advances in graph mining and the appearance of several approaches based on GNNs, as elucidated in Chapter 3. I believe GNN and other GNN-based methods are not yet tailored to handle the semantic richness of ontology-rich KGs, as evidenced in the literature [Zhang et al., 2021d; Zhou et al., 2020; Bourgeais et al., 2022] and the results presented in Chapter 4. One of the challenges arises from the requirement of node features in these methods, whereas ontology-rich KGs typically provide only labels. Nonetheless, these recent advances are promising, opening new opportunities. Exploring the different KG semantic aspects with GNNs could pave the way for a deeper understanding of ontology-rich KGs.

References

Abdelaziz, I., Fokoue, A., Hassanzadeh, O., Zhang, P., and Sadoghi, M. (2017). Large-scale structural and textual similarity-based mining of knowledge graph to predict drug–drug interactions. *Journal of Web Semantics*, 44:104 – 117. Industry and In-use Applications of Semantic Technologies. 55, 68

Alshahrani, M. and Hoehndorf, R. (2018). Semantic disease gene embeddings (SmuDGE): phenotype-based disease gene prioritization without phenotypes. *Bioinformatics*, 34(17):i901–i907. 55

Alshahrani, M., Khan, M. A., Maddouri, O., Kinjo, A. R., Queralt-Rosinach, N., and Hoehndorf, R. (2017). Neuro-symbolic representation learning on biological knowledge graphs. *Bioinformatics*, 33(17):2723–2730. 92, 99, 122

Alshahrani, M., Thafar, M. A., and Essack, M. (2021). Application and evaluation of knowledge graph embeddings in biomedical data. *PeerJ Computer Science*, 7:e341. 2, 64

Althubaiti, S., Karwath, A., Dallol, A., Noor, A., Alkhayyat, S. S., Alwassia, R., Mineta, K., Gojobori, T., Beggs, A. D., Schofield, P. N., et al. (2019). Ontology-based prediction of cancer driver genes. *Scientific Reports*, 9(1):17405. 53

Arnaout, H., Razniewski, S., Weikum, G., and Pan, J. Z. (2021a). Negative statements considered useful. *Journal of Web Semantics*, 71:100661. 111, 114, 115

Arnaout, H., Razniewski, S., Weikum, G., and Pan, J. Z. (2021b). Wikinegata: a knowledge base with interesting negative statements. *PVLDB (Proceedings of the VLDB Endowment)*, 14(12):2807–2810. 115

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115. 36

Asif, M., Martiniano, H. F. M. C. M., Vicente, A. M., and Couto, F. M. (2018). Identifying disease genes using machine learning and gene functional similarities, assessed through Gene Ontology. *PLOS ONE*, 13(12):1–15. 2, 58, 68, 92, 122

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). DBpedia: A Nucleus for a Web of Open Data. In *Proceedings of the 6th International The Semantic Web and 2nd Asian Conference on Asian Semantic Web Conference*, page 722–735, Berlin, Heidelberg. Springer-Verlag. 11

Baader, F., Horrocks, I., and Sattler, U. (2004). *Description logics*. Springer. 13

Balasubramanian, M. and Schwartz, E. L. (2002). The isomap algorithm and topological stability. *Science*, 295(5552):7. 16, 17, 18

Bandyopadhyay, S. and Mallick, K. (2017). A New Feature Vector Based on Gene Ontology Terms for Protein-Protein Interaction Prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 14(4):762–770. 1

Bang, D., Lim, S., Lee, S., and Kim, S. (2023). Biomedical knowledge graph learning for drug repurposing by extending guilt-by-association to multiple layers. *Nature Communications*, 14(1):3570. 44, 51, 61, 62, 63

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115. 2, 34, 36, 67, 73, 91

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The Semantic Web. *Scientific American*, 284(5):34–43. 11

Bettin, M., Di Nunzio, G. M., Dosso, D., Faggioli, G., Ferro, N., Marchetti, N., and Silvello, G. (2021). Deliverable 9.1 – Project ontology and terminology, including data mapper and RDF graph builder. 141

Betz, P., Meilicke, C., and Stuckenschmidt, H. (2022). Adversarial Explanations for Knowledge Graph Embeddings. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, Darmstadt, Vienna. International Joint Conferences on Artificial Intelligence Organization. 94

Bianchini, M., Gori, M., and Scarselli, F. (2005). Inside pagerank. *ACM Transactions on Internet Technology*, 5(1):92–128. 20

Binder, J., Ursu, O., Bologa, C., Jiang, S., Maphis, N., Dadras, S., Chisholm, D., Weick, J., Myers, O., Kumar, P., et al. (2022). Machine learning prediction and tau-based screening identifies potential Alzheimer's disease genes relevant to immunity. *Communications Biology*, 5(1):125. 2, 43, 53, 62

Biswas, S., Mitra, P., and Rao, K. S. (2019). Relation prediction of co-morbid diseases using knowledge graph completion. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 18(2):708–717. 41, 54

Bizer, C., Heath, T., and Berners-Lee, T. (2011). Linked data: The story so far. In *Semantic services, interoperability and web applications: emerging concepts*, pages 205–227. IGI global. 11

Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, page 1247–1250, New York, USA. Association for Computing Machinery. 11

Bonner, S., Barrett, I. P., Ye, C., Swiers, R., Engkvist, O., Hoyt, C. T., and Hamilton, W. L. (2022). Understanding the performance of knowledge graph embeddings in drug discovery. *Artificial Intelligence in the Life Sciences*, 2:100036. 43, 52

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013). Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems 26*. Curran Associates, Inc. 16, 18, 19, 34, 37, 78, 101

Borgwardt, K. M. and Kriegel, H.-P. (2005). Shortest-Path Kernels on Graphs. In *Proceedings of the 5th IEEE International Conference on Data Mining*, pages 74–81, USA. IEEE Computer Society. 16

Borgwardt, K. M., Ong, C. S., Schönauer, S., Vishwanathan, S. V. N., Smola, A. J., and Kriegel, H.-P. (2005). Protein function prediction via graph kernels. *Bioinformatics*, 21(suppl1):i47–i56. 37

Bourgeais, V., Zehraoui, F., Ben Hamdoune, M., and Hanczar, B. (2021). Deep GONet: self-explainable deep neural network based on Gene Ontology for phenotype prediction from gene expression data. *BMC Bioinformatics*, 22(10):1–25. 43, 59, 62, 63

Bourgeais, V., Zehraoui, F., and Hanczar, B. (2022). GraphGONet: a self-explaining neural network encapsulating the Gene Ontology graph for phenotype prediction on gene expression. *Bioinformatics*, 38(9). 143

Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32. 30, 101

Breit, A., Waltersdorfer, L., Ekaputra, F. J., Sabou, M., Ekelhart, A., Iana, A., Paulheim, H., Portisch, J., Revenko, A., Teije, A. t., et al. (2023). Combining machine learning and semantic web: A systematic mapping study. *ACM Computing Surveys*. 37

Bresso, E., Monnin, P., Bousquet, C., Calvier, F.-E., Ndiaye, N.-C., Petitpain, N., Smaïl-Tabbone, M., and Coulet, A. (2021). Investigating ADR mechanisms with Explainable AI: a feasibility study with knowledge graph mining. *BMC Medical Informatics and Decision Making*, 21(1):171. 43, 62, 63

Brown, P. J. and Zidek, J. V. (1980). Adaptive Multivariate Ridge Regression. *Annals of Statistics*, 8(1):64–74. 28

Bruna Estrach, J., Zaremba, W., Szlam, A., and LeCun, Y. (2014). Spectral networks and deep locally connected networks on graphs. In *Proceedings of the 2nd International Conference on Learning Representations*, USA. IEEE Computer Society. 32

Cai, H., Zheng, V. W., and Chang, K. C. (2018). A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 30(9):1616–1637. 16, 17

Canato, S., Santos, J. D., Carvalho, A. S., Aloria, K., Amaral, M. D., Matthiesen, R., Falcao, A. O., and Farinha, C. M. (2018). Proteomic interaction profiling reveals KIFC1 as a factor involved in early targeting of F508del-CFTR to degradation. *Cellular and Molecular Life Sciences*, 75:4495–4509. 106

Cao, S., Lu, W., and Xu, Q. (2016). Deep Neural Networks for Learning Graph Representations. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 17, 18, 20

Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka, E. R., and Mitchell, T. M. (2010). Toward an Architecture for Never-Ending Language Learning. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 11

Carvalho, R. M., Oliveira, D., and Pesquita, C. (2023). Knowledge Graph Embeddings for ICU readmission prediction. *BMC Medical Informatics and Decision Making*, 23(1):12. 44, 54

Celebi, R., Uyar, H., Yasar, E., Gumus, O., Dikenelli, O., and Dumontier, M. (2019). Evaluation of knowledge graph embedding approaches for drug-drug interaction prediction in realistic settings. *BMC Bioinformatics*, 20(1):726. 2, 41, 49, 93

Chai, X. (2020). Diagnosis method of thyroid disease combining knowledge graph and deep learning. *IEEE Access*, 8:149787–149795. 42, 49

Chari, S., Gruen, D. M., Seneviratne, O., and McGuinness, D. L. (2020). Foundations of Explainable Knowledge-Enabled Systems. In *Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges*, pages 23–48. IOS Press. 35, 68

Che, M., Yao, K., Che, C., Cao, Z., and Kong, F. (2021). Knowledge-graph-based drug repositioning against COVID-19 by graph convolutional network with attention mechanism. *Future Internet*, 13(1):13. 43, 59

Chen, J., Althagafi, A., and Hoehndorf, R. (2020a). Predicting candidate genes from phenotypes, functions and anatomical site of expression. *Bioinformatics*, 37(6):853–860. 54

Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O. M., Antonyrajah, D., and Horrocks, I. (2021a). Owl2vec*: Embedding of owl ontologies. *Machine Learning*, 110(7):1813–1845. 17, 18, 22, 26, 78, 79, 101

Chen, K.-H., Wang, T.-F., and Hu, Y.-J. (2019). Protein-protein interaction prediction using a hybrid feature representation and a stacked generalization scheme. *BMC Bioinformatics*, 20(1):308. 2, 16, 41, 57, 67, 68, 70, 92

Chen, T. and Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA. Association for Computing Machinery. 31, 101

Chen, X., Jia, S., and Xiang, Y. (2020b). A review: Knowledge reasoning over knowledge graph. *Expert Systems with Applications*, 141:112948. 38

Chen, Y., Ma, T., Yang, X., Wang, J., Song, B., and Zeng, X. (2021b). MUFFIN: multi-scale feature fusion for drug–drug interaction prediction. *Bioinformatics*, 37(17):2651–2658. 43, 49, 62

Cheng, X., Xiao, X., and Chou, K.-C. (2018). pLoc-mGneg: Predict subcellular localization of Gram-negative bacterial proteins by deep gene ontology learning via general PseAAC. *Genomics*, 110(4):231–239. 54

Cochez, M., Ristoski, P., Ponzetto, S. P., and Paulheim, H. (2017). Global RDF vector space embeddings. In *Proceedings of 16th International Semantic Web Conference*, pages 190–207, Cham, Switzerland. Springer International Publishing. 22

Consortium, G. (2021). The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Research*, 49(D1):D325–D334. 14, 75, 76, 100, 115, 121

Couto, F. M. and Silva, M. J. (2011). Disjunctive shared information between ontology concepts: application to Gene Ontology. *Journal of Biomedical Semantics*, 2:1–16. 96

Cui, H., Lu, Z., Li, P., and Yang, C. (2022). On positional and structural node features for graph neural networks on non-attributed graphs. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 3898–3902. 2

Dai, Y., Guo, C., Guo, W., and Eickhoff, C. (2021). Drug–drug interaction prediction with Wasserstein Adversarial Autoencoder-based knowledge graph embeddings. *Briefings in Bioinformatics*, 22(4):bbaa256. 42, 50

Daluwatumulle, G., Wijesinghe, R., and Weerasinghe, R. (2022). In Silico Drug Repurposing using Knowledge Graph Embeddings for Alzheimer's Disease. In *Proceedings of the 9th International Conference on Bioinformatics Research and Applications*, New York, USA. Association for Computing Machinery. 43, 56

d'Amato, C., Masella, P., and Fanizzi, N. (2022). An Approach Based on Semantic Similarity to Explaining Link Predictions on Knowledge Graphs. In *Proeccedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, WI-IAT '21, page 170–177, New York, USA. Association for Computing Machinery. 34

Defferrard, M., Bresson, X., and Vandergheynst, P. (2016). Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Red Hook, USA. Curran Associates Inc. 32

Degtyarenko, K., De Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcántara, R., Darsow, M., Guedj, M., and Ashburner, M. (2007). ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic Acids Research*, 36(suppl1):D344–D350. 14

Denison, D. G. T., Mallick, B. K., and Smith, A. F. M. (1998). A Bayesian CART algorithm. *Biometrika*, 85(2):363–377. 73

Dong, Y., Chawla, N. V., and Swami, A. (2017). metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA. Association for Computing Machinery. 17, 18, 21

Durán, J. M. (2021). Dissecting scientific explanation in AI (sXAI): A case for medicine and healthcare. *Artificial Intelligence*, 297:103498. 35

Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., and Adams, R. P. (2015). Convolutional Networks on Graphs for Learning Molecular Fingerprints. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, Cambridge, USA. MIT Press. 32

Ehrlinger, L. and Wöß, W. (2016). Towards a Definition of Knowledge Graphs. In *Proceedings of the 12th International Conference on Semantic Systems*, volume 48, New York, USA. Association for Computing Machinery. 1, 12, 13

Eilbeck, K., Quinlan, A., and Yandell, M. (2017). Settling the score: variant prioritization and Mendelian disease. *Nature Reviews Genetics*, 18(10):599–612. 99, 122

Fan, K., Guan, Y., and Zhang, Y. (2020). Graph2GO: a multi-modal attributed network embedding method for inferring protein functions. *GigaScience*, 9(8):giaa081. 62

Fang, Y., Wang, H., Wang, L., Di, R., and Song, Y. (2019). Diagnosis of COPD based on a knowledge graph and integrated model. *IEEE Access*, 7:46004–46013. 41, 49

Färber, M., Bartscherer, F., Menne, C., and Rettinger, A. (2018). Linked data quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web*, 9(1):77–129. 13

Flouris, G., Huang, Z., Pan, J. Z., Plexousakis, D., and Wache, H. (2006). Inconsistencies, negations and changes in ontologies. In *Proceedings of the 21st National Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 111

Fouss, F., Pirotte, A., Renders, J.-M., and Saerens, M. (2007). Random-Walk Computation of Similarities between Nodes of a Graph with Application to Collaborative Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):355–369. 23

Fu, G., Wang, J., Yang, B., and Yu, G. (2016). NegGOA: negative GO annotations selection using ontology structure. *Bioinformatics*, 32(19):2996–3004. 115

Gad-Elrab, M. H., Stepanova, D., Tran, T.-K., Adel, H., and Weikum, G. (2020). Excut: Explainable embedding-based clustering over knowledge graphs. In *Proceedings of 19th International Semantic Web Conference*, Berlin, Heidelberg. Springer-Verlag. 94

Gao, K. Y., Fokoue, A., Luo, H., Iyengar, A., Dey, S., and Zhang, P. (2018). Interpretable drug target prediction using deep neural representation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 58

Gao, Z., Ding, P., and Xu, R. (2022a). KG-Predict: A knowledge graph computational framework for drug repurposing. *Journal of Biomedical Informatics*, 132:104133. 44, 59, 60

Gao, Z., Pan, Y., Ding, P., and Xu, R. (2022b). A knowledge graph-based disease-gene prediction system using multi-relational graph convolution networks. In *AMIA Annual Symposium Proceedings*, Bethesda, USA. American Medical Informatics Association. 44, 60

Gärtner, T., Flach, P., and Wrobel, S. (2003). "On Graph Kernels: Hardness Results and Efficient Alternatives". In Schölkopf, B. and Warmuth, M. K., editors, *Learning Theory and Kernel Machines*, pages 129–143, Berlin, Heidelberg. Springer-Verlag. 37

Gaudet, P. and Dessimoz, C. (2017). Gene Ontology: pitfalls, biases, and remedies. In *The Gene Ontology Handbook*, pages 189–205. Humana Press. 114

Gavali, S., Ross, K., Chen, C., Cowart, J., and Wu, C. H. (2022). A knowledge graph representation learning approach to predict novel kinase–substrate interactions. *Molecular Omics*, 18(9):853–864. 44, 54, 61, 62, 63

Ge, Y., Tian, T., Huang, S., Wan, F., Li, J., Li, S., Wang, X., Yang, H., Hong, L., Wu, N., Yuan, E., Luo, Y., Cheng, L., Hu, C., Lei, Y., Shu, H., Feng, X., Jiang, Z., Wu, Y., Chi, Y., Guo, X., Cui, L., Xiao, L., Li, Z., Yang, C., Miao, Z., Chen, L., Li, H., Zeng, H., Zhao, D., Zhu, F., Shen, X., and Zeng, J. (2021). An integrative drug repositioning framework discovered a potential therapeutic agent targeting COVID-19. *Signal transduction and targeted therapy*, 6(1):165. 59

Gesese, G. A., Biswas, R., Alam, M., and Sack, H. (2021). A survey on knowledge graph embeddings with literals: Which model links better literal-ly? *Semantic Web*, 12(4):617–647. 17

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., and Kagal, L. (2018). Explaining explanations: An overview of interpretability of machine learning. In *Proceedings of the 5th IEEE International Conference on Data Science and Advanced Analytics*. IEEE. 34

Gilvary, C., Elkhader, J., Madhukar, N., Henchcliffe, C., Goncalves, M. D., and Elemento, O. (2020). A machine learning and network framework to discover new indications for small molecules. *PLOS Computational Biology*, 16(8):e1008098. 41, 56, 61, 63

Glikson, E. and Woolley, A. W. (2020). Human trust in artificial intelligence: Review of empirical research. *Academy of Management Annals*, 14(2):627–660. 38

Gonzales, C., Lee, E. H., Lee, K. L. K., Tang, J., and Miret, S. (2022). Hyperparameter Optimization of Graph Neural Networks for the OpenCatalyst Dataset: A Case Study. In *Proceedings of AI for Accelerated Materials Design NeurIPS 2022 Workshop*. 60

Grau, B. C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., and Sattler, U. (2008). OWL 2: The next step for OWL. *Journal of Web Semantics*, 6(4):309–322. 14, 114

Grover, A. and Leskovec, J. (2016). node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA. Association for Computing Machinery. 17, 18, 21

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., and Chen, T. (2018). Recent Advances in Convolutional Neural Networks. *Pattern Recognition*, 77(C):354–377. 32

Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., and Giannotti, F. (2018a). Local rule-based explanations of black box decision systems. *arXiv preprint arXiv:1805.10820*. 74

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. (2018b). A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51(5). 92

Gunning, D., Stefik, M., Choi, J., Miller, T., Stumpf, S., and Yang, G.-Z. (2019). XAI—Explainable artificial intelligence. *Science Robotics*, 4(37):eaay7120. 34

Hamilton, W. L. (2020). Graph representation learning. *Synthesis Lectures on Artifical Intelligence and Machine Learning*, 14(3):1–159. 1

Han, J., Liu, S., Rose, D. M., Schlaepfer, D. D., McDonald, H., and Ginsberg, M. H. (2001). Phosphorylation of the Integrin alpha-4 Cytoplasmic Domain Regulates Paxillin Binding. *Journal of Biological Chemistry*, 276(44):40903–40909. 104, 136

Hao, X., Chen, Q., Pan, H., Qiu, J., Zhang, Y., Yu, Q., Han, Z., and Du, X. (2023). Enhancing drug–drug interaction prediction by three-way decision and knowledge graph embedding. *Granular Computing*, 8(1):67–76. 44, 50

Harispe, S., Ranwez, S., Janaqi, S., and Montmain, J. (2015). *Semantic similarity from natural language and ontology analysis*. Morgan Claypool Publishers. 4, 23

Hayes, M. J., Rescher, U., Gerke, V., and Moss, S. E. (2004). Annexin–actin interactions. *Traffic*, 5(8):571–576. 82

He, S., Liu, K., Ji, G., and Zhao, J. (2015). Learning to Represent Knowledge Graphs with Gaussian Embedding. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, New York, USA. Association for Computing Machinery. 17, 18, 19

Hinnerichs, T. and Hoehndorf, R. (2021). DTI-Voodoo: machine learning over interaction networks and ontology-based background knowledge predicts drug–target interactions. *Bioinformatics*, 37(24):4835–4843. 51

Hitzler, P., Janowicz, K., and Lecue, F. (2020). On the Role of Knowledge Graphs in Explainable AI. *Semantic Web*, 11(1):41–51. 37

Hoehndorf, R., Dumontier, M., and Gkoutos, G. V. (2013). Evaluation of research in biomedical ontologies. *Briefings in Bioinformatics*, 14(6):696–712. 14

Hoehndorf, R., Schofield, P. N., and Gkoutos, G. V. (2011). PhenomeNET: a whole-phenome approach to disease gene discovery. *Nucleic Acids Research*, 39(18):e119. 68, 92

Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G. D., Gutierrez, C., Kirrane, S., Gayo, J. E. L., Navigli, R., Neumaier, S., Ngomo, A.-C. N., Polleres, A., Rashid, S. M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., and Zimmermann, A. (2021). Knowledge graphs. *ACM Computing Surveys*, 54(4):1–37. 1, 2, 12, 13

Holzinger, A., Biemann, C., Pattichis, C. S., and Kell, D. B. (2017). What do we need to build explainable AI systems for the medical domain? *arXiv preprint arXiv:1712.09923*. 34, 37, 67

Hu, J., Lepore, R., Dobson, R. J., Al-Chalabi, A., M. Bean, D., and Iacoangeli, A. (2021a). DGLinker: flexible knowledge-graph prediction of disease–gene associations. *Nucleic Acids Research*, 49(W1):W153–W161. 42, 53, 61, 63

Hu, L., Wang, X., Huang, Y.-A., Hu, P., and You, Z.-H. (2021b). A survey on computational models for predicting protein–protein interactions. *Briefings in Bioinformatics*, 22(5):bbab036. 99, 121

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. (2020). Open graph benchmark: Datasets for machine learning on graphs. *Advances in Neural Information Processing Systems*, 33:22118–22133. 142

Huang, Q., Yamada, M., Tian, Y., Singh, D., and Chang, Y. (2022). Graphlime: Local interpretable model explanations for graph neural networks. *IEEE Transactions on Knowledge and Data Engineering*. 35

Huntley, R. P., Sawford, T., Mutowo-Meullenet, P., Shypitsyna, A., Bonilla, C., Martin, M. J., and O'Donovan, C. (2015). The GOA database: gene ontology annotation updates for 2015. *Nucleic Acids Research*, 43(D1):D1057–D1063. 14, 75, 76, 100

Ieremie, Ioan and Ewing, Rob M and Niranjan, Mahesan (2022). TransformerGO: predicting protein–protein interactions by modelling the attention between sets of gene ontology terms. *Bioinformatics*, 38(8):2269–2277. 16, 47, 61, 67, 68, 71, 92

Ioannidis, V. N., Marques, A. G., and Giannakis, G. B. (2019). Graph neural networks for predicting protein functions. In *Proceedings of the 8th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pages 221–225. IEEE. 37

Jain, N., Kalo, J.-C., Balke, W.-T., and Krestel, R. (2021). Do embeddings actually capture knowledge graph semantics? In *The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18*, pages 143–159. Springer. 27

Jain, S. and Bader, G. D. (2010). An improved method for scoring protein-protein interactions using semantic similarity within the gene ontology. *BMC Bioinformatics*, 11(1):562. 56

Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. (2015). Knowledge Graph Embedding via Dynamic Mapping Matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, USA. Association for Computational Linguistics. 17, 18, 19, 37

Ji, G., Liu, K., He, S., and Zhao, J. (2016). Knowledge Graph Completion with Adaptive Sparse Transfer Matrix. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 17, 18, 19

Jiang, H., Shen, F., Gao, F., and Han, W. (2021). Learning efficient, explainable and discriminative representations for pulmonary nodules classification. *Pattern Recognition*, 113:107825. 36

Jiang, J. J. and Conrath, D. W. (1997). Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In *Proceedings of the 10th International Conference on Research in Computational Linguistics*, pages 19–33, Taipei, Taiwan. The Association for Computational Linguistics and Chinese Language Processing (ACLCLP). 24, 25

Jimenez-Sanchez, G., Childs, B., and Valle, D. (2001). Human disease genes. *Nature*, 409(6822):853–855. 6

Jordan, M. I. and Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260. 28

Joshi, P., Masilamani, V., and Mukherjee, A. (2022). A knowledge graph embedding based approach to predict the adverse drug reactions using a deep neural network. *Journal of Biomedical Informatics*, 132:104122. 44, 52

Kanatsoulis, C. I. and Sidiropoulos, N. D. (2021). TeX-Graph: Coupled tensor-matrix knowledge-graph embedding for COVID-19 drug repurposing. In *Proceedings of the 2021 SIAM International Conference on Data Mining*, Philadelphia, USA. Society for Industrial and Applied Mathematics. 43, 52

Karim, M. R., Cochez, M., Jares, J. B., Uddin, M., Beyan, O., and Decker, S. (2019). Drug-Drug Interaction Prediction Based on Knowledge Graph Embeddings and Convolutional-LSTM Network. In *Proceedings of the 10th ACM International Conference on Bioinformatics,*

Computational Biology and Health Informatics, New York, USA. Association for Computing Machinery. 2, 41, 49

Kastrin, A., Ferk, P., and Leskošek, B. (2018). Predicting potential drug-drug interactions on topological and semantic similarity features using statistical learning. *PLOS ONE*, 13(5):1–23. 56, 68

Kaufman, S., Rosset, S., Perlich, C., and Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4):1–21. 76

Kawichai, T., Suratanee, A., and Plaimas, K. (2021). Meta-path based gene ontology profiles for predicting drug-disease associations. *IEEE Access*, 9:41809–41820. 42, 53

Kazemi, S. M. and Poole, D. (2018). Simple embedding for link prediction in knowledge graphs. *Advances in Neural Information Processing Systems*, 31. 4

Kim, E., Choi, A.-s., Nam, H., et al. (2019). Drug repositioning of herbal compounds via a machine-learning approach. *BMC Bioinformatics*, 20(10):33–43. 41, 56

Kipf, T. N. and Welling, M. (2017). Semi-Supervised Classification with Graph Convolutional Networks. *arXiv:1609.02907*. 32

Köhler, S., Øien, N. C., Buske, O. J., Groza, T., Jacobsen, J. O., McNamara, C., Vasilevsky, N., Carmody, L. C., Gourdine, J., Gargano, M., et al. (2019). Encoding clinical data with the Human Phenotype Ontology for computational differential diagnostics. *Current Protocols in Human Genetics*, 103(1):e92. 127

Koza, J. R. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press. 73

Krämer, A., Billaud, J.-N., Tugendreich, S., Shiffman, D., Jones, M., and Green, J. (2021). The Coronavirus Network Explorer: Mining a large-scale knowledge graph for effects of SARS-CoV-2 on host cell function. *BMC Bioinformatics*, 22(1):1–20. 43, 46, 62

Krix, S., DeLong, L. N., Madan, S., Domingo-Fernández, D., Ahmad, A., Gul, S., Zaliani, A., and Fröhlich, H. (2023). MultiGML: Multimodal graph machine learning for prediction of adverse drug events. *Heliyon*, 9(9). 45, 59, 62, 63

Krötzsch, M. and Weikum, G. (2016). Journal of Web Semantics: Special Issue on Knowledge Graphs. *Semantic Web (Aug. 2016). url: http://www. websemanticsjournal. org/index. php/ps/announcement/view/19*. 13

Kulmanov, M. and Hoehndorf, R. (2020). DeepPheno: Predicting single gene loss-of-function phenotypes using an ontology-aware hierarchical classifier. *PLOS Computational Biology*, 16(11):1–22. 41, 60, 62

Kulmanov, M., Khan, M. A., and Hoehndorf, R. (2018). DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier. *Bioinformatics*, 34(4):660–668. 46

Kulmanov, M., Liu-Wei, W., Yan, Y., and Hoehndorf, R. (2019). EL embeddings: geometric construction of models for the description logic EL++. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 14, 17, 18, 22, 99, 121, 125

Kulmanov, M., Smaili, F. Z., Gao, X., and Hoehndorf, R. (2021). Semantic similarity and machine learning with ontologies. *Briefings in Bioinformatics*, 22(4):bbaa199. 5, 16, 23, 26, 27, 64, 68, 70, 92, 99, 111, 112, 121, 127

Köhler, S., Gargano, M., Matentzoglu, N., Carmody, L. C., Lewis-Smith, D., Vasilevsky, N. A., and Danis, D. e. a. (2020). The Human Phenotype Ontology in 2021. *Nucleic Acids Research*, 49(D1):D1207–D1217. 14, 100, 115, 121

Lan, W., Dong, Y., Chen, Q., Zheng, R., Liu, J., Pan, Y., and Chen, Y.-P. P. (2022a). KGANCDA: predicting circRNA-disease associations based on knowledge graph attention network. *Briefings in Bioinformatics*, 23(1):bbab494. 44, 59

Lan, W., Zhang, H., Dong, Y., Chen, Q., Cao, J., Peng, W., Liu, J., and Li, M. (2022b). DRGCNCDA: Predicting circRNA-disease interactions based on knowledge graph and disentangled relational graph convolutional network. *Methods*, 208:35–41. 44, 59

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444. 31, 32

Lee, G., Park, C., and Ahn, J. (2019). Novel deep learning model for more accurate prediction of drug-drug interaction effects. *BMC Bioinformatics*, 20:1–8. 57, 68

Lei, Z., Sun, Y., Nanehkaran, Y. A., Yang, S., Islam, M. S., Lei, H., and Zhang, D. (2020). A novel data-driven robust framework based on machine learning and knowledge graph for disease classification. *Future Generation Computer Systems*, 102:534–548. 42, 53, 62

Li, D., Li, D., Wang, C., and Chen, Y. (2021). Network Embedding Method Based on Semantic Information. In *Proceedings of the 3rd International Conference on Advanced Information Science and System*, New York, USA. Association for Computing Machinery. 99

Li, Q., Zhao, K., Bustamante, C. D., Ma, X., and Wong, W. H. (2019). Xrare: a machine learning method jointly modeling phenotypes and genetic evidence for rare disease diagnosis. *Genetics in Medicine*, 21(9):2126–2134. 41, 56

Lin, D. (1998). An Information-Theoretic Definition of Similarity. In *Proceedings of the 15th International Conference on Machine Learning*, San Francisco, USA. Morgan Kaufmann Publishers Inc. 24, 25

Lin, W., Lan, H., and Li, B. (2021). Generative causal explanations for graph neural networks. In *Proceedings of the International Conference on Machine Learning*, pages 6666–6679. PMLR. 35

Lin, X., Quan, Z., Wang, Z.-J., Ma, T., and Zeng, X. (2020). KGNN: Knowledge Graph Neural Network for Drug-Drug Interaction Prediction. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, volume 380. International Joint Conferences on Artificial Intelligence. 42, 58, 79

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). Learning Entity and Relation Embeddings for Knowledge Graph Completion. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 16, 18, 19, 37

Ling, W., Dyer, C., Black, A. W., and Trancoso, I. (2015). Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, USA. Association for Computational Linguistics. 21, 118

Liu, J., Zhang, Z., and Razavian, N. (2018a). Deep ehr: Chronic disease prediction using medical notes. In *Proceedings of the 3rd Machine Learning for Healthcare Conference*, USA. PMLR. 48

Liu, L. and Zhu, S. (2021). Computational Methods for prediction of human protein-phenotype associations: A Review. *Phenomics*, 1(4):171–185. 114, 115

Liu, W., Liu, J., and Rajapakse, J. C. (2018b). Gene Ontology Enrichment Improves Performances of Functional Similarity of Genes. *Scientific Reports*, 8(1):12100. 56

Lu, Q., Nguyen, T. H., and Dou, D. (2021). Predicting patient readmission risk from medical text via knowledge graph enhanced multiview graph convolution. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, New York, USA. Association for Computing Machinery. 43, 60

Luck, K., Kim, D.-K., Lambourne, L., Spirohn, K., Begg, B. E., Bian, W., Brignall, R., Cafarelli, T., Campos-Laborie, F. J., Charloteaux, B., et al. (2020). A reference map of the human binary protein interactome. *Nature*, 580(7803):402–408. 142

Luo, D., Cheng, W., Xu, D., Yu, W., Zong, B., Chen, H., and Zhang, X. (2020). Parameterized explainer for graph neural network. *Advances in Neural Information Processing Systems*, 33:19620–19631. 35

Luo, P., Xiao, Q., Wei, P.-J., Liao, B., and Wu, F.-X. (2019). Identifying Disease-Gene Associations With Graph-Regularized Manifold Learning. *Frontiers in Genetics*, 10. 122

Ma, C., Zhou, Z., Liu, H., and Koslicki, D. (2023a). KGML-xDTD: a knowledge graph–based machine learning framework for drug treatment prediction and mechanism description. *GigaScience*, 12:giad057. 44, 53, 61, 63

Ma, X., Wang, M., Lin, S., Zhang, Y., Zhang, Y., Ouyang, W., and Liu, X. (2023b). Knowledge and data-driven prediction of organ failure in critical care patients. *Health Information Science and Systems*, 11(1):7. 45, 54, 61, 63

Mei, Y., Chen, Q., Lensen, A., Xue, B., and Zhang, M. (2022). Explainable Artificial Intelligence by Genetic Programming: A Survey. *IEEE Transactions on Evolutionary Computation*, 27(3):621–641. 34, 73

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv:1301.3781*. 117, 118

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63(2):81. 74, 104

Mjolsness, E. and DeCoste, D. (2001). Machine learning for science: state of the art and future prospects. *Science*, 293(5537):2051–2055. 33, 67

Mohamed, S. K., Nounu, A., and Nováček, V. (2019). Drug target discovery using knowledge graph embeddings. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, New York, USA. Association for Computing Machinery. 41, 50

Mohamed, S. K., Nounu, A., and Nováček, V. (2021). Biological applications of knowledge graph embedding models. *Briefings in Bioinformatics*, 22(2):1679–1693. 2, 16, 55, 64, 126

Mohamed, S. K., Nováček, V., and Nounu, A. (2020). Discovering protein drug targets using knowledge graph embeddings. *Bioinformatics*, 36(2):603–610. 42, 50

Mrksic, N., Séaghdha, D. Ó., Thomson, B., Gasic, M., Rojas-Barahona, L. M., Su, P.-H., Vandyke, D., Wen, T.-H., and Young, S. J. (2016). Counter-fitting Word Vectors to Linguistic Constraints. In *Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics*, USA. Association for Computational Linguistics. 143

Mukherjee, S., Cogan, J. D., Newman, J. H., Phillips, J. A., Hamid, R., Meiler, J., and Capra, J. A. (2021). Identifying digenic disease genes via machine learning in the Undiagnosed Diseases Network. *The American Journal of Human Genetics*, 108(10):1946–1963. 2, 42, 57, 61, 63, 68

Nicholson, D. N. and Greene, C. S. (2020). Constructing knowledge graphs and their biomedical applications. *Computational and Structural Biotechnology Journal*, 18:1414–1428. 1, 2

Nickel, M., Rosasco, L., and Poggio, T. (2016). Holographic Embeddings of Knowledge Graphs. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 17, 18, 20

Nickel, M., Tresp, V., and Kriegel, H.-P. (2011). A Three-Way Model for Collective Learning on Multi-Relational Data. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, Madison, USA. Omnipress. 17, 18, 19, 37

Nováček, V. and Mohamed, S. K. (2020). Predicting polypharmacy side-effects using knowledge graph embeddings. *AMIA Joint Summits on Translational Science Proceedings*, 2020:449. 42, 51

Nunes, S., Sousa, R. T., and Pesquita, C. (2023). Predicting gene-disease associations with knowledge graph embeddings over multiple ontologies. *Journal of Biomedical Semantics*. 43, 99, 100, 122

Olayan, R. S., Ashoor, H., and Bajic, V. B. (2018). DDR: efficient computational method to predict drug–target interactions using graph mining and machine learning approaches. *Bioinformatics*, 34(7):1164–1173. 57

Oliveira, D. and Pesquita, C. (2018). Improving the interoperability of biomedical ontologies with compound alignments. *Journal of Biomedical Semantics*, 9(1). 141

Osisanwo, F., Akinsola, J., Awodele, O., Hinmikaiye, J., Olakanmi, O., Akinjobi, J., et al. (2017). Supervised machine learning algorithms: classification and comparison. *International Journal of Computer Trends and Technology*, 48(3):128–138. 134

Padhiar, I., Seneviratne, O., Chari, S., Gruen, D., and McGuinness, D. L. (2021). Semantic modeling for food recommendation explanations. In *Proceedings of IEEE 37th International Conference on Data Engineering Workshops*, pages 13–19. IEEE. 36

Palmonari, M. and Minervini, P. (2020). Knowledge graph embeddings and explainable AI. *Knowledge Graphs for Explainable Artificial Intelligence: Foundations, Applications and Challenges*, 47:49. 34, 68, 91, 94

Patel, R., Guo, Y., Alhudhaif, A., Alenezi, F., Althubiti, S. A., Polat, K., et al. (2021). Graph-based link prediction between human phenotypes and genes. *Mathematical Problems in Engineering*, 2022. 54

Paulheim, H. (2017). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8(3):489–508. 13, 37

Pekar, V. and Staab, S. (2002). Taxonomy Learning: Factoring the Structure of a Taxonomy into a Semantic Classification Decision. In *Proceedings of the 19th International Conference on Computational Linguistics*, USA. Association for Computational Linguistics. 23, 24

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). DeepWalk: Online Learning of Social Representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA. Association for Computing Machinery. 17, 18, 21

Pesquita, C., Faria, D., Bastos, H., Falcao, A., and Couto, F. (2007). Evaluating GO-based semantic similarity measures. In *Proceedings of the 10th Annual Bio-Ontologies Meeting*. ISMB/ECCB. 25

Pesquita, C., Faria, D., Bastos, H., Ferreira, A. E., Falcão, A. O., and Couto, F. M. (2008). Metrics for GO based protein semantic similarity: a systematic evaluation. *BMC Bioinformatics*, 9:1–16. 78

Pesquita, C., Faria, D., Falcão, A. O., Lord, P., and Couto, F. M. (2009). Semantic Similarity in Biomedical Ontologies. *PLOS Computational Biology*, 5(7):1–12. 23, 25

Pezeshkpour, P., Tian, Y., and Singh, S. (2019). Investigating robustness and interpretability of link prediction via adversarial modifications. In *Conference of the North American Chapter of the Association for Computational Linguistics*, USA. Association for Computational Linguistics. 34, 94, 98

Piñero, J., Ramírez-Anguita, J. M., Saüch-Pitarch, J., Ronzano, F., Centeno, E., Sanz, F., and Furlong, L. I. (2019). The DisGeNET knowledge platform for disease genomics: 2019 update. *Nucleic Acids Research*, 48(D1):D845–D855. 100, 122

Poli, R., Langdon, W. B., McPhee, N. F., and Koza, J. R. (2008). *A field guide to genetic programming*. Published via `http://lulu.com`. 29

Poole, M. A. and O'Farrell, P. N. (1971). The Assumptions of the Linear Regression Model. *Transactions of the Institute of British Geographers*, 6(52):145–158. 28

Pope, P. E., Kolouri, S., Rostami, M., Martin, C. E., and Hoffmann, H. (2019). Explainability methods for graph convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10772–10781. 35

Portisch, J., Heist, N., and Paulheim, H. (2022). Knowledge graph embedding for data mining vs. knowledge graph embedding for link prediction–two sides of the same coin? *Semantic Web*, 13(3):399–422. 16, 37, 91

Portisch, J. and Paulheim, H. (2021). Putting RDF2Vec in order. In *Proceedings of International Semantic Web Conference 2021: posters, demos and industry tracks*, volume 2980. 17, 18, 21, 124

Purificato, E., Manikandan, B. A., Karanam, P. V., Pattadkal, M. V., and De Luca, E. W. (2021). Evaluating Explainable Interfaces for a Knowledge Graph-Based Recommender System. In *Proceedings of IntRS@ RecSys*, pages 73–88. 36

Quan, Y., Xiong, Z.-K., Zhang, K.-X., Zhang, Q.-Y., Zhang, W., and Zhang, H.-Y. (2023). Evolution-strengthened knowledge graph enables predicting the targetability and druggability of genes. *PNAS nexus*, 2(5):pgad147. 44, 51

Quinlan, J. R. (1990). Decision trees and decision-making. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(2):339–346. 29

Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30. 23, 24

Rajabi, E. and Etminani, K. (2022). Knowledge-graph-based explainable AI: A systematic review. *Journal of Information Science*, page 01655515221112844. 37, 38

Ramon, J. and Gärtner, T. (2003). Expressivity versus efficiency of graph kernels. In *Proceedings of the 1st International Workshop on Mining Graphs, Trees and Sequences*, pages 65–74. 16

Ren, Z.-H., You, Z.-H., Yu, C.-Q., Li, L.-P., Guan, Y.-J., Guo, L.-X., and Pan, J. (2022a). A biomedical knowledge graph-based method for drug–drug interactions prediction through combining local and global features with deep neural networks. *Briefings in Bioinformatics*, 23(5):bbac363. 44, 50

Ren, Z.-H., Yu, C.-Q., Li, L.-P., You, Z.-H., Guan, Y.-J., Wang, X.-F., and Pan, J. (2022b). BioDKG–DDI: predicting drug–drug interactions based on drug knowledge graph fusing biochemical information. *Briefings in Functional Genomics*, 21(3):216–229. 43, 49

Resnik, P. (1995). Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, San Francisco, USA. Morgan Kaufmann Publishers Inc. 23, 24, 25, 58

Rhee, S., Seo, S., and Kim, S. (2018). Hybrid approach of relation network and localized graph convolutional filtering for breast cancer subtype classification. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3527–3534. 37

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why Should I Trust You?": explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA. Association for Computing Machinery. 74

Ristoski, P. and Paulheim, H. (2016a). RDF2Vec: RDF graph embeddings for data mining. In *Proceedings of the 15th International Semantic Web Conference*, pages 498–514, Cham, Switzerland. Springer International Publishing. 17, 18, 21, 78, 79, 101, 117

Ristoski, P. and Paulheim, H. (2016b). Semantic Web in data mining and knowledge discovery: A comprehensive survey. *Journal of Web Semantics*, 36:1–22. 36

Robinson, P., Köhler, S., Oellrich, A., Genetics, S., Wang, K., Mungall, C., Lewis, S., Washington, N., Bauer, S., Seelow, D., Krawitz, P., Gilissen, C., Haendel, M., and Smedley, D. (2014). Improved exome prioritization of disease genes through cross-species phenotype comparison. *PCR Methods and Applications*, 24(2):340–348. 122

Roscher, R., Bohn, B., Duarte, M. F., and Garcke, J. (2020). Explainable machine learning for scientific insights and discoveries. *IEEE Access*, 8:42200–42216. 33, 67

Rossi, A., Firmani, D., Merialdo, P., and Teofili, T. (2022a). Explaining link prediction systems based on knowledge graph embeddings. In *Proceedings of the 2022 International Conference on Management of Data*, New York, USA. Association for Computing Machinery. 92

Rossi, A., Firmani, D., Merialdo, P., and Teofili, T. (2022b). Kelpie: an explainability framework for embedding-based link prediction models. *PVLDB (Proceedings of the VLDB Endowment)*, 15(12):3566–3569. 34, 94, 98

Roweis, S. T. and Saul, L. K. (2000). Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323–2326. 16, 17, 18

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215. 35

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536. 31, 101

Saadat, H., Shah, B., Halim, Z., and Anwar, S. (2022). Knowledge graph-based convolutional network coupled with sentiment analysis towards enhanced drug recommendation. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pages 1–12. 3, 44, 59

Sang, S., Yang, Z., Wang, L., Liu, X., Lin, H., and Wang, J. (2018). SemaTyP: a knowledge graph based literature mining method for drug discovery. *BMC Bioinformatics*, 19:1–11. 52

Santos, J. D., Pinto, F. R., Ferreira, J. F., Amaral, M. D., Zaccolo, M., and Farinha, C. M. (2020). Cytoskeleton regulators CAPZA2 and INF2 associate with CFTR to control its plasma membrane levels under EPAC1 activation. *Biochemical Journal*, 477(13):2561–2580. 106

Schlichtkrull, M., Kipf, T. N., Bloem, P., van den Berg, R., Titov, I., and Welling, M. (2018). Modeling Relational Data with Graph Convolutional Networks. In *Proceedings of the European Semantic Web Conference*, Cham, Switzerland. Springer International Publishing. 32

Schlicker, A., Domingues, F. S., Rahnenführer, J., and Lengauer, T. (2006). A new measure for functional similarity of gene products based on Gene Ontology. *BMC Bioinformatics*, 7:1–16. 58

Seco, N., Veale, T., and Hayes, J. (2004). An Intrinsic Information Content Metric for Semantic Similarity in WordNet. In *Proceedings of the 16th European Conference on Artificial Intelligence*, NLD. IOS Press. 24, 73

Sezaki, T., Tomiyama, L., Kimura, Y., Ueda, K., and Kioka, N. (2013). Dlg5 interacts with the TGF-beta receptor and promotes its degradation. *FEBS Letters*, 587(11):1624–1629. 86, 106

Shapiro, J. (1999). Genetic algorithms in machine learning. In *Advanced Course on Artificial Intelligence*, pages 146–168. Springer. 30

Shen, F., Peng, S., Fan, Y., Wen, A., Liu, S., Wang, Y., Wang, L., and Liu, H. (2019). HPO2Vec+: Leveraging heterogeneous knowledge resources to enrich node embeddings for the Human Phenotype Ontology. *Journal of Biomedical Informatics*, 96:103246. 41, 53

Shervashidze, N. and Borgwardt, K. M. (2009). Fast Subtree Kernels on Graphs. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, pages 1660–1668, Red Hook, USA. Curran Associates Inc. 37

Shervashidze, N., Schweitzer, P., van Leeuwen, E. J., Mehlhorn, K., and Borgwardt, K. M. (2011). Weisfeiler-Lehman Graph Kernels. *Journal of Machine Learning Research*, 12(77):2539–2561. 16, 37

Shervashidze, N., Vishwanathan, S., Petri, T., Mehlhorn, K., and Borgwardt, K. (2009). Efficient graphlet kernels for large graph comparison. In van Dyk, D. and Welling, M., editors, *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, volume 5, pages 488–495, Cambridge, USA. MIT Press. 16, 37

Smaili, F. Z., Gao, X., and Hoehndorf, R. (2018a). Onto2Vec: joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics*, 34(13):i52–i60. 17, 18, 22

Smaili, F. Z., Gao, X., and Hoehndorf, R. (2018b). OPA2Vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics*, 35(12):2133–2140. 17, 18, 22, 55, 99, 121, 122

Soman, K., Nelson, C. A., Cerono, G., and Baranzini, S. E. (2023). Time-aware Embeddings of Clinical Data using a Knowledge Graph. *Pacific Symposium on Biocomputing*, 28:97—108. 44, 49, 62

Sosa, D. N., Derry, A., Guo, M., Wei, E., Brinton, C., and Altman, R. B. (2019). A literature-based knowledge graph embedding method for identifying drug repurposing opportunities in rare diseases. *Pacific Symposium on Biocomputing*, 25:463–474. 3, 41, 52, 61, 63

Sousa, R. T., Silva, S., , and Pesquita, C. (2023a). Benchmark datasets for biomedical knowledge graphs with negative statements. In *Workshop on Semantic Web solutions for large-scale biomedical data analytics at Extended Semantic Web Conference*. 7

Sousa, R. T., Silva, S., Paulheim, H., and Pesquita, C. (2023b). Biomedical Knowledge Graph Embeddings with Negative Statements. In *International Semantic Web Conference*. 7

Sousa, R. T., Silva, S., and Pesquita, C. (2020). Evolving knowledge graph similarity for supervised learning in complex biomedical domains. *BMC Bioinformatics*, 21:1–19. 1, 41, 71, 94, 100, 121

Sousa, R. T., Silva, S., and Pesquita, C. (2021). evoKGsim+: a framework for tailoring knowledge graph-based similarity for supervised learning. In *Proceedings of the Extended Semantic Web Conference - Poster and Demo Track*, Cham, Switzerland. Springer International Publishing. 42, 68, 93

Sousa, R. T., Silva, S., and Pesquita, C. (2022). Explaining Protein-Protein Interaction Predictions with Genetic Programming. In *Late-breaking abstracts EvoStar*, page 30. 7

Sousa, R. T., Silva, S., and Pesquita, C. (2023c). Explainable Representations for Relation Prediction in Knowledge Graphs. In *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning*, pages 635–646. 7

Staab, S. and Studer, R. (2010). *Handbook on ontologies*. Springer-Verlag. 1, 13, 68

Steenwinckel, B., Vandewiele, G., Weyns, M., Agozzino, T., Turck, F. D., and Ongenae, F. (2022). INK: knowledge graph embeddings for node classification. *Data Mining and Knowledge Discovery*, 36(2):620–667. 37

Su, C., Tong, J., Zhu, Y., Cui, P., and Wang, F. (2018). Network embedding in biomedical data science. *Briefings in Bioinformatics*, 21(1):182–197. 17, 21, 64

Su, X., Hu, L., You, Z., Hu, P., and Zhao, B. (2022a). Attention-based knowledge graph representation learning for predicting drug-drug interactions. *Briefings in Bioinformatics*, 23(3):bbac140. 44, 50

Su, X., You, Z.-H., Huang, D.-s., Wang, L., Wong, L., Ji, B., and Zhao, B. (2022b). Biomedical knowledge graph embedding with capsule network for multi-label drug-drug interaction prediction. *IEEE Transactions on Knowledge and Data Engineering*, 35(6):5640–5651. 44, 58, 62

Suchanek, F. M., Kasneci, G., and Weikum, G. (2007). Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, New York, USA. Association for Computing Machinery. 11

Szklarczyk, D., Gable, A. L., Nastou, K. C., Lyon, D., Kirsch, R., Pyysalo, S., Doncheva, N. T., Legeay, M., et al. (2021). The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Research*, 49(D1):D605–D612. 75, 76, 100, 121

Tomczak, A., Mortensen, J. M., Winnenburg, R., Liu, C., Alessi, D. T., Swamy, V., Vallania, F., Lofgren, S., Haynes, W., Shah, N. H., et al. (2018). Interpretation of biological experiments changes with evolution of the Gene Ontology and its annotations. *Scientific Reports*, 8(1):1–10. 76

Trapnell, B. C., Whitsett, J. A., and Nakata, K. (2003). Pulmonary alveolar proteinosis. *New England Journal of Medicine*, 349(26):2527–2539. 105

Traverso, I., Vidal, M.-E., Kämpgen, B., and Sure-Vetter, Y. (2016). GADES: A graph-based semantic similarity measure. In *Proceedings of the 12th International Conference on Semantic Systems*, New York, USA. Association for Computing Machinery. 26

Traverso-Ribón, I. and Vidal, M.-E. (2018). GARUM: a semantic similarity measure based on machine learning and entity characteristics. In *Proceedings of the 29th International Conference Database and Expert Systems Applications*, Cham, Switzerland. Springer International Publishing. 24

Trouillon, T., Welbl, J., Riedel, S., Gaussier, E., and Bouchard, G. (2016). Complex Embeddings for Simple Link Prediction. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*, USA. PMLR. 17, 18, 20

Tsuda, K. and Saigo, H. (2010). Graph classification. *Managing and mining graph data*, pages 337–363. 37

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11). 101

Vandewiele, G., Steenwinckel, B., Bonte, P., Weyns, M., Paulheim, H., Ristoski, P., Turck, F. D., and Ongenae, F. (2018). Walk Extraction Strategies for Node Embeddings with RDF2Vec in Knowledge Graphs. In *Proceedings of the International Conference Database and Expert Systems Applications - Workshops*, Cham, Switzerland. Springer International Publishing. 22

Vanunu, O., Magger, O., Ruppin, E., Shlomi, T., and Sharan, R. (2010). Associating Genes and Protein Complexes with Disease via Network Propagation. *PLoS Computational Biology*, 6. 122

Vilela, J., Asif, M., Marques, A. R., Santos, J. X., Rasga, C., Vicente, A., and Martiniano, H. (2023). Biomedical knowledge graph embeddings for personalized medicine: Predicting disease-gene associations. *Expert Systems*, 40(5):e13181. 44, 53

Vu, M. and Thai, M. T. (2020). Pgm-explainer: Probabilistic graphical model explanations for graph neural networks. *Advances in Neural Information Processing Systems*, 33:12225–12235. 35

Wang, D., Cui, P., and Zhu, W. (2016). Structural Deep Network Embedding. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, USA. Association for Computing Machinery. 17, 18, 20

Wang, D., Yang, Q., Abdul, A., and Lim, B. Y. (2019a). Designing theory-driven user-centric explainable AI. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–15. 68

Wang, J. Z., Du, Z., Payattakool, R., Yu, P. S., and Chen, C.-F. (2007). A new method to measure the semantic similarity of GO terms. *Bioinformatics*, 23(10):1274–1281. 58

Wang, L., Wong, L., Li, Z., Huang, Y., Su, X., Zhao, B., and You, Z. (2022a). A machine learning framework based on multi-source feature fusion for circRNA-disease association prediction. *Briefings in Bioinformatics*, 23(5):bbac388. 43, 57

Wang, M., Ma, X., Si, J., Tang, H., Wang, H., Li, T., Ouyang, W., Gong, L., Tang, Y., He, X., et al. (2021a). Adverse drug reaction discovery using a tumor-biomarker knowledge graph. *Frontiers in genetics*, 11:625659. 42, 51, 61, 63

Wang, M., Wang, H., Liu, X., Ma, X., and Wang, B. (2021b). Drug-drug interaction predictions via knowledge graph and text embedding: instrument validation study. *JMIR Medical Informatics*, 9(6):e28277. 43, 49

Wang, Q., Mao, Z., Wang, B., and Guo, L. (2017). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743. 2, 4, 17, 19, 26, 37, 91

Wang, S., Du, Z., Ding, M., Rodriguez-Paton, A., and Song, T. (2022b). KG-DTI: a knowledge graph based deep learning method for drug-target interaction predictions and Alzheimer's disease drug repositions. *Applied Intelligence*, 52(1):846–857. 43, 51

Wang, S., Xu, F., Li, Y., Wang, J., Zhang, K., Liu, Y., Wu, M., and Zheng, J. (2021c). KG4SL: knowledge graph neural network for synthetic lethality prediction in human cancers. *Bioinformatics*, 37(Supplement_1):i418–i425. 43, 60, 62

Wang, X., Wang, D., Xu, C., He, X., Cao, Y., and Chua, T.-S. (2019b). Explainable reasoning over knowledge graphs for recommendation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 5329–5336. 38

Wang, Y., Yang, S., Zhao, J., Du, W., Liang, Y., Wang, C., Zhou, F., Tian, Y., and Ma, Q. (2019c). Using machine learning to measure relatedness between genes: a multi-features model. *Scientific Reports*, 9(1):4192. 41, 57, 68

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014). Knowledge Graph Embedding by Translating on Hyperplanes. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, Washington DC, USA. AAAI Press. 16, 18, 19, 37, 101

Warwick Vesztrocy, A. and Dessimoz, C. (2020). Benchmarking Gene Ontology function predictions using negative annotations. *Bioinformatics*, 36(Supplement_1):i210–i218. 115, 121

Watson, D. S., Gultchin, L., Taly, A., and Floridi, L. (2021). Local explanations via necessity and sufficiency: Unifying theory and practice. In *Proceedings of the 37th Conference on Uncertainty in Artificial Intelligence*, USA. PMLR. 92

Whetzel, P. L., Noy, N. F., Shah, N. H., Alexander, P. R., Nyulas, C., Tudorache, T., and Musen, M. A. (2011). BioPortal: enhanced functionality via new Web services from the National Center for Biomedical Ontology to access and use ontologies in software applications. *Nucleic Acids Research*, 39(suppl2):W541–W545. 14

Wollschlaeger, B., Eichenberg, E., and Kabitzsch, K. (2020). Explain Yourself: A Semantic Annotation Framework to Facilitate Tagging of Semantic Information in Health Smart Homes. In *Proceedings of HEALTHINF*, pages 133–144. 37

Wu, B., Yang, X., Pan, S., and Yuan, X. (2021). Adapting membership inference attacks to GNN for graph classification: Approaches and implications. In *Proceedings of IEEE International Conference on Data Mining*, pages 1421–1426. IEEE. 37

Wu, X., Zhu, L., Guo, J., Zhang, D.-Y., and Lin, K. (2006). Prediction of yeast protein–protein interaction network: insights from the Gene Ontology and annotations. *Nucleic Acids Research*, 34(7):2137–2150. 56

Wu, Y. (2023). Predicting Synthetic Lethality in Human Cancers via Knowledge Graph Summarization. In *Proceedings of the 2023 3rd International Conference on Bioinformatics and Intelligent Computing*, New York, USA. Association for Computing Machinery. 45, 60, 62

Xie, Y., Liang, Y., Gong, M., Qin, A., Ong, Y.-S., and He, T. (2022). Semisupervised graph neural networks for graph classification. *IEEE Transactions on Cybernetics*. 37

Xiong, B., Potyka, N., Tran, T.-K., Nayyeri, M., and Staab, S. (2022). Faithful Embeddings for EL++ Knowledge Bases. In *Proceeding of the International Semantic Web Conference*, pages 22–38, Cham, Switzerland. Springer International Publishing. 17, 18, 22, 99, 125

Xiong, Z., Huang, F., Wang, Z., Liu, S., and Zhang, W. (2021). A multimodal framework for improving in silico drug repositioning with the prior knowledge from knowledge graphs. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(5):2623–2631. 42, 52

Xu, Q.-S. and Liang, Y.-Z. (2001). Monte Carlo cross validation. *Chemometrics and Intelligent Laboratory Systems*, 56(1):1–11. 124

Xue, H., Peng, J., and Shang, X. (2019). Predicting disease-related phenotypes using an integrated phenotype similarity measurement based on HPO. *BMC Systems Biology*, 13(2):1–12. 127

Yang, B., Yih, S. W.-t., He, X., Gao, J., and Deng, L. (2015). Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In *Proceedings of the International Conference on Learning Representations*. 17, 18, 20, 37, 78, 101

Yao, J., Sun, W., Jian, Z., Wu, Q., and Wang, X. (2022). Effective knowledge graph embeddings based on multidirectional semantics relations for polypharmacy side effects prediction. *Bioinformatics*, 38(8):2315–2322. 44, 52

Ye, C., Swiers, R., Bonner, S., and Barrett, I. (2022). A knowledge graph-enhanced tensor factorisation model for discovering drug targets. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(6):3070–3080. 44, 53

Ye, Q., Hsieh, C.-Y., Yang, Z., Kang, Y., Chen, J., Cao, D., He, S., and Hou, T. (2021). A unified drug–target interaction prediction framework based on knowledge graph and recommendation system. *Nature Communications*, 12(1):6775. 42, 50

Ye, Q., Yang, R., Cheng, C.-l., Peng, L., Lan, Y., et al. (2023). Combining the External Medical Knowledge Graph Embedding to Improve the Performance of Syndrome Differentiation Model. *Evidence-Based Complementary and Alternative Medicine*, 2023. 44, 54

Ying, Z., Bourgeois, D., You, J., Zitnik, M., and Leskovec, J. (2019). GNNexplainer: Generating explanations for graph neural networks. *Advances in Neural Information Processing Systems*, 32. 35

Yu, Y., Huang, K., Zhang, C., Glass, L. M., Sun, J., and Xiao, C. (2021). SumGNN: multi-typed drug interaction prediction via efficient knowledge graph summarization. *Bioinformatics*, 37(18):2988–2995. 43, 58, 62, 63

Yuan, H., Yu, H., Gui, S., and Ji, S. (2022). Explainability in graph neural networks: A taxonomic survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):5782–5799. 34

Yue, X., Wang, Z., Huang, J., Parthasarathy, S., Moosavinasab, S., Huang, Y., Lin, S. M., Zhang, W., Zhang, P., and Sun, H. (2019). Graph embedding on biomedical networks: methods, applications and evaluations. *Bioinformatics*, 36(4):1241–1251. 45

Zakeri, P., Simm, J., Arany, A., ElShal, S., and Moreau, Y. (2018). Gene prioritization using Bayesian matrix factorization with genomic and phenotypic side information. *Bioinformatics*, 34:i447 – i456. 122

Zhang, F., Song, H., Zeng, M., Wu, F.-X., Li, Y., Pan, Y., and Li, M. (2020a). A deep learning framework for gene ontology annotations with sequence-and network-based information. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 18(6):2208–2217. 42, 46

Zhang, F., Sun, B., Diao, X., Zhao, W., and Shu, T. (2021a). Prediction of adverse drug reactions based on knowledge graph embedding. *BMC Medical Informatics and Decision Making*, 21(1):1–11. 42, 52

Zhang, M., Cui, Z., Neumann, M., and Chen, Y. (2018). An end-to-end deep learning architecture for graph classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32. 37

Zhang, R., Hristovski, D., Schutte, D., Kastrin, A., Fiszman, M., and Kilicoglu, H. (2021b). Drug repurposing for COVID-19 via knowledge graph completion. *Journal of Biomedical Informatics*, 115:103696. 42, 51, 61, 63

Zhang, S., Lin, X., and Zhang, X. (2021c). Discovering DTI and DDI by knowledge graph with MHRW and improved neural network. In *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine*. IEEE. 43, 50

Zhang, S.-B. and Tang, Q.-R. (2016). Protein–protein interaction inference based on semantic similarity of Gene Ontology terms. *Journal of Theoretical Biology*, 401:30–37. 57, 67, 68, 92, 121

Zhang, W., Deng, S., Wang, H., Chen, Q., Zhang, W., and Chen, H. (2020b). Xtranse: Explainable knowledge graph embedding for link prediction with lifestyles in e-commerce. In *Proceedings of the 9th Joint International Conference*, pages 78–87. Springer. 34

Zhang, W., Paudel, B., Zhang, W., Bernstein, A., and Chen, H. (2019). Interaction Embeddings for Prediction and Explanation in Knowledge Graphs. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, page 96–104, New York, NY, USA. Association for Computing Machinery. 34

Zhang, X. and Che, C. (2021). Drug repurposing for parkinson's disease by integrating knowledge graph completion model and knowledge fusion of medical literature. *Future Internet*, 13(1):14. 42, 52

Zhang, X.-M., Liang, L., Liu, L., and Tang, M.-J. (2021d). Graph neural networks and their current applications in bioinformatics. *Frontiers in Genetics*, 12:690049. 64, 143

Zhang, Y.-H., Zeng, T., Chen, L., Huang, T., and Cai, Y.-D. (2021e). Determining protein–protein functional associations by functional rules based on gene ontology and KEGG pathway. *Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics*, 1869(6):140621. 42, 47, 62, 67

Zhong, X., Kaalia, R., and Rajapakse, J. C. (2019). GO2Vec: transforming GO terms and proteins to vector representations via graph embeddings. *BMC Genomics*, 20(9):918. 2, 79

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81. 4, 143

Zhu, Y., Zhou, Y., Liu, Y., Wang, X., and Li, J. (2023). SLGNN: synthetic lethality prediction in human cancers based on factor-aware knowledge graph neural network. *Bioinformatics*, 39(2):btad015. 45, 60, 62, 63

Zong, N., Kim, H., Ngo, V., and Harismendy, O. (2017). Deep mining heterogeneous networks of biomedical linked data to predict novel drug–target associations. *Bioinformatics*, 33(15):2337–2344. 55

Appendix A

Explaining Protein-Protein Interactions with Knowledge Graph-based Semantic Similarity

Explaining Protein-Protein Interactions with Knowledge Graph-based Semantic Similarity

Rita T. Sousa, Sara Silva, Catia Pesquita

[a]*LASIGE, Faculdade de Ciências da Universidade de Lisboa, Lisboa, Portugal*

Abstract

The application of artificial intelligence and machine learning methods for several biomedical applications, such as protein-protein interaction prediction, has gained significant traction in recent decades. However, explainability is a key aspect of using machine learning as a tool for scientific discovery. Explainable artificial intelligence approaches help clarify algorithmic mechanisms and identify potential bias in the data.

Given the complexity of the biomedical domain, explanations should be grounded in domain knowledge which can be achieved by using ontologies and knowledge graphs. These knowledge graphs express knowledge about a domain by capturing different perspectives of the representation of real-world entities. However, the most popular way to explore knowledge graphs with machine learning is through using embeddings, which are not explainable. As an alternative, knowledge graph-based semantic similarity offers the advantage of being explainable. Additionally, similarity can be computed to capture different semantic aspects within the knowledge graph and increasing the explainability of predictive approaches.

We propose a novel method to generate explainable vector representations, KGsim2vec, that uses aspect-oriented semantic similarity features to represent pairs of entities in a knowledge graph. Our approach employs a set of machine learning models, including decision trees, genetic programming, random forest and eXtreme gradient boosting, to predict relations between entities.

The experiments reveal that considering multiple semantic aspects when representing the similarity between two entities improves explainability and predictive performance. KGsim2vec performs better than black-box methods based on knowledge graph embeddings or graph neural networks. Moreover, KGsim2vec produces global models that can capture biological phenomena

and elucidate data biases.

Keywords: Machine Learning, Explainable Artificial Intelligence, Knowledge Graph, Semantic Similarity, Protein-Protein Interaction Prediction

1. Background

The potential of artificial intelligence (AI) as a tool for scientific discovery in the biomedical domain has long been recognized, with machine learning (ML), pattern mining and reasoning playing roles in several steps of the scientific process (Mjolsness and DeCoste, 2001). One of AI's most promising and successful applications is its ability to predict protein-protein interactions (PPIs). Proteins often interact with each other to carry out vital physiological functions. Through the integration of ML algorithms, AI predictive models have enabled researchers to identify potential protein interactions with implications for drug discovery and personalized medicine (Zhang and Tang, 2016; Chen et al., 2019; Zhang et al., 2021b; Ieremie et al., 2022).

Although AI plays an important role in the scientific process, its application in science requires explainability, a fundamental piece to transform AI into a scientific tool that is able to uncover new knowledge, to understand the mechanisms that underlie the natural phenomena that are being predicted and to distinguish between meaningful predictions and spurious correlations (Barredo Arrieta et al., 2020). However, the vast majority of scientific projects that employ AI are not concerned with explainability (Roscher et al., 2020). In the biomedical domain, both the complexity of the data and the natural phenomena under study highlight the necessity of domain knowledge to support explainability (Holzinger et al., 2017). Explainable AI is gaining traction as a potential solution to ensure that algorithms and their predictions can be human-understandable. (Durán, 2021) distinguishes scientific explainable AI and other types of explainable AI. According to the author, current approaches offer explanations that answer *how* the algorithm reached a given output but not why. To address this issue, they propose the concept of a genuine scientific explanation within the context of AI that should answer the *why* question. A knowledge-enabled explainable AI system includes a representation of the domain knowledge in the field of application, which is explored to generate user-comprehensible and context-aware explanations of the mechanistic functioning of the AI system and the knowledge used (Chari

et al., 2020).

Biomedical ontologies express knowledge about a domain and allow the description of complex biological phenomena that are not easily captured in mathematical form (Staab and Studer, 2010). As such, they provide the scaffolding for comparing biological entities at a higher level of complexity by comparing the ontology classes with which they are annotated. Measuring the semantic similarity (SS) between biomedical entities through their ontology annotations has become a cornerstone bioinformatics application in protein-protein interaction prediction (Zhang and Tang, 2016; Chen et al., 2019; Wang et al., 2019b), disease-associated gene identification (Hoehndorf et al., 2011; Asif et al., 2018; Mukherjee et al., 2021), and drug-drug interaction prediction (Abdelaziz et al., 2017; Kastrin et al., 2018; Lee et al., 2019). Semantic similarity has been combined with ML approaches in different supervised and unsupervised learning tasks, but in recent years, a spate of novel knowledge graph (KG) embeddings and deep learning-based approaches have been employed over the same tasks with success (Kulmanov et al., 2020; Chen et al., 2019; Ieremie et al., 2022). Knowledge graph embeddings map each node of a knowledge graph to a lower-dimensional space in which its graph position and the structure of its local graph neighborhood are preserved as much as possible (Wang et al., 2017). However, in some applications, classical semantic similarity measures still outperform knowledge graph embeddings (Sousa et al., 2021). One advantage of employing semantic similarity-based features over knowledge graph embeddings is that similarity assessment is a natural explanatory mechanism (Wang et al., 2019a), whereas knowledge graph embeddings are opaque vectorial representations (Palmonari and Minervini, 2020).

Although a semantic similarity score as a feature is explainable in its essence, since it reflects the similarity of the entities according to the domain represented by the ontology, it is a very compact explanation, often reduced to a single numerical score or at most to one score per ontology root of a general-purpose knowledge graph. This similarity score tells us if two proteins are similar but not why they are similar. For instance, in Gene Ontology (GO), it is common to measure the semantic similarity of annotated gene products according to its three branches: biological process, cellular component and molecular function. However, providing a very general explanation, such as the fact that both proteins have a higher biological process similarity, would not enhance the ML model's reliability or elucidate the biological phenomena. We hypothesize that by measuring the semantic

similarity between entities targeting specific subgraphs of the ontology we can increase the explainability of semantic similarity-based features without substantial performance sacrifices. These subgraphs capture different *semantic aspects* (SAs), i.e., different perspectives of the representation of ontology-annotated entities.

In this work, we tackle two key challenges. Firstly, we address the lack of explainability in well-known knowledge graph-based representations, such as knowledge graph embeddings. Secondly, we address the general-purpose nature of knowledge graphs that leads to knowledge graph-based representations of entities lacking meaningful interpretations. We propose KGsim2vec, a novel method to generate explainable vector representations by representing entity pairs in a knowledge graph through aspect-oriented semantic similarity features. This technique explores the rich semantics of the ontology to identify the semantic aspects and compute the similarity for each aspect. Then the similarities are given as features for an ML model for relation prediction. Given a prediction, our method explains it by using either the ML model or posthoc techniques. Additionally, we propose a novel approach to evaluate explanation quality, which combines the number of features in an explanation with their informativeness as measured in the knowledge graph.

For evaluation, we focus on protein-protein interaction prediction based on Gene Ontology. Given the high costs and challenges associated with experimentally determining protein interactions, several knowledge graph-based approaches have been used to identify protein pairs that are likely to interact and should be validated in experimental assays (Zhong and Rajapakse, 2020; Maetschke et al., 2011; Bandyopadhyay and Mallick, 2017; Jain and Bader, 2010), making the process more efficient. However, most of these approaches are black-box and only a few use explainable AI techniques to uncover these interactions. Explanatory mechanisms can elucidate the potential mechanisms behind the predicted relation, which can help determine the appropriate experimental procedure to confirm the predicted relation. Additionally, explanatory mechanisms can identify data biases that may result in misclassification, leading to discarding the candidate pair.

Our main contributions are the following:

- We propose KGsim2vec[1], a novel method for generating explainable knowledge graph-based similarity features that represent entity pairs.

[1]The code is available at https://github.com/liseda-lab/ExplainablePPI.

- We design a novel approach for evaluating the quality of explanations, which takes into account both their size and informativeness.
- We report extensive experimental results for protein-protein interaction prediction, which demonstrate the effectiveness of KGsim2vec in producing useful explanations for relation prediction.

2. Related Work

This work builds upon the relevant research in two domains: the explainable AI techniques and the use of ontologies in ML.

2.1. Explainable Artificial Intelligence techniques

Explainable AI is capable of providing a human-understandable description of the logic, behavior or factors that influence the learning process (Barredo Arrieta et al., 2020). Several taxonomies have been proposed to classify explainable AI techniques. Barredo Arrieta et al. (2020) divided explainable AI approaches into two groups: transparent models, which are interpretable by design, or post-hoc explainability, where models are explained employing external techniques. In the state of the art, the number of models recognized as transparent and interpretable for humans are few and include decision trees, linear models, and genetic programming models (Mei et al., 2022). Post-hoc explainability includes techniques that enhance the interpretability of models that are not interpretable by design and can be divided into model-agnostic techniques (designed to be applied to any ML model) and model-specific techniques (tailored to explain a particular ML model). These techniques for post-hoc explainability can rely on text explanations, visual explanations, local explanations, explanations by example, explanations by simplification, and feature relevance explanations (Barredo Arrieta et al., 2020). Explanations by simplification are one of the most used techniques for producing simplified models that only represent specific sections of a model. Feature relevance methods are also frequently employed and consist of computing the relevance score of each variable in the ML model. A comparison of the importance values unveils the most relevant variables to the predictions made by the model.

Although most explainable AI techniques focus on explaining black-box models with post-hoc techniques, some works instead support using inherently interpretable ML models. Rudin (2019) identified several reasons why

black-box explanations are not the best choice. In the first place, explanations do not represent what the original black-box model computes to the extent that a complex model would not be needed if the explanations were perfectly faithful. In addition, explanations often do not make sense, and even when they do, they do not provide enough detail to understand what the black-box is doing. Following this, several works have been proposed that employ rule-based approaches. Anguita-Ruiz et al. (2020) present a rule-based explainable AI strategy to mine time-delayed gene relationships from in vivo human temporal microarray data and find biologically relevant sequential rules. The output gene rules are then assessed in quality and robustness and integrated with external biological resources. Bourgeais et al. (2022) proposes GraphGONet, a self-explaining neural network that integrates a biomedical ontology into its hidden layers for phenotype prediction on gene expression. KGsim2vec distinguishes itself from these works by focusing on generating explainable features based on the semantic similarity computed for different semantic aspects of a knowledge graph.

Despite the vital role of protein interactions within our biological systems, only some works have aimed to elucidate these interactions through explainable AI techniques. Recently, we also proposed an approach, SEEK, which aims to explain protein interactions based on knowledge graph embeddings (Sousa et al., 2023). However, it is important to note that SEEK primarily focuses on explaining specific interactions using knowledge graph embedding-based representations. In contrast, this work focuses on similarity-based representations. In addition, KGsim2vec can provide global explanations when used with transparent ML models.

2.2. Ontologies and Machine Learning

In the last decades, several semantic similarity measures have been proposed with most measures falling in the category of taxonomic semantic similarity (also referred to as ontology-based semantic similarity, or only semantic similarity). Taxonomic semantic similarity measures are generally designed by an expert based on assumptions about how an ontology is used and what should constitute a similarity. They extensively use the taxonomical aspect of an ontology, comparing classes based on subclass/superclass relations. Semantic similarity measures can be distinguished based on the entities they intend to compare since we can measure the similarity between either ontology classes (Resnik, 1995) or real-world entities (annotated with

a set of classes) (Pesquita et al., 2007; Traverso et al., 2016; Traverso-Ribón and Vidal, 2018).

Recently, a few approaches combining taxonomic semantic similarity with ML have been proposed. GARUM (Traverso-Ribón and Vidal, 2018) is based on a supervised regression algorithm that receives several similarity measures of hierarchy, neighborhood, shared information, and attributes and then predicts a final similarity score. evoKGsim (Sousa et al., 2020) employs genetic programming to learn appropriate combinations of semantic similarity scores to predict protein interactions.

However, most of the work exploring ontologies with ML is focused on knowledge graph embeddings (Hogan et al., 2021). Knowledge graph embedding methods map each knowledge graph entity into a vector representation. In scenarios where pair representations are used, such as protein-protein interaction prediction, these methods first learn distinctive vector representations for each knowledge graph entity. Subsequently, they employ operators to merge the representations of the two entities forming the pair. Several methods for building graph embeddings have been proposed (Cai et al., 2018). While some focus on exploring the graph facts solely (like translational distance models (Bordes et al., 2013; Wang et al., 2014; Lin et al., 2015; He et al., 2015) or semantic matching (Yang et al., 2015; Trouillon et al., 2016; Nickel et al., 2016)), others also include additional information, such as entity types, relation paths, axioms and rules, or textual information. More recently, knowledge graph embedding approaches that tailor representations by considering specific aspects of a knowledge graph have been proposed. Path-based approaches, such as RDF2Vec (Ristoski and Paulheim, 2016) and OWL2Vec* (Chen et al., 2021), have been proposed by transforming the ontology graph into node sequences that are given as input to a neural language model. OPA2Vec (Smaili et al., 2018) considers the lexical portion of the knowledge graph, specifically the labels of entities. EL (Kulmanov et al., 2019) and BoxEL (Xiong et al., 2022) are geometric approaches that consider the logical structure of the ontology.

Kulmanov et al. (2020) provides an overview of methods incorporating semantic similarity measures and ontology embeddings into ML methods with biomedical applications. However, to the best of our knowledge, none of these approaches provide scientific explanations. Knowledge graph embeddings are not interpretable and cannot answer the question about *how* the method arrived at a specific output. Although interpretable by design, the semantic similarity is usually calculated using a general-purpose knowledge

graph, failing to answer the *why* question.

3. Methods

3.1. Problem Formulation

We focus on ontology-rich knowledge graphs with ontologies, where ontologies are used to describe individual instances, while the instances themselves are usually flat with no connections between them. As a result, these knowledge graphs have two distinct types of nodes: nodes corresponding to individual entities and nodes corresponding to ontology classes. These knowledge graphs also contain two types of edges: one type that connects ontology classes to each other and another type that links individual entities to the classes that describe them. For example, in the Gene Ontology knowledge graph, an edge between a protein (individual entity) and a Gene Ontology class (ontology class) indicates that a particular protein P performs a specific function F described in the Gene Ontology. At the same time, an edge between two Gene Ontology classes, $F1$ and $F2$ can represent the fact that one Gene Ontology class is a subclass of the other one. Knowledge graph semantic similarity measures compute the similarity between two entities by comparing the ontology classes which are connected to each entity considering the structure of the ontology itself.

We aim to learn a relation between two knowledge graph entities when the relation itself is not explicitly defined in the knowledge graph. To tackle this prediction task, recent approaches employ knowledge graph embedding methods. These methods generate vector representations for each entity, which are then combined to be used as input features for ML methods. However, these vector representations are non-explainable since each dimension does not represent any specific meaning. Furthermore, these approaches rely on creating a representation of each entity using the whole knowledge graph, ignoring the different semantic aspects of the knowledge graph. A semantic aspect is a perspective of the knowledge graph entities, and it can be represented as a subgraph. For example, the three branches of the Gene Ontology (biological process, cellular component and molecular function) can represent three semantic aspects. Entities are characterized according to various semantic aspects, but only a few of them might be relevant for predicting a particular relationship. In a prior investigation (Sousa et al., 2020), we demonstrated that not all branches of the Gene Ontology are equally important for protein-protein interaction prediction.

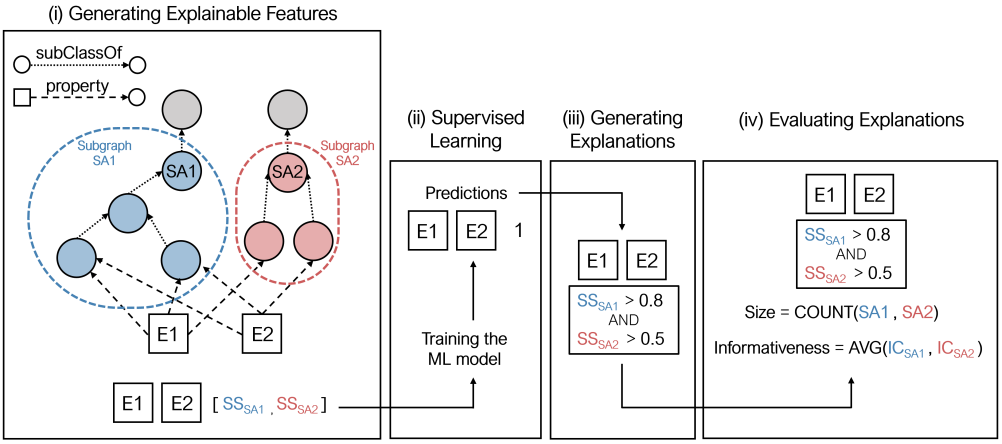


Figure 1: Overview of KGsim2vec with the main steps: (i) generating explainable features (ii) supervised learning (iii) generating explanations (iv) evaluating explanations.

3.2. Overview

We propose KGsim2vec, a novel method to generate explainable vector representations of entity pairs in a knowledge graph to support learning with minimal losses in performance when compared to opaque models. Our framework computes the explainable vector representations, then applies ML algorithms to generate predictive models, and finally generates explanations (represented in Figure 1).

The first step is generating explainable features. To do that, the knowledge graph is transformed into an RDF graph, which facilitates the subsequent processing. Then our approach extracts the knowledge graph semantic aspects to compute the semantic similarity and generate a pair representation. The second step is concerned with employing supervised learning methods to learn a relation prediction model taking as input the pair representation. The last steps correspond to generating and evaluating explanations for the predictions.

3.3. Generating Explainable Features

Our novel method, KGsim2vec, generates an explainable vector representation of entity pairs in a knowledge graph described according to the same ontology. The representation is based on the semantic similarities between the entities according to different semantic aspects of the ontology, i.e., subgraphs of the ontology at the same depth (Algorithm 1).

Algorithm 1 KGsim2vec

```
α ← minimum_feature_number
β ← minimum_height
γ ← minimum_coverage
function GET SA(classes)
    if β > 0 then
        classes ← FILTER CLASSES BY HEIGHT(classes)
    if γ > 0 then
        classes ← FILTER CLASSES BY COVERAGE(classes)
    if len(classes) ≥ α then
        return classes
    else
        new_classes ← ∅
        for c in classes do
            new_classes.append(GET SUBCLASSES(c))
        return GET SA(new_classes)

function GET SS SCORE(entity_1,entity_2,subgraph)
    a_1 ← GET ANNOTATIONS(entity_1, subgraph)
    a_2 ← GET ANNOTATIONS(entity_2, subgraph)
    return max(GET IC(GET MICA(c_1, c_2)): c_1 ∈ a_1, c_2 ∈ a_2

function GET EXPLAINABLE VECTORS(entity_pairs, ontology)
    vectors ← ∅          ▷ dictionary to hold explainable vectors for each entity pair
    root ← GET ROOT(ontology)
    semantic_aspects ← GET SA(root)
    for s in semantic_aspects do
        sg ← GET SUBGRAPH(s)          ▷ extracts an ontology subgraph rooted in s
        for e_1, e_2 in entity_pairs do
            vectors[e_1, e_2].append(GET SS SCORE(e_1,e_2,sg))
    return vectors
```

To extract the semantic aspects, we perform a breadth-first search on the ontology graph to find the depth (i.e. distance to the root(s)) at which the number of classes is greater than α and retrieve those classes. This parameter can be set to manipulate the size and consequently the level of detail afforded by the explainable vectors. Other criteria can also be explored to filter the subgraphs. In addition to depth, we also experimented with setting a minimum height (β) – i.e., distance to a leaf class – which would remove subgraphs of insufficient depth, as well as a minimum coverage (γ) – i.e., percentage of entities annotated in the semantic aspects – which removes

subgraphs that are seldom used to describe the entities.

Once the semantic aspects have been identified, the semantic similarities between each pair of entities are computed according to each semantic aspect. We employed the maximum pairwise similarity between all classes that annotate each entity. To measure class similarity, we computed the Information Content (IC) of the Most Informative Common Ancestor (MICA) between the classes. We employed $\mathrm{IC}_{\mathrm{Seco}}$ (Seco et al., 2004), a structure-based approach based on the number of direct and indirect descendants and given by

$$\mathrm{IC}_{\mathrm{Seco}}(c) = 1 - \frac{\log\left[\mathrm{N\_descendants}(c) + 1\right]}{\log\left[\mathrm{N\_classes}\right]} \tag{1}$$

where N_descendants(c) is the number of indirect and direct descendants of class c (including class c), and N_classes is the total number of classes in the ontology.

3.4. Supervised Learning

After obtaining the vector representations, we use ML algorithms to learn relation prediction models. We focus on representative tree-based ML algorithms: two interpretable models, Decision Trees (DT) and Genetic Programming (GP), and two black-box ones, Random Forest (RF) and eXtreme Gradient Boosting (XGB).

Decision trees (Denison et al., 1998) meet the characteristics of transparent models (algorithmic transparency, decomposability and simulatability) and are a familiar representation (Barredo Arrieta et al., 2020). Genetic programming (Koza, 1992) is an evolutionary computation technique inspired by Darwinian natural selection and Mendelian genetics and can return interpretable models by combining features, operators and numerical values (Mei et al., 2022). However, the size of the models can also influence interpretability. While learning over complex data, decision trees and genetic programming models may grow very large increasing the cognitive effort required to interpret the solutions. To tackle these challenges, we generated models that take into consideration their depth during training:

(i) DT6 where the maximum depth that trees are allowed to reach is 6;

(ii) GP6x with a fitness function that penalizes models with a depth greater than 6, and using only interpretable operators (i.e., maximum, minimum, addition and subtraction, since operators such as multiplication and division have a less straightforward meaning for interpretability).

Regarding random forest and extreme gradient boosting, they are ensemble models that combine the decisions from multiple decision trees. They are classified as black-box models and require post-hoc techniques.

We used scikit-learn[2] with default values for decision trees (criterion=gini; min samples split=2; min samples leaf=1), and gplearn 3.0[3] for genetic programming (generations=50; size of population=500; fitness function=RMSE; parsimony coefficient=10^{-5}; function set=[+,−,max,min] as used in Sousa et al. (2020)).

3.5. Generating Explanations

For interpretable models, the explanation is the model itself. For example, a decision tree is constructed by beginning with the root node that contains the whole learning sample and then splitting a node into two child nodes repeatedly. Each decision trees can be converted into a set of decision rules with the form: IF condition 1 AND condition 2 AND condition 3 AND ... THEN outcome, where the number of conditions is the number of decision nodes from root to leaf. The genetic programming models can also be converted to a mathematical formula easily interpretable, by reading their trees depth-first.

However, for the black-box models, this is not possible. Therefore, we added a surrogate model to produce local models to explain individual predictions. We employed two of the most well-known post-hoc explainability methods: LIME (Local Interpretable Model-Agnostic Explanations) (Ribeiro et al., 2016) and LORE (Local Rule-Based Explanations) (Guidotti et al., 2018).

LIME can explain the predictions of any classifier or regressor by approximating it with an interpretable model that is locally faithful to the ML model. This algorithm starts by randomly generating instances in the neighborhood of the instances to be explained, weighs them by the proximity to the instance, and then infers linear models as comprehensible local predictors. The LIME output is a list of weighted features in the format `(feature f > x, y)`, where `feature f > x` frames the value of feature `f` for that data sample and `y` is the contribution of feature `f` to the prediction of a data sample. The number of features that appear in the LIME explanation

[2]https://scikit-learn.org/

[3]https://gplearn.readthedocs.io/en/stable/

is given as a parameter. In our experiments, we tested two values for this parameter: 3 (LIME 3feat) and 8 (LIME 8feat).

Another similar approach to LIME is LORE. This method first generates a synthetic neighborhood on which it learns a local interpretable predictor. A local explanation is then extracted. The LORE output is more straightforward since it is a single decision rule which characterizes the conditions concerning the features' values for the decision of the black box. These rules are built by generating a set of neighbors of the data sample through a genetic algorithm and then extracting from such a set a decision tree.

3.6. Evaluating Explanations

To evaluate the explanations, we considered two aspects: size and informativeness. Since each feature represents the similarity for a specific ontology semantic aspect rooted in a specific ontology class, we measure the specificity of each feature according to the information content of the class (IC_{Seco}). The higher the information content value, the more informative this feature will be. The informativeness of an explanation is the average of the information content of the explanation features. A good explanation would then be composed of a few features to be easily understood (cognitive studies indicate humans are able to hold 7 ± 2 objects in short-term memory (Miller, 1956)), but those few features should be as informative as possible.

4. Experimental Results

KGsim2vec targets relation prediction tasks cast as a classification task that takes as input entity pairs and a knowledge graph back-boned by an ontology. The ontology is structured as a directed acyclic graph, where each class is linked to its ancestor through subclass relations. As a result, each class is more specific than its ancestors. Furthermore, these relations are transitive, indicating that they inherit all the ancestor classes up to the root. We evaluate KGsim2vec on protein-protein interaction prediction using the Gene Ontology knowledge graph. The data used are described in the following sections.

4.1. Data

The understanding of biological processes relies on the study of protein-protein interaction. However, experimental detection of protein-protein interactions is time-consuming and laborious. Despite proteins being annotated

with Gene Ontology (either for experimental evidence or automatically generated), only a limited number of protein interaction sites have been experimentally validated in current databases.

The target relations to predict are obtained from STRING (Szklarczyk et al., 2020). This database is one of the largest available protein-protein interaction databases that integrates physical interactions and functional associations between proteins collected from several sources. All interaction evidence is benchmarked and scored to estimate the confidence on whether a proposed association is biologically meaningful given all the contributing evidence. We considered the following criteria to select protein pairs:

(i) each protein must be annotated with the Gene Ontology;

(ii) protein interactions must be extracted from curated databases or experimentally determined (as opposed to computationally determined);

(iii) interactions must have a confidence score above 0.950 to retain only high confidence interaction.

The protein-protein interaction dataset contains 23571 interacting protein pairs and an equal number of negative pairs that have been generated through random negative sampling from the same pool of proteins.

The Gene Ontology knowledge graph describes proteins and is built by integrating the Gene Ontology (Consortium, 2021) and protein annotation data (Huntley et al., 2014). Gene Ontology (Consortium, 2021) defines the universe of classes, also called "Gene Ontology terms", associated with gene product (proteins or RNA) functions and how these functions are related to each other. Gene Ontology can be represented in terms of a graph, where the nodes represent Gene Ontology classes and the edges represent relationships between the classes (e.g., is_a; $part\_of$; has_part; $regulates$; $negatively\_regulates$ and $positively\_regulates$). As shown in Figure 2, functions in Gene Ontology are described concerning three aspects: the biological processes a gene product is involved in, the molecular functions a gene product executes, and the cellular components where a gene product is. The three Gene Ontology domains are represented as separated root ontology classes since they are unrelated and do not share any common parent node.

A Gene Ontology annotation (Huntley et al., 2014) is a statement about the function (or Gene Ontology class) of a particular gene product. Gene Ontology annotations and Gene Ontology together form the Gene Ontology

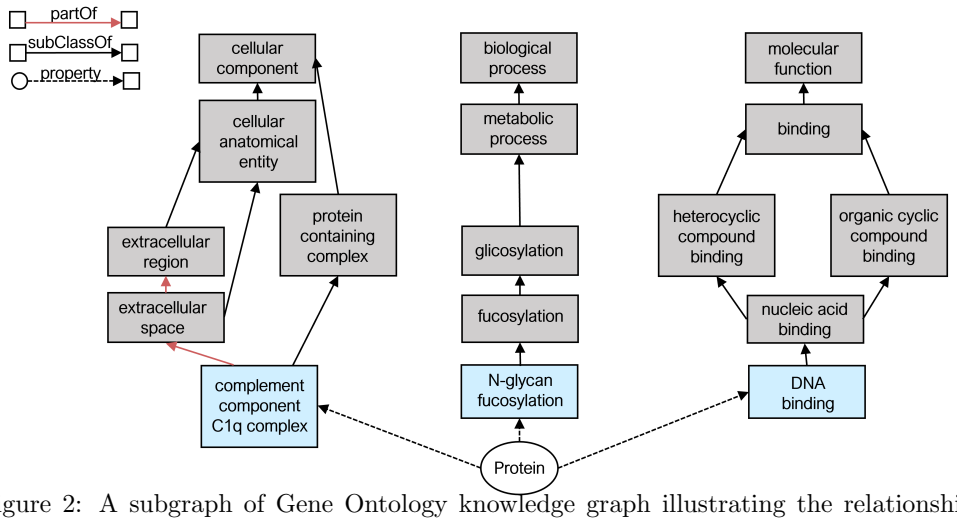


Figure 2: A subgraph of Gene Ontology knowledge graph illustrating the relationships between Gene Ontology classes and between proteins and Gene Ontology classes.

knowledge graph, where the graph's nodes represent proteins or Gene Ontology classes. The edges represent relationships between the Gene Ontology classes or links between proteins annotated with Gene Ontology classes. The Gene Ontology knowledge graph allows measuring the similarity between gene products by comparing the set of concepts they are annotated with. To avoid the potential for data circularity, we removed from the Gene Ontology knowledge graph the *'protein-containing_complex'* branch and the corresponding Gene Ontology annotations.

4.2. Performance Evaluation

We evaluated the predictive performance of KGsim2vec against popular knowledge graph embedding approaches and Graph Neural Networks (GNN). The knowledge graph embeddings were generated using the whole knowledge graph and three knowledge graph embedding approaches: RDF2Vec (Ristoski and Paulheim, 2016), OWL2Vec* (Chen et al., 2021) and GO2vec (Zhong et al., 2019) (an application of node2vec to the Gene Ontology). Then, the vectors representing each protein are combined using the Hadamard operator and given as input to the ML approaches. For the graph neural networks, we used the framework proposed in Lin et al. (2020), which encodes each protein's semantic features based on their neighbourhood and the interaction is predicted based on the learned embeddings. Each model was evaluated using

10-fold cross-validation and, for each fold, the weighted average of F-measures was computed. This metric accounts for class imbalance by computing the F-measure for each class and then calculating the average of all computed F-measures, weighted by the number of instances of each class.

In Table 1, we report the median weighted average of F-measures and the interquartile range of the 10 weighted average of F-measures values for different ML algorithms using our explanainable representation method or the knowledge graph embeddings. Table 2 reports the median weighted average of F-measures and the interquartile range using a graph neural network. Our method was applied with a straightforward set of parameters ($\alpha = 10$, $\beta = 0$ and $\gamma = 0$). Statistically significant differences are determined using pairwise non-parametric Kruskal-Wallis tests at $p < 0.01$.

Table 1: Weighted average of F-measures medians (M) and interquartile ranges (IQR) using our KGsim2vec or the embeddings coupled with different ML approaches. The best result for each ML approach is in bold. KGsim2vec performance values are italicized/underlined when improvements are statistically significant.

	RF		**XGB**		**DT**		**DT6**		**GP**		**GP6x**	
	M	IQR	M	IQR	M	IQR	M	IQR	M	IQR	M	IQR
KGsim2vec	***0.919***	0.005	0.915	0.004	***0.899***	0.003	***0.906***	0.002	***0.866***	0.005	***0.866***	0.006
RDF2Vec	0.904	0.004	**0.917**	0.009	0.783	0.007	0.747	0.006	0.756	0.005	0.776	0.021
OWL2Vec*	0.861	0.002	0.873	0.004	0.710	0.011	0.683	0.007	0.656	0.035	0.693	0.021
GO2Vec	0.881	0.007	0.904	0.002	0.715	0.005	0.687	0.016	0.728	0.016	0.749	0.011

Table 2: Weighted average of F-measures medians (M) and interquartile ranges (IQR) using a graph neural network.

	M	IQR
GNN	0.815	0.007

The results in Table 1 show that our method outperforms RDF2Vec when combined with all ML methods except extreme gradient boosting. These differences are statistically significant.

4.3. Explanations Evaluation

Knowledge graph embeddings are, of course, non-explainable, since each feature does not have a particular meaning. As such, this evaluation focuses on comparing the explanations generated by applying our method both with interpretable ML approaches (DT6 and GP6x) and surrogate methods over

non-interpretable ones (LIME and LORE over random forest and extreme gradient boosting).

Figure 3 shows that the explanations generated by DT6 and LORE are the smallest and least informative, whereas GP6x finds a compromise between the number of features and their informativeness. LIME's performance is dependent on defining the number of features to consider: when using fewer features, informativeness drops below that obtained by GP6x, and approximates DT6 when the same explanation size is considered, but for a similar size to GP6X (8 features), its explanations are the most informative.

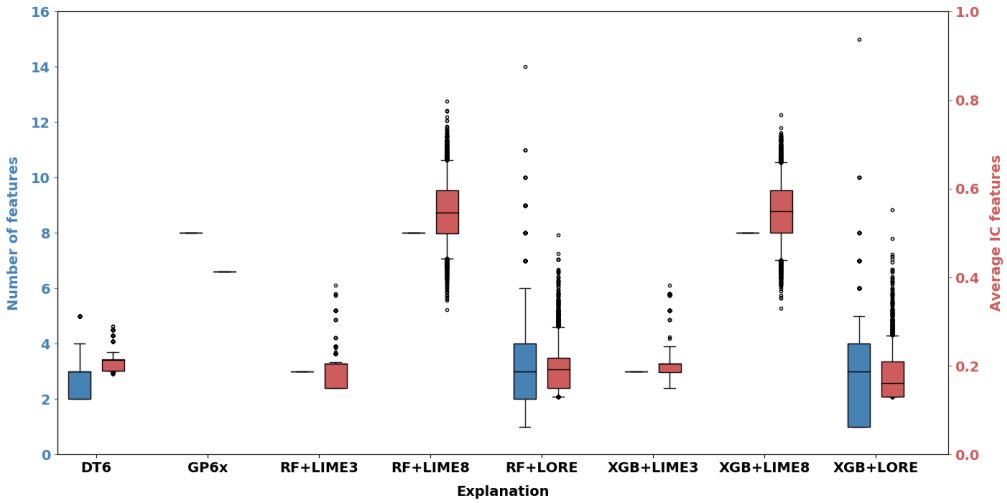


Figure 3: Size and informativeness of the explanations obtained for the first partition samples.

4.4. Explanations by Example

Explanations by example consider the extraction of representative data examples related to the result generated by a specific model, enabling a better understanding of the model itself.

In Tables 3 to 8, present explanations for six protein pairs chosen randomly from the protein-protein interaction dataset representing possible outcomes (true positive, false positive, true negative, and false negative). The tables present a chart with semantic similarity measured for the most relevant semantic aspects, a short description of the interaction status and the

generated explanations with their size and informativeness. Since in all selected examples, extreme gradient boosting and random forest agreed on the predictions resulting in equivalent explanations, only random forest is shown. However, two runs were performed for LORE since its explanations for the same instance and ML model can vary between runs due to the stochastic neighbor generating strategy it employs.

4.4.1. 40S Ribosomal Protein S12 and 40S Ribosomal Protein S10

40S ribosomal protein S12[4] and 40S ribosomal protein S10[5] make up the first pair (Table 3). The eukaryotic small ribosomal subunits (40S) play a central role in protein translation since it contains the decoding centre where mRNA codons are recognized by complementary anticodons of tRNAs bearing amino acid residues for protein synthesis. Multiple binding sites characterize this subunit.

These proteins have 12 direct annotations in common, namely two specific biological processes classes: *"nuclear-transcribed_mRNA _catabolic_process, nonsense-mediated_decay"* class (defined as a nonsense-mediated decay pathway that prevents the translation of mRNAs into potentially harmful proteins); *"SRP-dependent_cotranslational_ protein_targeting_to_ membrane"* class (SRP is a cytosolic particle that transiently binds to the endoplasmic reticulum and it is essential for the targeting of proteins to a membrane in translation). For all analyzed models, we verified that the high similarity computed for the *"metabolic_process"* and *"cellular_process"* semantic aspects always appears in the explanations. These explanations are in agreement with the fact that the two proteins participate in the same metabolic processes.

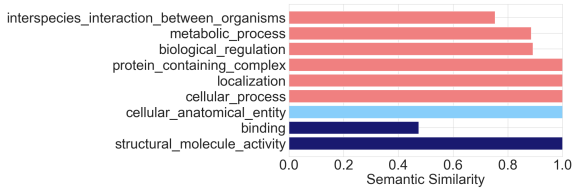
4.4.2. Neuroblast Differentiation-associated Protein AHNAK and Protein S100-A10

The neuroblast differentiation-associated protein AHNAK[6] and the protein S100-A10[7] constitute the second pair (Table 4). Neuroblast differentiation-associated protein AHNAK is a sizeable structural scaffold protein that may play a role in such diverse processes as blood-brain barrier formation, cell structure and migration, cardiac calcium channel regulation, and tumour

[4]https://www.uniprot.org/uniprot/P25398
[5]https://www.uniprot.org/uniprot/P46783
[6]https://www.uniprot.org/uniprot/Q09666
[7]https://www.uniprot.org/uniprot/P60903

Table 3: Explanations of ML models for the 40S ribosomal protein S12 – 40S ribosomal protein S10 positive pair.

40S ribosomal protein S12 – 40S ribosomal protein S10



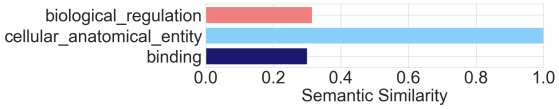
40S ribosomal protein S12 and 40S ribosomal protein S10 are components of the 40S ribosomal subunit that plays a central role in protein translation and is characterized by multiple binding sites.

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (+)	$SS_{\text{metabolic\_process}} > 0.5167$ AND $SS_{\text{cellular\_process}} > 0.8032$	2	0.161
GP6x (+)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}) \geq 0.5$	9	0.423
LIME 3feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4694), (SS_{\text{metabolic\_process}} > 0.61, 0.2611), (SS_{\text{biological\_regulation}} > 0.57, 0.1752)$	3	0.151
LIME 8feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4800), (SS_{\text{metabolic\_process}} > 0.61, 0.2656), (SS_{\text{pigmentation}} \leq 0, -0.2017), (SS_{\text{biological\_regulation}} > 0.57, 0.1929), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1490), (SS_{\text{protein\_folding\_chaperone}} \leq 0, -0.1440), (SS_{\text{molecular\_carrier\_activity}} \leq 0, 0.1256), (SS_{\text{multi-organism\_process}} \leq 0, -0.1233)$	8	0.466
LORE 1 (+)	$SS_{\text{biological\_regulation}} > -0.1206$ AND $SS_{\text{response\_to\_stimulus}} \leq 0.1863$ AND $SS_{\text{biological\_adhesion}} \leq 0.0056$ AND $SS_{\text{cellular\_process}} > 0.5715$	4	0.278
LORE 2 (+)	$SS_{\text{metabolic\_process}} > 0.7834$	1	0.190

metastasis. Protein S100-A10 is an integral part of cellular structural scaffolding that interacts with plasma membrane proteins through its association with annexin II.

The protein pair share four direct annotations since they both have the same function (*"protein_binding"* and are localized in the same cellular components, namely cytoplasm, extracellular exosome and membrane raft. Although the proteins share some semantic annotations, they are very general. Therefore, all the analyzed ML methods fail to predict this protein pair interaction. However, according to the literature, they are likely involved in the mediated organization of the actin cytoskeleton (Hayes et al., 2004). Both proteins are poorly described under the Gene Ontology, which may explain why the ML models fail.

Table 4: Explanations of ML models for the S100-A10 – neuroblast differentiation-associated protein positive pair.

S100-A10 protein – neuroblast differentiation-associated protein			
			Protein S100-A10 is an integral part of cellular structural scaffolding that works together with neuroblast differentiation-associated protein AHNAK, a membrane-associated protein, in the development of the intracellular membrane.
Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (−)	if $SS_{\text{binding}} \leq 0.4693$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.1162$ AND $SS_{\text{cellular\_process}} \leq 0.0399$ AND $SS_{\text{biological\_regulation}} \leq 0.5484$	4	0.216
GP6x (−)	$\max(SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}, SS_{\text{signaling}}, SS_{\text{translation\_regulator\_activity}}) < 0.5$	9	0.442
LIME 3feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3510), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1522), (SS_{\text{metabolic\_process}} \leq 0, -0.1211)$	3	0.206
LIME 8feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3128), (SS_{\text{molecular\_carrier\_activity}} \leq 0, -0.2570), (SS_{\text{detoxification}} \leq 0, -0.1822), (SS_{\text{intraspecies\_interaction\_between\_organisms}} \leq 0, -0.1779), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1487), (SS_{\text{metabolic\_process}} \leq 0, -0.1299), (SS_{\text{molecular\_adaptor\_activity}} \leq 0, -0.1126), (SS_{\text{growth}} \leq 0, 0.0456)$	8	0.479
LORE 1 (−)	$SS_{\text{cellular\_process}} \leq 0.6046$ AND $SS_{\text{metabolic\_process}} \leq 0.5021$ AND $SS_{\text{biological\_regulation}} \leq 0.5970$ AND $SS_{\text{binding}} \leq 0.4632$	4	0.189
LORE 2 (−)	$SS_{\text{cellular\_process}} \leq 0.4916$ AND $SS_{\text{metabolic\_process}} \leq 0.5396$ AND $SS_{\text{biological\_regulation}} \leq 0.5419$ AND $SS_{\text{binding}} \leq 0.4449$ AND $SS_{\text{multicellular\_organismal\_process}} \leq 0.0522$ AND $SS_{\text{localization}} \leq 0.2706$	6	0.235

4.4.3. Proline Protein and Guanine Nucleotide-binding Protein

The third example pair is composed of proline protein[8] and a guanine nucleotide-binding protein[9] (Table 5). The proline-rich protein 5-like associates with the mTORC2 complex that regulates cellular processes, including survival and organization of the cytoskeleton. The guanine nucleotide-binding protein-like 3 is a GTPase binding nuclear protein that has been reported to be involved in various biological processes, including cell proliferation, cellular senescence and tumorigenesis.

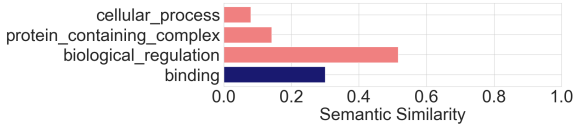
Although the two proteins have annotations for the three Gene Ontology domains, they have only one direct annotation in common: the molecular function class *"protein_binding"*. Furthermore, they only have in common that they are both involved in the negative regulation of the protein modification process. In summary, these two proteins have binding functions

[8]https://www.uniprot.org/uniprot/Q6MZQ0
[9]https://www.uniprot.org/uniprot/Q9NVN8

but they do not participate in the same biological processes, translating into low similarity values for several semantic aspects and explaining why ML algorithms do not predict interaction.

Table 5: Explanations of ML models for the Proline-rich 5-like – Guanine nucleotide-binding 3-like negative pair.

Proline-rich 5-like – Guanine nucleotide-binding 3-like



The proline-rich protein 5-like associates with the mTORC2 complex that regulates the organization of the cytoskeleton. In opposition, guanine nucleotide-binding protein-like 3 is a GTPase-binding nuclear protein.

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (−)	$SS_{\text{binding}} \leq 0.4693$ AND $0.0399 < SS_{\text{cellular\_process}} \leq 0.6145$ AND $SS_{\text{biological\_regulation}} \leq 0.5484$	3	0.189
GP6x (−)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}, SS_{\text{translation\_regulator\_activity}}) < 0.5$	10	0.454
LIME 3feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3192), (SS_{\text{cellular\_anatomical\_entity}} \leq 0.53, -0.1842), (SS_{\text{metabolic\_process}} \leq 0, -0.1227)$	3	0.206
LIME 8feat (−)	$(SS_{\text{cellular\_process}} \leq 0.16, -0.3109), (SS_{\text{molecular\_carrier\_activity}} \leq 0, 0.2230), (SS_{\text{cellular\_anatomical\_entity}} \leq 0.53, -0.1621), (SS_{\text{molecular\_adaptor\_activity}} \leq 0, -0.1319), (SS_{\text{metabolic\_process}} \leq 0, -0.1259), (SS_{\text{biomineralization}} \leq 0, -0.1256), (SS_{\text{structural\_molecule\_activity}} \leq 0, -0.1197), (SS_{\text{multicellular\_organismal\_process}} \leq 0, -0.1090)$	8	0.464
LORE 1 (−)	$SS_{\text{cellular\_process}} \leq 0.4285$ AND $SS_{\text{metabolic\_process}} \leq 0.7601$ AND $SS_{\text{biological\_regulation}} \leq 0.7260$	3	0.151
LORE 2 (−)	$SS_{\text{cellular\_process}} \leq 0.6524$ AND $SS_{\text{metabolic\_process}} \leq 0.6210$ AND $SS_{\text{biological\_regulation}} \leq 0.8630$	3	0.151

4.4.4. Protransforming Growth Factor α and Disks Large Homolog 2

The fourth pair is composed by protransforming growth factor (TGF) α[10] and disks large homolog 2[11] (Table 6) and correspond to a false positive. TGF-α is a mitogenic polypeptide that acts synergistically with TGF-β to promote anchorage-independent cell proliferation. Disks large homolog 2 is a member of the membrane-associated guanylate kinase (MAGUK) family and forms a heterodimer with a related family member that may interact

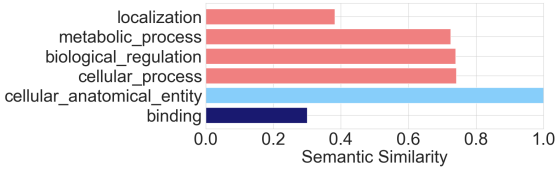
[10]https://www.uniprot.org/uniprot/P01135
[11]https://www.uniprot.org/uniprot/Q15700

at postsynaptic sites to form a postsynaptic protein scaffold of excitatory synapses.

Regarding Gene Ontology annotations, the two proteins are localized in the basolateral plasma membrane and participate in MAPK cascade, which contributes to a high semantic similarity for several semantic aspects. All the models wrongly predict an interaction. These predictions are justified by high similarity values for the most relevant semantic aspects. Curiously, although no information on their interaction was found in the literature, TGF-β is regulated by Dlg5 and both proteins activate the MAPK cascade (Sezaki et al., 2013). This led us to think that this is not a true negative pair, but rather an unknown interaction that was mistakenly used as a negative example via random negative sampling.

Table 6: Explanations of ML models for protransforming growth factor α – Disks large homolog 2 negative pair.

Protransforming growth factor α – Disks large homolog 2



There is no evidence of interaction between these proteins, but there is evidence of an interaction between highly similar proteins: TGF-β is regulated by Dlg5 and both proteins activate the MAPK cascade (Sezaki et al., 2013).

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (+)	$SS_{\text{cellular\_anatomical\_entity}} > 0.7589$ AND $SS_{\text{metabolic\_process}} > 0.6023$ AND $0.6145 < SS_{\text{cellular\_process}} \leq 0.7531$	3	0.206
GP6x (+)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}) \geq 0.5$	9	0.423
LIME 3feat (+)	$(SS_{\text{metabolic\_process}} > 0.61, 0.2603), (SS_{\text{biological\_regulation}} > 0.57, 0.1788), (0.49 < SS_{\text{cellular\_process}} \leq 0.84, 0.1702)$	3	0.151
LIME 8feat (+)	$(SS_{\text{metabolic\_process}} > 0.61, 0.2552), (SS_{\text{biological\_regulation}} > 0.57, 0.1876), (SS_{\text{protein\_tag}} \leq 0, -0.1700), (0.49 < SS_{\text{cellular\_process}} \leq 0.84, 0.1639), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1, 0.1626), (SS_{\text{translation\_regulator\_activity}} \leq 0, 0.1258), (SS_{\text{intraspecies\_interaction\_between\_organisms}} \leq 0, -0.0989), (SS_{\text{biomineralization}} \leq 0, 0.0486)$	8	0.494
LORE 1 (+)	$SS_{\text{cellular\_process}} > 0.5126$	1	0.131
LORE 2 (+)	$SS_{\text{cellular\_anatomical\_entity}} > 0.8126$ AND $SS_{\text{molecular\_adaptor\_activity}} \leq 0.0179$ AND $SS_{\text{cellular\_process}} > 0.5870$ AND $SS_{\text{response\_to\_stimulus}} \leq 0$ AND $SS_{\text{localization}} > 0.3259$ AND $SS_{\text{multicellular\_organismal\_process}} \leq 0.0254$	6	0.326

4.4.5. Paxillin and Integrin α-4 positive

The next positive pair is paxillin[12] and integrin α-4[13] (Table 7). Paxillin is a signal transduction adaptor protein that helps cells to the extracellular matrix. Integrin α-4 is essential for immune response, embryogenesis, hematopoiesis, and inflammation. There is evidence in the literature of the link between these proteins Han et al. (2001); Liu et al. (2002). The cytoplasmic domain of α-4 binds tightly to paxillin, leading to increased cell migration and an altered cytoskeletal organization reduces cell spreading.

Regarding the GO annotations, paxillin and integrin α-4 do not perform the same metabolic processes as the first pair. However, they are both located in *"focal_adhesion"* (a small region on the surface of a cell that anchors the cell to the extracellular matrix) and participate in the *"substrate_adhesion-dependent_cell_spreading"* (a BP that results in flattening of a cell as a consequence of its adhesion to a substrate). The explanations seem to corroborate our knowledge as they show that the two proteins interact as they participate in the same cellular processes (even if they do not participate in the same metabolic processes).

4.4.6. 26S proteasome regulatory subunit 4 and Proteasome subunit β type-3

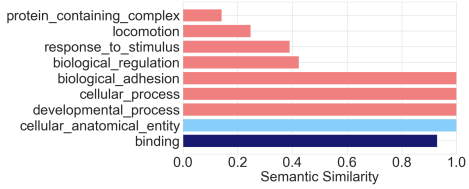
26S proteasome regulatory subunit 4[14] and proteasome subunit β type-3[15] make up the last pair (Table 8). 26S proteasome regulatory subunit 4 is a component of the 26S proteasome, a multiprotein complex involved in the ATP-dependent degradation of ubiquitinated proteins. Proteasome subunit beta type-3 is a non-catalytic component of the 20S core proteasome complex involved in the proteolytic degradation of most intracellular proteins. The 20S core proteasome complex associates with two 19S regulatory particles and forms the 26S proteasome.

These proteins have 29 direct annotations in common, namely specific biological regulation classes: *"regulation of mitotic cell cycle phase transition"* class; *"positive regulation of canonical Wnt signaling pathway"* class; *"regulation of hematopoietic stem cell differentiation"* class; *"negative regulation of canonical Wnt signaling pathway"* class. We verified that the high similarity computed for the *"cellular_process"* semantic aspect for all analysed models

[12]https://www.uniprot.org/uniprot/P49023
[13]https://www.uniprot.org/uniprot/P13612
[14]https://www.uniprot.org/uniprot/P62191
[15]https://www.uniprot.org/uniprot/P49720

Table 7: Explanations of ML models for the Paxillin – Integrin α-4 positive pair.

Paxillin – Integrin α-4



Paxillin is a signal transduction adaptor protein that binds to the cytoplasmic domain of α-4, being involved in cytoskeletal organization.

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (+)	$SS_{\text{metabolic\_process}} \leq 0.6048$ AND $SS_{\text{cellular\_process}} > 0.8535$	2	0.161
GP6x (+)	$\max(SS_{\text{behavior}}, SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{immune\_system\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{structural\_molecule\_activity}}, SS_{\text{translation\_regulator\_activity}})$	10	0.454
LIME 3feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4703), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1.00, 0.1502), (SS_{\text{metabolic\_process}} \leq 0.00, -0.1312)$	3	0.206
LIME 8feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4665), (SS_{\text{intraspecies\_interaction\_between\_organisms}} \leq 0.00, -0.2148), (SS_{\text{molecular\_carrier\_activity}} \leq 0.00, -0.1549), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1.00, 0.1463), (SS_{\text{pigmentation}} \leq 0.00, -0.1377), (SS_{\text{metabolic\_process}} \leq 0.00', -0.1253), (SS_{\text{protein\_folding\_chaperone}} \leq 0.00, 0.0864), (SS_{\text{antioxidant\_activity}} \leq 0.00, 0.0843)$	8	0.560
LORE 1 (+)	$SS_{\text{cellular\_process}} > 0.6890$	1	0.131
LORE 2 (+)	$SS_{\text{cellular\_process}} > 0.5213$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.6059$	2	0.213

always appears in the explanations.

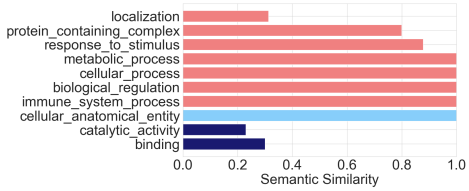
4.5. Frequent Rules Analysis

Table 9 shows the most frequent DT6 rules across different models and the number of pairs explained by those rules. Since decision trees do not always learn the same cutoff values, after individual models, the cutoff values were rounded to one decimal place to ensure that similar rules with slight variations in cutoff values are treated as the same. Once the cutoff values are standardized, the frequency of each decision tree rule is computed across the different models. Table 9 shows the rules that appear in at least half of the models.

Many rules are clearly supported by existing scientific knowledge, for instance, rule 2indicates that high values in $SS_{\text{metabolic\_process}}$ and $SS_{\text{cellular\_process}}$ imply an interaction, which makes sense. Proteins participating in the same metabolic or cellular process are very likely to interact. For somewhat lower $SS_{\text{cellular\_process}}$ values, a positive interaction now requires a high *biologi-*

Table 8: Explanations of ML models for the 26S proteasome regulatory subunit 4 – proteasome subunit β type-3 positive pair.

26S proteasome regulatory subunit 4 – Proteasome subunit β type-3



26S proteasome regulatory subunit 4 and proteasome subunit β type-3 are components of the 26S proteasome that play a key role in maintaining protein homeostasis.

Model (pred.)	**Explanation**	**Size**	**IC**
DT6 (+)	$SS_{\text{cellular\_anatomical\_entity}} > 0.7973$ AND $SS_{\text{cellular\_process}} > 0.7531$	2	0.214
GP6x (+)	$\max(SS_{\text{catalytic\_activity}}, SS_{\text{cellular\_process}}, SS_{\text{interspecies\_interaction\_between\_organisms}}, SS_{\text{molecular\_transducer\_activity}}, SS_{\text{multicellular\_organismal\_process}}, SS_{\text{translation\_regulator\_activity}}, SS_{\text{molecular\_adaptor\_activity}} + SS_{\text{molecular\_function\_regulator}})$	8	0.413
LIME 3feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4711), (SS_{\text{metabolic\_process}} > 0.61, 0.2673), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1.00, 0.1569)$	3	0.206
LIME 8feat (+)	$(SS_{\text{cellular\_process}} > 0.84, 0.4662), (SS_{\text{metabolic\_process}} > 0.61, 0.2682), (SS_{\text{biological\_regulation}} > 0.57, 0.1691), (0.86 < SS_{\text{cellular\_anatomical\_entity}} \leq 1.00, 0.1529), (SS_{\text{detoxification}} \leq 0.00, -0.1437), (SS_{\text{molecular\_adaptor\_activity}} \leq 0.00, 0.1338), (SS_{\text{structural\_molecule\_activity}} \leq 0.00, -0.1316), (SS_{\text{pigmentation}} \leq 0.00, 0.0638)$	8	0.422
LORE 1 (+)	$SS_{\text{cellular\_process}} > 0.7072$	1	0.131
LORE 2 (+)	$SS_{\text{cellular\_process}} > 0.4648$ AND $SS_{\text{biological\_regulation}} > 0.8409$	2	0.131

cal_regulation (rule 3). Another rule in compliance with biological knowledge is rule 1, which takes into account the low values of $SS_{\text{metabolic\_process}}$, $SS_{\text{cellular\_process}}$, and $SS_{\text{biological\_regulation}}$ to indicate that the interaction between the two proteins is not likely. However, some of these general rules do not reflect biology but likely capture the incidental characteristics of the underlying data. For instance, rule 17classifies positive interactions as those with very low similarity scores in several features, which is probably an attempt to classify poorly annotated proteins (on average, each protein of the dataset has around 23 annotations). This hypothesis justifies that, although the rules appear in most models, the number of protein pairs for which those rules apply is low. In contrast, rules applied to a higher number of pairs seem to be capturing the natural phenomenon.

In addition to the most frequent rules encoding biological information relevant to the protein-protein interaction predictions, sometimes they also capture the phenomenon of functional annotation. The interaction between a pair can be predicted even if it has low similarity values due to the poorly

Table 9: Analysis of the most frequent rules across different DT6 models and using the similarity for 51 semantic aspects as input.

	Rule	**Pred.**	**#Models**	**#Pairs**
1	IF $SS_{\text{binding}} \leq 0.5$ AND $0.0 < SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	–	10	17216
2	IF $SS_{\text{metabolic\_process}} > 0.6$ AND $SS_{\text{cellular\_process}} > 0.8$	+	5	4837
3	IF $SS_{\text{metabolic\_process}} \leq 0.6$ AND $SS_{\text{cellular\_process}} > 0.9$	+	5	2359
4	IF $SS_{\text{cellular\_anatomical\_entity}} > 0.8$ AND $0.6 < SS_{\text{cellular\_process}} \leq 0.8$	+	6	2082
5	IF $SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.0$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	–	10	1403
6	IF $SS_{\text{binding}} \leq 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.7$ AND $SS_{\text{cellular\_process}} \leq 0.5$ AND $0.5 < SS_{\text{biological\_regulation}} \leq 0.9$	–	10	841
7	IF $0.5 < SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.8$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	10	574
8	IF $SS_{\text{binding}} \leq 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.7$ AND $0.5 < SS_{\text{cellular\_process}} \leq 0.6$ AND $0.5 < SS_{\text{biological\_regulation}} \leq 0.9$	+	10	515
9	IF $SS_{\text{binding}} > 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.7$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $0.5 < SS_{\text{biological\_regulation}} \leq 0.9$	+	8	472
10	IF $0.5 < SS_{\text{binding}} \leq 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.9$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	–	6	350
11	IF $SS_{\text{binding}} > 0.7$ AND $SS_{\text{cellular\_anatomical\_entity}} > 0.9$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	6	268
12	IF $SS_{\text{binding}} \leq 0.1$ AND $SS_{\text{cellular\_anatomical\_entity}} \leq 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.0$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	6	248
13	IF $0.5 < SS_{\text{binding}} \leq 0.5$ AND $0.1 < SS_{\text{cellular\_anatomical\_entity}} \leq 0.8$ AND $SS_{\text{cellular\_process}} \leq 0.6$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	–	10	217
14	IF $SS_{\text{binding}} \leq 0.5$ AND $0.1 < SS_{\text{cellular\_anatomical\_entity}} \leq 0.8$ AND $0.6 < SS_{\text{cellular\_process}} \leq 0.8$ AND $SS_{\text{biological\_regulation}} \leq 0.7$	–	7	210
15	IF $0.1 < SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} \leq 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.0$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	–	6	157
16	IF $SS_{\text{cellular\_anatomical\_entity}} \leq 0.8$ AND $0.6 < SS_{\text{cellular\_process}} \leq 0.8$ AND $SS_{\text{biological\_regulation}} > 0.7$	+	6	101
17	IF $0.5 < SS_{\text{binding}} \leq 0.5$ AND $SS_{\text{cellular\_anatomical\_entity}} \leq 0.1$ AND $SS_{\text{cellular\_process}} \leq 0.1$ AND $SS_{\text{biological\_regulation}} \leq 0.5$	+	7	93
18	IF $SS_{\text{cellular\_anatomical\_entity}} \leq 0.7$ AND $0.0 < SS_{\text{cellular\_process}} \leq 0.5$ AND $SS_{\text{biological\_regulation}} > 0.7$	–	5	86

annotated proteins. These results reinforce the need for interpretability and explainability to understand what is actually being learned.

4.6. Ablation Studies

We experimented with β and γ to filter out leaf classes ($\beta = 1$ or classes that annotated less than 1% of the proteins ($\gamma = 0.01$). Table 10 shows that there are no significant differences in performance when employing these filters and effectively reducing the number of semantic aspects considered.

The impact on explanation size and informativeness is shown in Figures 4 and 5. The informativeness of the longer explanations given by GP6x and LIME decreases for both variants, indicating that although performance is not modified by filtering out leaf classes or low annotations classes, this can have an impact on explainability.

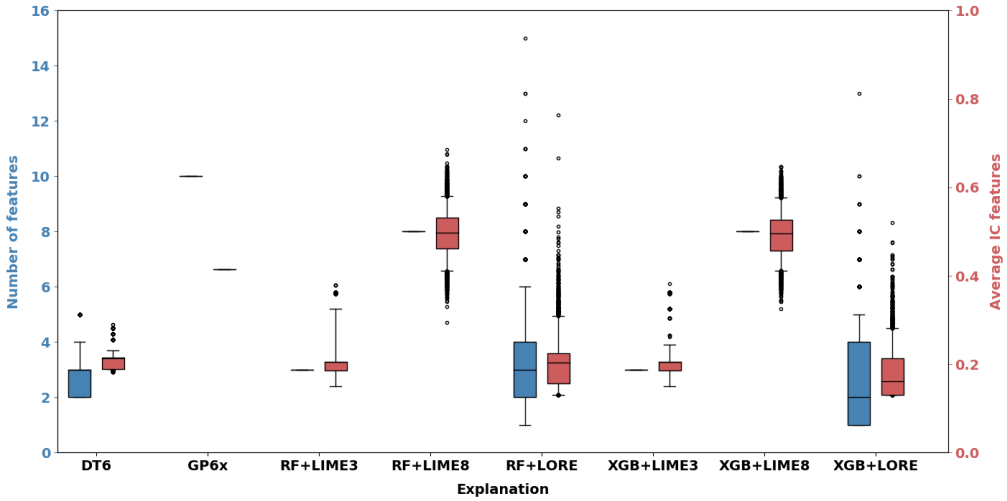


Figure 4: Size and informativeness of the explanations obtained for the first partition samples with $\beta = 1$.

Table 10: Weighted average of F-measures medians (M) and interquartile ranges (IQR) using different parameters for KGsim2vec.

α, β, γ	SAs	RF		XGB		DT		DT6		GP		GP6x	
		M	IQR	M	IQR	M	IQR	M	IQR	M	IQR	M	IQR
10, 0, 0	51	0.919	0.005	0.915	0.004	0.899	0.003	0.906	0.002	0.866	0.005	0.866	0.006
10, 1, 0	42	0.920	0.004	0.915	0.004	0.899	0.003	0.906	0.002	0.869	0.008	0.866	0.004
10, 0, 0.01	24	0.919	0.005	0.915	0.004	0.899	0.002	0.906	0.002	0.878	0.019	0.867	0.005

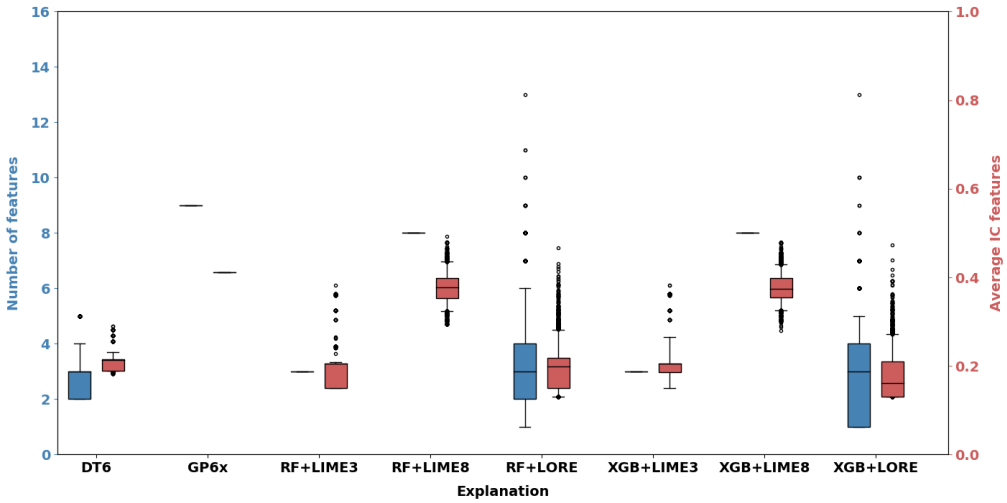


Figure 5: Size and informativeness of the explanations obtained for the first partition samples with $\gamma = 0.01$.

5. Conclusion

Explainability is crucial to support the adoption of ML as a scientific tool that helps understand natural phenomena and drives hypotheses. In the biomedical domain, the abundance of data described with ontologies and integrated into knowledge graphs affords a unique opportunity to explore domain knowledge to improve the explainability of ML applications. However, most state-of-the-art ML approaches based on knowledge graphs employ knowledge graph embeddings, which are not explainable. When the prediction target is finding a relation between two entities represented in the knowledge graph, similarity presents itself as a natural explanatory mechanism. Ontologies and knowledge graphs can support similarity computation, and measuring similarity between entity pairs according to different aspects represented in the knowledge graph opens the door to elucidate relevant aspects behind relations between entities.

Our novel method, KGsim2vec, generates vector representations of entity pairs in a knowledge graph that can be used to explain relations between them. The explanations are based on computing semantic similarity according to different aspects represented in the knowledge graph. The quality of an explanation combines the standard approach based on the number of aspects in the informativeness with their informativeness as measured by the

information content captured by each aspect. We evaluate KGsim2vec on protein-protein interaction prediction, a very relevant application of knowledge graph-based supervised learning in the biomedical domain. The experimental results have shown that KGsim2vec can outperform opaque representations given by knowledge graph embeddings or end-to-end deep learning approaches. In addition, it generates interesting explanations that capture biological phenomena and gaps in the current knowledge.

Recently, protein language models like ProteinBERT (Brandes et al., 2022) and ProtTrans (Elnaggar et al., 2021) have emerged to learn protein representations using protein sequences. Complementing these advancements, some approaches, such as OntoProtein (Zhang et al., 2021a), integrated knowledge graphs and protein sequences and achieved promising results. A possible direction for future work lies in expanding our approach to incorporate additional sources of information.

Acknowledgements

The authors thank Daniel Faria for the comments that significantly improved the manuscript.

Funding information

C.P., S.S., and R.T.S. are funded by FCT, Portugal, through LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020). R.T.S. acknowledges the FCT PhD grant (ref. SFRH/BD/145377/2019). This work was also partially supported by the KATY project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017453, and in part by projet 41, HfPT: Health from Portugal, funded by the Portuguese Plano de Recuperação e Resiliência.

References

Abdelaziz, I., Fokoue, A., Hassanzadeh, O., Zhang, P., Sadoghi, M., 2017. Large-scale structural and textual similarity-based mining of knowledge graph to predict drug–drug interactions. Journal of Web Semantics 44, 104 – 117. Industry and In-use Applications of Semantic Technologies.

Anguita-Ruiz, A., Segura-Delgado, A., Alcalá, R., Aguilera, C.M., Alcalá-Fdez, J., 2020. XAI for the identification of biologically relevant gene expression patterns in longitudinal human studies, insights from obesity research. PLOS Computational Biology 16.

Asif, M., Martiniano, H.F.M.C.M., Vicente, A.M., Couto, F.M., 2018. Identifying disease genes using machine learning and gene functional similarities, assessed through gene ontology. PLOS ONE 13, 1–15.

Bandyopadhyay, S., Mallick, K., 2017. A new feature vector based on gene ontology terms for protein-protein interaction prediction. IEEE/ACM Trans. on Computational Biology and Bioinformatics 14, 762–770.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., Herrera, F., 2020. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information Fusion 58, 82–115.

Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., Yakhnenko, O., 2013. Translating embeddings for modeling multi-relational data, in: Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2, Curran Associates Inc., Red Hook, NY, USA. p. 2787–2795.

Bourgeais, V., Zehraoui, F., Hanczar, B., 2022. GraphGONet: a self-explaining neural network encapsulating the gene ontology graph for phenotype prediction on gene expression. Bioinformatics Btac147.

Brandes, N., Ofer, D., Peleg, Y., Rappoport, N., Linial, M., 2022. Proteinbert: a universal deep-learning model of protein sequence and function. Bioinformatics 38, 2102–2110.

Cai, H., Zheng, V.W., Chang, K.C., 2018. A comprehensive survey of graph embedding: problems, techniques, and applications. IEEE Transactions on Knowledge and Data Engineering 30, 1616–1637.

Chari, S., Gruen, D.M., Seneviratne, O., McGuinness, D.L., 2020. Foundations of explainable knowledge-enabled systems, in: Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges. IOS Press, pp. 23–48.

Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O.M., Antonyrajah, D., Horrocks, I., 2021. OWL2Vec*: embedding of OWL ontologies. Machine Learning , 1–33.

Chen, K.H., Wang, T.F., Hu, Y.J., 2019. Protein-protein interaction prediction using a hybrid feature representation and a stacked generalization scheme. BMC Bioinformatics 20, 308.

Consortium, G., 2021. The gene ontology resource: enriching a GOld mine. Nucleic Acids Research 49, D325–D334.

Denison, D.G.T., Mallick, B.K., Smith, A.F.M., 1998. A Bayesian CART algorithm. Biometrika 85, 363–377.

Durán, J.M., 2021. Dissecting scientific explanation in ai (sxai): A case for medicine and healthcare. Artificial Intelligence 297, 103498.

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., et al., 2021. Prottrans: Toward understanding the language of life through self-supervised learning. IEEE transactions on pattern analysis and machine intelligence 44, 7112–7127.

Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., Giannotti, F., 2018. Local rule-based explanations of black box decision systems. arXiv preprint arXiv:1805.10820 .

Han, J., Liu, S., Rose, D.M., Schlaepfer, D.D., McDonald, H., Ginsberg, M.H., 2001. Phosphorylation of the integrin alpha-4 cytoplasmic domain regulates paxillin binding. Journal of Biological Chemistry 276, 40903–40909.

Hayes, M.J., Rescher, U., Gerke, V., Moss, S.E., 2004. Annexin–actin interactions. Traffic 5, 571–576.

He, S., Liu, K., Ji, G., Zhao, J., 2015. Learning to Represent Knowledge Graphs with Gaussian Embedding, in: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, Association for Computing Machinery, New York, USA. URL: https://doi.org/10.1145/2806416.2806502, doi:10.1145/2806416.2806502.

Hoehndorf, R., Schofield, P.N., Gkoutos, G.V., 2011. PhenomeNET: a whole-phenome approach to disease gene discovery. Nucleic Acids Research 39, e119.

Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G.D., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., Ngomo, A.C.N., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., Zimmermann, A., 2021. Knowledge graphs. ACM Computing Surveys 54, 1–37.

Holzinger, A., Biemann, C., Pattichis, C.S., Kell, D.B., 2017. What do we need to build explainable ai systems for the medical domain? arXiv preprint arXiv:1712.09923 .

Huntley, R.P., Sawford, T., Mutowo-Meullenet, P., Shypitsyna, A., Bonilla, C., Martin, M.J., O'donovan, C., 2014. The GOA database: gene ontology annotation updates for 2015. Nucleic Acids Research 43, D1057–D1063.

Ieremie, I., Ewing, R.M., Niranjan, M., 2022. TransformerGO: predicting protein–protein interactions by modelling the attention between sets of gene ontology terms. Bioinformatics 38, 2269–2277.

Jain, S., Bader, G.D., 2010. An improved method for scoring protein-protein interactions using semantic similarity within the gene ontology. BMC Bioinformatics 11, 562.

Kastrin, A., Ferk, P., Leskošek, B., 2018. Predicting potential drug-drug interactions on topological and semantic similarity features using statistical learning. PLOS ONE 13, 1–23.

Koza, J.R., 1992. Genetic Programming: On the Programming of Computers by Means of Natural Selection. volume 1. MIT press, Cambridge.

Kulmanov, M., Liu-Wei, W., Yan, Y., Hoehndorf, R., 2019. EL embeddings: geometric construction of models for the description logic EL++. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence .

Kulmanov, M., Smaili, F.Z., Gao, X., Hoehndorf, R., 2020. Semantic similarity and machine learning with ontologies. Briefings in bioinformatics , bbaa199.

Lee, G., Park, C., Ahn, J., 2019. Novel deep learning model for more accurate prediction of drug-drug interaction effects. BMC Bioinformatics 20, 1–8.

Lin, X., Quan, Z., Wang, Z.J., Ma, T., Zeng, X., 2020. Kgnn: Knowledge graph neural network for drug-drug interaction prediction., in: IJCAI, pp. 2739–2745.

Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X., 2015. Learning entity and relation embeddings for knowledge graph completion, in: Proc. of the 29th AAAI Conf. on Artificial Intelligence, p. 2181–2187.

Liu, S., Kiosses, W.B., Rose, D.M., Slepak, M., Salgia, R., Griffin, J.D., Turner, C.E., Schwartz, M.A., Ginsberg, M.H., 2002. A fragment of paxillin binds the alpha-4 integrin cytoplasmic domain (tail) and selectively inhibits alpha-4-mediated cell migration. Journal of Biological Chemistry 277, 20887–20894.

Maetschke, S.R., Simonsen, M., Davis, M.J., Ragan, M.A., 2011. Gene Ontology-driven inference of protein–protein interactions using inducers. Bioinformatics 28, 69–75.

Mei, Y., Chen, Q., Lensen, A., Xue, B., Zhang, M., 2022. Explainable artificial intelligence by genetic programming: A survey. IEEE Transactions on Evolutionary Computation .

Miller, G.A., 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological review 63, 81.

Mjolsness, E., DeCoste, D., 2001. Machine learning for science: state of the art and future prospects. science 293, 2051–2055.

Mukherjee, S., Cogan, J.D., Newman, J.H., Phillips, J.A., Hamid, R., Meiler, J., Capra, J.A., 2021. Identifying digenic disease genes via machine learning in the Undiagnosed Diseases Network. The American Journal of Human Genetics 108, 1946–1963.

Nickel, M., Rosasco, L., Poggio, T., 2016. Holographic Embeddings of Knowledge Graphs, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence, AAAI Press, Washington DC, USA.

Palmonari, M., Minervini, P., 2020. Knowledge graph embeddings and explainable ai. Knowledge Graphs for Explainable Artificial Intelligence: Foundations, Applications and Challenges 47, 49.

Pesquita, C., Faria, D., Bastos, H., Falcao, A., Couto, F., 2007. Evaluating GO-based semantic similarity measures, in: Proceedings of the 10th Annual Bio-Ontologies Meeting, Vienna, Austria. pp. 37–40.

Resnik, P., 1995. Using information content to evaluate semantic similarity in a taxonomy, in: Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. p. 448–453.

Ribeiro, M.T., Singh, S., Guestrin, C., 2016. "Why should i trust you?": explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, New York, NY, USA. p. 1135–1144.

Ristoski, P., Paulheim, H., 2016. RDF2Vec: RDF graph embeddings for data mining, in: Groth, P., Simperl, E., Gray, A., Sabou, M., Krötzsch, M., Lecue, F., Flöck, F., Gil, Y. (Eds.), The Semantic Web – International Semantic Web Conference 2016, Springer International Publishing, Cham. pp. 498–514.

Roscher, R., Bohn, B., Duarte, M.F., Garcke, J., 2020. Explainable machine learning for scientific insights and discoveries. Ieee Access 8, 42200–42216.

Rudin, C., 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence 1, 206–215.

Seco, N., Veale, T., Hayes, J., 2004. An intrinsic information content metric for semantic similarity in wordnet, in: Proceedings of the 16th European Conference on Artificial Intelligence, IOS Press, NLD. p. 1089–1090.

Sezaki, T., Tomiyama, L., Kimura, Y., Ueda, K., Kioka, N., 2013. Dlg5 interacts with the TGF-beta receptor and promotes its degradation. FEBS Letters 587, 1624–1629.

Smaili, F.Z., Gao, X., Hoehndorf, R., 2018. OPA2Vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. Bioinformatics 35, 2133–2140.

Sousa, R.T., Silva, S., Pesquita, C., 2020. Evolving knowledge graph similarity for supervised learning in complex biomedical domains. BMC Bioinformatics 21, 6.

Sousa, R.T., Silva, S., Pesquita, C., 2021. evoKGsim+: a framework for tailoring knowledge graph-based similarity for supervised learning, in: ESWC 2021 Poster and Demo Track.

Sousa, R.T., Silva, S., Pesquita, C., 2023. Explainable representations for relation prediction in knowledge graphs. `arXiv:2306.12687`.

Staab, S., Studer, R., 2010. Handbook on ontologies. Springer-Verlag.

Szklarczyk, D., Gable, A.L., Nastou, K.C., Lyon, D., Kirsch, R., Pyysalo, S., Doncheva, N.T., Legeay, M., Fang, T., Bork, P., Jensen, L.J., von Mering, C., 2020. The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. Nucleic Acids Research 49, D605–D612.

Traverso, I., Vidal, M.E., Kämpgen, B., Sure-Vetter, Y., 2016. GADES: A graph-based semantic similarity measure, in: Proceedings of the 12th International Conference on Semantic Systems, Association for Computing Machinery, New York, NY, USA. pp. 101–104.

Traverso-Ribón, I., Vidal, M.E., 2018. Garum: A semantic similarity measure based on machine learning and entity characteristics, in: Hartmann, S., Ma, H., Hameurlain, A., Pernul, G., Wagner, R.R. (Eds.), Database and Expert Systems Applications - Volume 11029, Springer International Publishing, Cham. pp. 169–183.

Trouillon, T., Welbl, J., Riedel, S., Gaussier, E., Bouchard, G., 2016. Complex embeddings for simple link prediction, in: Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, JMLR.org, New York, NY, USA. p. 2071–2080.

Wang, D., Yang, Q., Abdul, A., Lim, B.Y., 2019a. Designing theory-driven user-centric explainable ai, in: Proceedings of the 2019 CHI conference on human factors in computing systems, pp. 1–15.

Wang, Q., Mao, Z., Wang, B., Guo, L., 2017. Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering 29, 2724–2743.

Wang, Y., Yang, S., Zhao, J., Du, W., Liang, Y., Wang, C., Zhou, F., Tian, Y., Ma, Q., 2019b. Using machine learning to measure relatedness between genes: a multi-features model. Scientific Reports 9, 4192.

Wang, Z., Zhang, J., Feng, J., Chen, Z., 2014. Knowledge graph embedding by translating on hyperplanes, in: Proceedings of the 28th AAAI Conference on Artificial Intelligence, AAAI Press, Québec City, Québec, Canada. p. 1112–1119.

Xiong, B., Potyka, N., Tran, T.K., Nayyeri, M., Staab, S., 2022. Faithful Embeddings for EL++ Knowledge Bases, in: International Semantic Web Conference, Springer. pp. 22–38.

Yang, B., tau Yih, W., He, X., Gao, J., Deng, L., 2015. Embedding entities and relations for learning and inference in knowledge bases. `arXiv:1412.6575`.

Zhang, N., Bi, Z., Liang, X., Cheng, S., Hong, H., Deng, S., Zhang, Q., Lian, J., Chen, H., 2021a. Ontoprotein: Protein pretraining with gene ontology embedding, in: International Conference on Learning Representations.

Zhang, S.B., Tang, Q.R., 2016. Protein–protein interaction inference based on semantic similarity of gene ontology terms. Journal of Theoretical Biology 401, 30–37.

Zhang, Y.H., Zeng, T., Chen, L., Huang, T., Cai, Y.D., 2021b. Determining protein–protein functional associations by functional rules based on gene ontology and KEGG pathway. Biochimica et Biophysica Acta (BBA)-Proteins and Proteomics 1869, 140621.

Zhong, X., Kaalia, R., Rajapakse, J.C., 2019. GO2Vec: transforming GO terms and proteins to vector representations via graph embeddings. BMC Genomics 20, 1–10.

Zhong, X., Rajapakse, J.C., 2020. Graph embeddings on gene ontology annotations for protein–protein interaction prediction. BMC bioinformatics 21, 1–17.

Appendix B

Explainable representations for relation prediction in knowledge graphs

Explainable representations for relation prediction in knowledge graphs

Rita T. Sousa[1] , **Sara Silva**[1] , **Catia Pesquita**[1]
[1]LASIGE, Faculdade de Ciências da Universidade de Lisboa
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.com

Abstract

Knowledge graphs represent real-world entities and their relations in a semantically-rich structure supported by ontologies. Exploring this data with machine learning methods often relies on knowledge graph embeddings, which produce latent representations of entities that preserve structural and local graph neighbourhood properties, but sacrifice explainability. However, in tasks such as link or relation prediction, understanding which specific features better explain a relation is crucial to support complex or critical applications.

We propose SEEK, a novel approach for explainable representations to support relation prediction in knowledge graphs. It is based on identifying relevant shared semantic aspects (i.e., subgraphs) between entities and learning representations for each subgraph, producing a multi-faceted and explainable representation.

We evaluate SEEK on two real-world highly complex relation prediction tasks: protein-protein interaction prediction and gene-disease association prediction. Our extensive analysis using established benchmarks demonstrates that SEEK achieves significantly better performance than standard learning representation methods while identifying both sufficient and necessary explanations based on shared semantic aspects.

1 Introduction

Knowledge Graphs (KGs) (Ehrlinger and Wöß 2016) are a representation of factual information about entities in the real world and how they relate to each other, having been widely used to support various applications including machine learning (ML) (Hogan et al. 2021). Particularly, in scientific domains, KGs have become highly relevant because they allow for the description and linking of information about entities based on ontologies (Staab and Studer 2010), allowing the description of complex natural phenomena that are not easily captured in mathematical form (Nicholson and Greene 2020).

In recent years, KG embedding methods (Wang et al. 2017) have become increasingly popular to bridge the gap between the complex representations a KG affords and the vectorial representations most ML methods take as input, since they map KGs into low-dimensional spaces preserving syntactic and structural properties. KG embeddings are popularly employed in link prediction via a scoring function or as features for supervised learning (Portisch, Heist, and Paulheim 2022). However, this represents a significant trade-off: KG embeddings sacrifice the full and rich interpretability offered by KGs, especially when structured by rich ontologies, for the more simple to process latent representations (Palmonari and Minervini 2020). The effectiveness and usefulness of KG embeddings approach hinges on the crucial assumption that KG embeddings serve as semantically meaningful representations of the underlying entities. To validate such an assumption, KG embedding methods would need to be explainable (i.e., they would need to afford a human-understandable description of the logic, behavior or factors that influence the representation learning process), but in the vast majority of cases they are not. This is a fundamental requirement to ensure the scientific validity of KG embeddings, or any artificial intelligence (AI) method, as a tool that can be used to uncover new knowledge, help understand the mechanisms underlying natural phenomena, and distinguish meaningful predictions from spurious correlations (Barredo Arrieta et al. 2020).

In this work, we focus specifically on the problem of predicting a relation between KG entities that is not defined in the KG. Predicting relations such as protein-protein interactions (PPI) or gene-disease associations (GDA) by exploring KGs and ontologies has been the focus of extensive research in the biomedical domain. Both algorithmic (Zhang and Tang 2016; Hoehndorf, Schofield, and Gkoutos 2011; Asif et al. 2018) and ML approaches (Kulmanov et al. 2021) have been employed to achieve this with success, with KG embeddings particularly excelling at the task (Chen, Wang, and Hu 2019; Ieremie, Ewing, and Niranjan 2022; Alshahrani et al. 2017). However, understanding the nature of these relations requires discerning which aspects of the KG have the most influence on a prediction. This empowers users not only in assessing the reliability of the model itself but also in potentially elucidating the phenomena underlying the relation. For example, if we were to explain the interaction between the proteins *Protransforming growth factor* α and *Disks large homolog 2*, generating an explanation based on the fact that they both perform the very specific function *MAPK cascade*, we would likely increase trust as well as highlight a relevant aspect for interaction. In contrast, a very general explanation, such as the fact that both proteins are present in the *plasma membrane* would contribute to neither.

We propose SEEK (Shared Explainable Embeddings for

Knowledge graphs), a novel method for generating explainable KG embeddings that represent entity pairs for relation prediction. The intuition behind this is that an entity pair can be represented by combining embeddings that represent each of their shared semantic aspects, rather than simply combining their respective embeddings. This technique explores the rich semantics of the ontology to identify the shared semantic aspects between related entities based on computing their disjoint common ancestors. Then, these pair embeddings are used to train a supervised ML model for relation prediction. SEEK is fundamentally different from link prediction methods since it produces representations of pairs of entities based on shared semantic aspects, whereas link prediction methods rely on learning representations of individual KG entities and apply a scoring function to estimate the likelihood of triples.

Given a prediction, our method explains it by computing the importance of each shared semantic aspect in identifying it. Inspired by (Watson et al. 2021; Rossi et al. 2022a) we consider that an explanation includes two complementary views: the set of semantic aspects that, if absent from an entity pair, would render the model incapable of generating that prediction (i.e., necessary explanations); the set of semantic aspects that, if shared by any entity pair, would prompt the model to produce that prediction (sufficient explanations). Since SEEK explains specific predictions rather than the global mechanism of the model, it consequently falls under the category of local post-hoc explanation methods as proposed by (Guidotti et al. 2018).

We evaluate the effectiveness of SEEK in two different tasks, PPI prediction and GDA prediction. Predicting PPI is a crucial task in molecular biology (Li et al. 2021; Hu et al. 2021), and several KG embedding-based methods have been employed to tackle it (Kulmanov et al. 2019; Smaili, Gao, and Hoehndorf 2019; Kulmanov et al. 2021; Kulmanov et al. 2019; Xiong et al. 2022). Due to the high costs and challenges involved in experimentally determining PPI, computational methods can be used to identify protein pairs that are likely to interact, which are subsequently validated through experimental assays rendering the process more efficient. Likewise, predicting the relation between genes and diseases is essential to understand disease mechanisms and identifying potential biomarkers or therapeutic targets (Eilbeck, Quinlan, and Yandell 2017). Once again, computational approaches to identify the most promising associations to be further validated are commonly employed, with recent approaches applying KG embedding methods (Alshahrani et al. 2017; Smaili, Gao, and Hoehndorf 2019; Nunes, Sousa, and Pesquita 2021). However, opaque methods such as KG embeddings are unable to provide explanations behind each prediction. Explanatory mechanisms can elucidate the potential mechanisms behind the predicted relation, which can be helpful to determine the type of experimental procedure that should be applied to confirm the precited relation but also to identify data biases that can result in misclassification and should be grounds to discard the candidate pair. Our extensive experiments show that our method produces useful explanations besides improving performance over state-of-the-art embedding methods.

Our main contributions are the following:

- We propose SEEK, a novel method for generating explainable KG embeddings that represent entity pairs for relation prediction.
- We develop extensions of popular KG embedding methods implementing SEEK.
- We design explanation methods that quantify the importance of specific KG semantic aspects for specific relation predictions.
- We report extensive experimental results demonstrating that SEEK is able to produce effective explanations for relation prediction as well as generally improving predictive performance on multiple models and biomedical datasets.

2 Problem overview

We define a KG as a labeled directed graph $KG = (V, E, R)$ where V is the set of vertices that represent entities, R is the set of relations and E is the set of edges that connect vertices through relations. Our particular focus is on ontology-rich KGs with ontologies defined using Web Ontology Language (OWL) (Grau et al. 2008) since biomedical ontologies are typically developed in OWL or have an OWL version. These are frequently found in scientific fields like biomedicine. In these KGs, ontologies are typically used to describe individual instances, while the instances themselves are usually flat with no connections between them. Consequently, there will be two types of vertices in the KG: those that correspond to individual entities and those that correspond to ontology classes, as well as two types of edges: those that relate ontology classes to each other, and those that link individuals to the classes that describe them. For example, using OWL 2 whose constructs correspond to SROIQ(D), we can indicate that a protein P carries out a function F described in the Gene Ontology (GO) by declaring the axiom $P \sqsubseteq \exists hasFunction.F$. KG embedding methods are then able to learn representations of biomedical entities by exploring the links that connect an entity to the ontology classes that describe it, as well as the structure of the ontology itself.

Our objective is to learn a relation between two KG entities, a pair, when the relation itself is not explicitly defined in the KG, using embeddings as inputs for a supervised ML algorithm. This is a fundamentally distinct task from link prediction, where the training set relations are part of the KG. To tackle this relation prediction task, common approaches typically employ three steps: (1) generate embeddings for each entity in the KG; (2) aggregate the embeddings of each entity in a pair into a single representation; (3) use these aggregated representations as input to a supervised learning algorithm to learn a relation prediction model (Sousa, Silva, and Pesquita 2021; Celebi et al. 2019). This generates non-explainable predictions since KG embeddings are, of course, non-explainable, as each dimension does not represent any particular meaning, which poses a serious limitation to the use of KG embeddings in a scientific setting.

Moreover, this particular formulation results in two oversimplifications, which may limit its effectiveness and usefulness. Firstly, it relies on the aggregation of individual embeddings to represent a pair of entities, instead of directly learning an embedding that represents the pair. One should clarify that simply representing the pair as yet another entity on the KG would not be a viable solution, as it would limit the applicability of the approach to pairs seen at representation learning time and thus fail to generalize to novel pairs. Secondly, it focuses on creating an overall representation of each entity, rather than capturing the different semantic aspects that may contribute to the relation we aim to predict. In large KGs, it is not uncommon for entities to be described according to multiple semantic aspects, but only a few may be relevant for the prediction of a particular relation. In a previous study (Sousa, Silva, and Pesquita 2020), it was demonstrated that not all branches of the GO are equally important for predicting PPIs.

The problem we tackle is then two-fold: (1) to generate latent representations that represent an entity pair directly and (2) to generate latent representations that are amenable to explanation and can capture the relevant semantic aspects for relation prediction.

3 Related work

3.1 Knowledge graph embeddings

KG embedding methods represent KG entities and their relations in a lower-dimensional space preserving the KG semantic information as much as possible. These embeddings have been employed as features in a variety of downstream tasks, such as link prediction, triple classification, or entity typing. KG embeddings have been successfully employed in a number of scientific applications, with particular success in the life sciences (Mohamed, Nounu, and Nováček 2021; Kulmanov et al. 2021; Chen, Wang, and Hu 2019; Ieremie, Ewing, and Niranjan 2022). There are several types of KG embeddings, including translational models, semantic matching models, or random walk-based KG embedding approaches.

Translational methods use distance-based scoring functions. TransE (Bordes et al. 2013) is a well-known approach that assumes the vector of the head entity plus the relation vector should be close to the vector of the tail entity if a relation holds between two entities. However, TransE only handles one-to-one relationships. To overcome this limitation, TransH (Wang et al. 2014) introduces a unique relation-specific hyperplane for each relationship.

Semantic matching models rely on scoring functions based on similarity, which can represent the underlying meaning of entities and relationships in vector spaces. An example of such a method is DistMult (Yang et al. 2015), which uses tensor factorization to create vector embeddings for entities and diagonal matrices for relationships.

Random walk-based embedding techniques perform walks in the graph to produce a corpus of sequences that is given as input to a neural language model (Mikolov et al. 2013) to learn a latent low-dimensional representation of each entity within the corpus of sequences. RDF2Vec (Ristoski and Paulheim 2016) is used to learn embeddings over RDF graphs.

More recently, KG embedding approaches that tailor representations by considering specific aspects of a KG have been proposed. EL (Kulmanov et al. 2019) and BoxEL (Xiong et al. 2022) embeddings are geometric approaches that consider the logical structure of the ontology. OWL2Vec* (Chen et al. 2021) is very similar to RDF2Vec, but it was designed to learn embeddings of OWL ontologies, which are used to represent knowledge in a more expressive and formal way than RDF graphs. OPA2Vec (Smaili, Gao, and Hoehndorf 2019) considers the lexical portion of the KG, specifically the labels of entities, when generating triples.

3.2 Explainable artificial intelligence techniques

The scientific community has long recognized the potential of AI as a tool for scientific discovery, with ML, pattern mining, and reasoning playing crucial roles in several steps of the scientific process (Mjolsness and DeCoste 2001). Despite this, the vast majority of scientific projects that utilize AI do not prioritize explainability (Roscher et al. 2020). In the biomedical domain, the complexity of both the data and the natural phenomena under study emphasizes the importance of domain knowledge to support explainability (Holzinger et al. 2017). A knowledge-enabled explainable AI (XAI) system includes a representation of the domain knowledge specific to the application field. This knowledge is explored for generating explanations that are both comprehensible to users and contextually aware of the mechanistic functioning of the AI system and the knowledge it employs (Chari et al. 2020).

XAI aims to address several key objectives, including promoting algorithmic fairness, detecting potential biases or issues in training data, ensuring that algorithms function as intended, and bridging the gap between the ML community and other scientific disciplines (Gilpin et al. 2018). According to (Barredo Arrieta et al. 2020), XAI approaches can be classified into two types: models that are transparent by design, such as decision trees, linear models, and genetic programming models (Mei et al. 2022), or post-hoc explainability techniques that are used to improve the interpretability of models that are not transparent by design. Post-hoc explainability techniques can be categorized as either model-specific or model-agnostic if they are applicable to any ML model. Post-hoc techniques may include visual explanations, explanations by example, explanations by simplification, or feature relevance explanations.

KG embeddings are not explainable, and there is no widely accepted methodology to effectively explain the predictions of KG embeddings (Palmonari and Minervini 2020). CRIAGE (Pezeshkpour, Tian, and Singh 2019) and Kelpie (Rossi et al. 2022b) have made striding efforts towards explaining link prediction based on KG embeddings by identifying the fact to add into or remove from the KG that affects the prediction for a target fact. Betz *et al.* (Betz, Meilicke, and Stuckenschmidt 2022) also propose a post hoc method that uses adversarial attacks on KG embedding models to identify triples that serve as logical explanations for

specific predictions. These works differ fundamentally from ours by focusing on single facts about each entity, whereas we focus on shared aspects between entity pairs. Additionally, all of these works face the computational challenge posed by having to retrain the KG model after removing a single fact to explain each prediction, and devise heuristic approaches to minimize this aspect. Our approach does not require retraining the model. Instead, we generate explanations by identifying shared semantic aspects and making predictions with the trained model. ExCut (Gad-Elrab et al. 2020) is another approach that uses KG embeddings to identify clusters of entities and then combines it with rule-mining methods to learn interpretable labels.

4 Methods

4.1 Overview

SEEK is a novel approach that generates explainable vector representations of KG entity pairs to support relation prediction tasks with minimal loss in performance.

Figure 1 shows an overview of the SEEK approach. In the first step, the KG is transformed into an RDF graph, which facilitates the subsequent processing. Representations for each ontology class are then learned using a KG embedding method. Notably, SEEK is agnostic to the specific KG embedding method employed and can accommodate a broad range of techniques.

The second step is concerned with identifying the shared semantic aspects between the entities of the pair, which are determined by computing the disjoint common ancestors of all classes related to them. The identification of these semantic aspects is essential for the subsequent generation of accurate and meaningful explanations. Having identified the relevant semantic aspects, the final representations of entity pairs are then generated by aggregating the embeddings of the shared semantic aspects.

In the third and final step, supervised learning methods are employed to learn a relation prediction model taking as input the pair embeddings. This model is then used to generate explanations by adopting a perturbation-inspired approach where the contribution of each semantic aspect to the final prediction is assessed in terms of its sufficiency and necessity. The necessary explanations provide insights into the semantic aspects that are necessary for a particular decision to be made, while the sufficient explanations reveal the aspects that are sufficient to support a particular decision. These explanations enable a more thorough understanding of predicted relations and which KG aspects influence it and can be invaluable in identifying potential biases or errors.

4.2 Generating the RDF graph and learning embeddings

Ontology-rich KGs are typically defined in OWL. However, the majority of graph processing and analysis tools require RDF graphs. Therefore, the initial step is to convert the KG into an RDF graph following the guidelines provided by the W3C[1]. The conversion process involves transforming simple axioms directly into RDF triples, such as subsumption

[1]https://www.w3.org/TR/owl2-mapping-to-rdf/

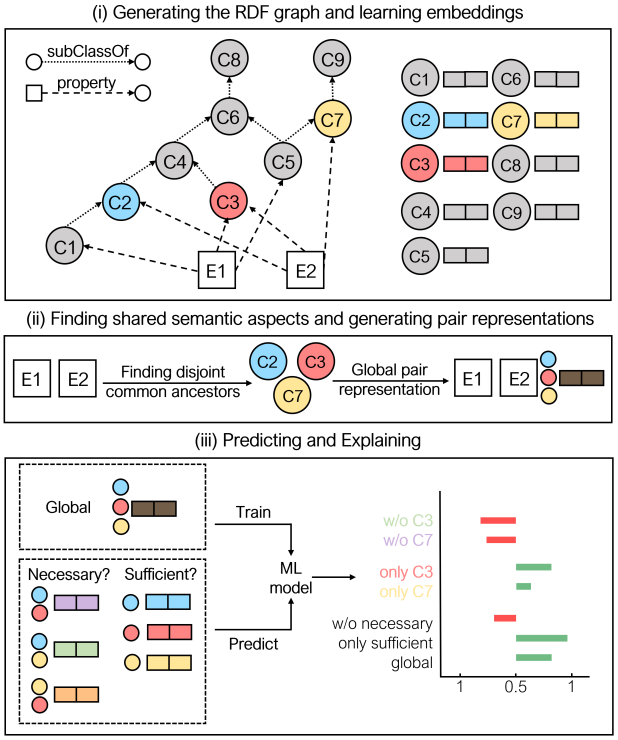


Figure 1: Overview of SEEK with the main steps: (i) generating the RDF graph and learning embeddings (ii) finding shared semantic aspects and generating pair representations (iii) predicting and explaining.

axioms or data and annotation properties associated with an entity. Multiple triples are created for more complex axioms involving class expressions, which usually require blank nodes. The relations between entities and the ontology classes describing them are usually stored in annotation files in the biomedical domain. These annotations are processed into object properties. After conversion, the nodes in the RDF graph represent ontology classes or individuals, and the edges represent named relations. Finally, we employ a KG embedding method to learn latent representations of all the ontology classes in the KG.

4.3 Finding shared semantic aspects and generating pair representations

To generate a representation for an entity pair we explore the concept of semantic aspect (i.e., a subgraph of the KG that captures a specific perspective of the domain). We propose to represent a pair of KG entities by the set of semantic aspects they share, unfolding their relationship into different dimensions each based on a shared aspect. We define the shared semantic aspects as the set of disjoint common ancestors computed over the set of classes that describe each entity.

Let us take two entities e_1 and e_2 and their set of linked classes C_1, C_2. To compute the set of disjoint shared as-

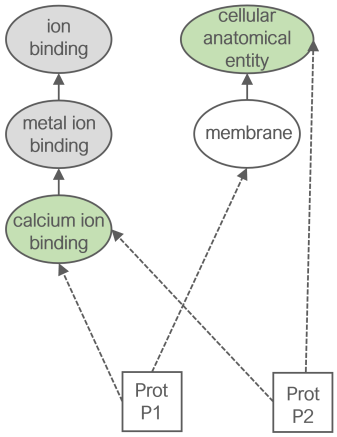


Figure 2: A GO KG subgraph to represent the shared semantic aspects of two entities. Green classes represent the disjoint common ancestors of proteins P1 and P2. Grey classes represent the remaining common ancestors.

pects, we first compute the disjoint common ancestors of C_1 and C_2. Following (Couto and Silva 2011), we define that a_1 and a_2 are disjoint common ancestors of a class c if $c \sqsubseteq a_1$, $c \sqsubseteq a_2$, $a_1 \not\sqsubseteq a_2$ and $a_2 \not\sqsubseteq a_1$. We first compute C_a, the set of common ancestors between the two sets C_1 and C_2, and then filter this set to include only the disjoint common ancestors, each of which represents a shared semantic aspect. The shared semantic aspects of two sets only include indirect common ancestors if they do not subsume other common ancestors. Considering the example in Figure 2, the shared semantic aspects of proteins P1 and P2 correspond to *calcium ion binding* and *cellular anatomical entity*.

To represent an entity pair we take the embeddings of each class in the shared semantic aspects set and aggregate them using simple operators such as the Hadamard product, the sum, the average or the L1-norm.

4.4 Predicting and explaining

After obtaining the vector representations, we use supervised ML algorithms to learn relation prediction models and ultimately produce explanations for predicted relations. First, we train our model using the global representation of the pair, generated by aggregating all shared semantic aspect embeddings. Then, for each prediction we want to explain, we generate multiple representations that differ by the presence or absence of a semantic aspect. To understand which semantic aspects are necessary for the prediction, we generate representations that remove each aspect in turn (see Algorithm 1), whereas to understand which aspects are sufficient for the prediction, we generate representations that include a single aspect (see Algorithm 2). A semantic aspect is considered necessary for a prediction if the predicted class changes when it is removed. Likewise, a semantic aspect is considered sufficient for a prediction if the predicted class does not change when it is the only aspect considered.

We define an explanation as the set of the most relevant shared semantic aspects identified as necessary or sufficient. A necessary explanation is a shared semantic aspect that, when removed from the pair representation, causes the classifier to change its prediction. A sufficient explanation is a shared semantic aspect that, when used alone to represent a pair, causes the classifier to maintain its prediction. A relation may be explained by multiple necessary and sufficient explanations.

This approach is similar to how saliency XAI methods inject perturbations in the feature space to capture the importance of features. However, it addresses a significant challenge that perturbation or modification-based methods face, including those that aim to explain KG embeddings (Pezeshkpour, Tian, and Singh 2019; Rossi et al. 2022b), which is the need to relearn representations after performing the modification to the data. SEEK avoids this hurdle since it is based on composite representations of ontology classes, which are easy to modify and do not require retraining since the ontology itself is never altered, so the learned class embeddings remain fixed.

The final explanation can be represented as a chart where sufficient and necessary shared semantic aspects are presented alongside their impact on the prediction. In Figure 1, both C3 and C7 are necessary to support the prediction since, without either of them, the prediction value changes when compared to the prediction obtained for the global representation. C3 is also a sufficient aspect since it can single-handedly produce a prediction that agrees with the global one. The explanation can be further enriched with the prediction of the global approach, a prediction made with all sufficient shared semantic aspects, and a prediction made without any of the necessary shared semantic aspects, all predictions including their respective likelihood.

Algorithm 1 Generation of necessary explanations

Input:the entity pair (e_1, e_2);
the KG embedding model K;
the relation prediction model M;
Output: the set of disjoint shared aspects that are necessary for explaining the prediction

$N \leftarrow empty$
$D \leftarrow$ GET DISJOINT SHARED ASPECTS$((e_1, e_2))$
$E \leftarrow$ GET EMBEDDINGS(K, D)
$v \leftarrow$ AGGREGATE(E)
$p \leftarrow$ PREDICT(M, v)
for $d \in D$ **do**
  $e' \leftarrow E.delete(d)$
  $v' \leftarrow$ AGGREGATE(e')
  $p' \leftarrow$ PREDICT(M, v')
  if $p \neq p'$ **then**
    $N.append(d)$
return N

Algorithm 2 Generation of sufficient explanations

Input:the entity pair (e_1, e_2);
the KG embedding model K;
the relation prediction model M;
Output: the set of disjoint shared aspects that are sufficient for explaining the prediction

$S \leftarrow empty$
$D \leftarrow$ GET DISJOINT SHARED ASPECTS$((e_1, e_2))$
$E \leftarrow$ GET EMBEDDINGS(K, d)
$v \leftarrow$ AGGREGATE(E)
$p \leftarrow$ PREDICT(M, v)
for $d \in D$ **do**
  $v' \leftarrow$ GET EMBEDDING(K, d)
  $p' \leftarrow$ PREDICT(M, v')
  if $p == p'$ **then**
    $S.append(d)$
return S

5 Experimental Results

5.1 Experimental setup

We evaluate SEEK on two biomedical relation prediction tasks: predicting PPIs and predicting GDA. Both tasks are grounded on ontology-rich KGs, where PPI employs the GO and GDA is based on the Human Phenotype Ontology (HP). Additionally, prior studies have shown that different branches of these ontologies have varying impacts on achieving precise predictions (Sousa, Silva, and Pesquita 2020).

Our work targets relation prediction tasks cast as a classification task that takes as input entity pairs and a KG backboned by an ontology. Ontologies are arranged in a directed acyclic graph, where ontology classes are connected by subclass relations such that each class is more specific than its ancestor. Moreover, these relationships are transitive, meaning they inherit all ancestors to the root. The data used are described in the following sections.

Table 1: Statistics for each task regarding classes, nodes, and edges. Positive and negative pairs correspond to the number of positive and negative relations.

	PPI	**GDA**
Ontology classes	50422	15656
Literals and blank nodes	462874	443489
Instances	6738	4523
Annotations	349500	160009
Positive Pairs	23571	8189
Negative Pairs	23571	8189

Protein-protein interaction prediction The target relations to predict are obtained from the STRING database (Szklarczyk et al. 2020), one of the largest PPI databases that integrate physical interactions and functional associations between proteins from various sources. We filtered the protein pairs to include only pairs that met the following criteria: (i) each protein must be annotated with the GO, (ii) interactions must be extracted from curated databases or experimentally determined, and (iii) interactions must have a confidence score above 0.950. The PPI dataset contains 23571 interacting protein pairs as well as 23571 negative pairs derived from random negative sampling of the same set of proteins.

The GO KG is used to describe proteins and is built by integrating the GO (Consortium 2021) and protein annotation data (Huntley et al. 2014). The GO defines a hierarchy of classes that describe protein functions and their relationships. It can be represented as a graph where nodes are GO classes and edges define relationships between them (e.g., is_a; $part\_of$; has_part; $regulates$; $negatively\_regulates$ and $positively\_regulates$), being the majority is_a relations. The three domains of GO (biological processes, molecular functions, and cellular components) are represented as separate root ontology classes since they do not share any common ancestor. A GO annotation is a statement about the function F of a protein P, and it is added in the KG as an assertion $\langle P, hasFunction, F\rangle$. In GO KG, nodes represent proteins or GO classes, and edges represent links between GO classes or annotations. Table 1 describes the statistics about PPI data.

Gene-disease association Prediction The target relations to predict are obtained from DisGeNET (Piñero et al. 2019). We follow the approach in (Nunes, Sousa, and Pesquita 2021), which excludes associations whose sources are used to create some of the ontology annotations. Moreover, each gene and disease must have at least one HP annotation. This resulted in a balanced dataset with a total of 16378 gene-disease pairs.

In this experiment, we employ the HP KG comprising the HP (Köhler et al. 2020) and HP annotation data to describe genes and diseases. HP characterizes the phenotypic abnormalities in human hereditary diseases concerning five semantic aspects, namely phenotypic abnormalities, mode of inheritance, clinical course, clinical modifier and frequency. Regarding the HP annotations, they link genes and diseases to HP classes and are added in the KG in the same fashion as in the PPI experiment. The statistics about GDA data are also shown in Table 1.

Models SEEK is independent of the KG embedding method and of the supervised ML algorithm. For our experiments, we implemented five representative KG embeddings covering translational, semantic matching and random walk-based methods: RDF2Vec (Ristoski and Paulheim 2016), OWL2Vec* (Chen et al. 2021), TransE (Bordes et al. 2013), TransH (Wang et al. 2014) and distMult (Yang et al. 2015). RDF2Vec and OWL2Vec* are path-based approaches adapted to RDF graphs that employ neural language models over random walks on the graph. TransE and TransH are translational distance embedding approaches that exploit distance-based scoring functions. distMult is a semantic matching approach that exploits similarity-based scoring functions.

To generate a pair representation, we use the average as the aggregation which ensures that the values of each dimension remain within the distribution. In the case of necessary

explanations, removing one similar semantic aspect will result in a very similar aggregated representation, revealing that the semantic brings little novel information for the prediction (since a similar semantic aspect is still considered). In the case of sufficient explanations, semantic aspects are evaluated independently.

As supervised ML algorithms, we employ two ensemble methods, Random Forest (RF) (Breiman 2001) and eXtreme Gradient Boosting (XGB) (Chen and Guestrin 2016), and a neural network-based method, Multilayer Perceptron (MLP) (Rumelhart, Hinton, and Williams 1986).

5.2 Results and Discussion

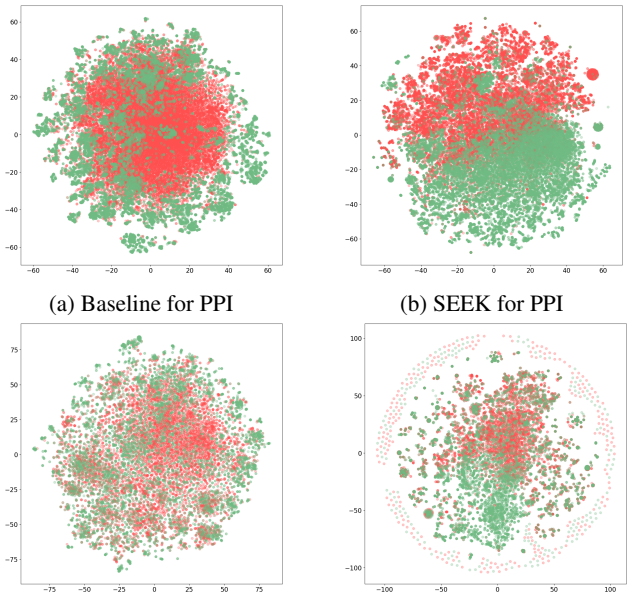
Performance evaluation To assess our method, we compare the relation prediction performance of our pair representations against the state-of-the-art approach of entity vector aggregation using representative KG embedding methods, supervised ML algorithms and the Hadamard operator. We do not compare SEEK to other KG embedding explanation methods such as Kelpie or CRIAGE because they learn embeddings that target link prediction, whereas SEEK learns embeddings to serve as features for supervised ML. We evaluate the predictive performance of our approach against our baselines for each task using 10-fold cross-validation. For each partition, the precision (Pr), recall (Re) and weighted average f1-score (F1) are computed, and we report the median of the obtained scores (Table 2) and statistical significance of the observed differences.

The results demonstrate that SEEK outperforms the baseline in all cases but one for PPI prediction, while achieving similar or improved scores for GDA. Curiously, the performance of translational methods shows a marked improvement when using SEEK, likely due to the fact that these methods struggle with learning entity representations, but not ontology class representations.

To better understand the differences between the pair representations obtained using the baselines and the ones obtained using SEEK, we plot the RDF2Vec embeddings using t-SNE (Van der Maaten and Hinton 2008), a nonlinear dimensionality reduction technique that is particularly well-suited for visualizing high-dimensional data (Figure 3). These plots show that our pair representations decrease the overlap between positive and negative pairs and thus are likely to be capturing more meaningful representations.

Effectiveness of explanations The effectiveness of the explanations is measured based on how predictive performance varies under two scenarios: when pairs are represented without the *necessary* shared semantic aspects; when pairs are represented by *sufficient* shared semantic aspects only. Table 3 presents the results obtained for the PPI and GDA tasks using the two best performing KG embedding methods.

In the necessary scenario, we extract the necessary explanations for all correctly predicted relations and produce an ablated representation that does not include any of the necessary shared semantic aspects. The performance variation, in terms of precision (Pr), recall (Re) and F1-score (F1), is measured as the difference in predictive performance between the global representation and the ablated representation. The more negative Δ Pr, Δ Re or Δ F1 are, the more effective are the necessary explanations.



(a) Baseline for PPI (b) SEEK for PPI

(c) Baseline for GDA (d) SEEK for GDA

Figure 3: t-SNE plots comparing SEEK to the baseline using RDF2Vec. Positive pairs in green and negative pairs in red.

In the sufficient scenario, we extract the sufficient explanations for all incorrectly predicted relations and produce an ablated representation that only includes the sufficient shared semantic aspects. The performance variation is also measured as the difference in predictive performance between the global representation and the ablated representation, but in this case the performance of the global representation is actually zero for all scores, since this is only applied to incorrectly predicted relations. A higher Δ value indicates increased effectiveness.

Explanation length The lengths of the explanations, as measured by the number of shared semantic aspects that compose them, are presented in Tables 4 and 5. In both tasks, the length of necessary explanations is markedly lower than the length of sufficient explanations, highlighting that for many relations there are no necessary shared semantic aspects. When comparing the shown results to the original number of shared semantic aspects, 9.1 ($\pm$ 6.5) for PPI and 8.5 ($\pm$ 11.0) for GDA, we can verify that sufficient explanations amount to roughly 30% of shared semantic aspects. These sizes are congruent with the number of objects (7 ± 2) humans are able to hold in short-term memory according to cognitive studies (Miller 1956).

Examples of explanations Table 6 presents explanations for four protein pairs chosen randomly from the PPI dataset. Each pair represents each of the four possible outcomes: a true positive, a false positive, a true negative, and a false negative.

Table 2: Medians of precision, recall, and weighted average f1-score (Pr, Re, F1) comparing our approach SEEK to the baseline when coupled with different supervised ML methods for PPI and GDA prediction. SEEK performance values are underlined when improvements are statistically significant with p-value < 0.05 for the Wilcoxon test against the baselines.

		PPI Prediction						**GDA Prediction**					
		Baseline			SEEK			Baseline			SEEK		
		Pr	Re	F1	Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
RDF2Vec	XGB	0.905	0.917	0.910	0.920	0.910	0.915	0.736	0.708	0.724	0.772	0.626	0.719
	RF	0.921	0.881	0.902	0.922	0.892	0.910	0.783	0.673	0.740	0.787	0.625	0.723
	MLP	0.897	0.907	0.902	0.908	0.924	0.917	0.700	0.705	0.696	0.730	0.645	0.703
OWL2Vec*	XGB	0.890	0.881	0.888	0.933	0.925	0.929	0.700	0.664	0.688	0.780	0.647	0.728
	RF	0.913	0.832	0.875	0.922	0.915	0.919	0.730	0.618	0.690	0.780	0.662	0.737
	MLP	0.872	0.865	0.869	0.934	0.923	0.931	0.648	0.676	0.650	0.749	0.642	0.720
distMult	XGB	0.897	0.905	0.902	0.914	0.910	0.912	0.718	0.668	0.704	0.764	0.649	0.722
	RF	0.904	0.860	0.884	0.910	0.897	0.905	0.745	0.636	0.706	0.766	0.637	0.716
	MLP	0.894	0.894	0.896	0.881	0.895	0.888	0.731	0.681	0.715	0.768	0.589	0.698
TransE	XGB	0.642	0.613	0.638	0.914	0.912	0.913	0.526	0.509	0.524	0.755	0.650	0.721
	RF	0.590	0.542	0.583	0.908	0.900	0.905	0.505	0.474	0.502	0.765	0.640	0.719
	MLP	0.250	0.500	0.333	0.882	0.899	0.890	0.500	1.000	0.333	0.779	0.555	0.694
TransH	XGB	0.642	0.614	0.637	0.921	0.918	0.919	0.511	0.493	0.510	0.767	0.651	0.726
	RF	0.586	0.551	0.579	0.912	0.908	0.910	0.500	0.453	0.494	0.770	0.642	0.720
	MLP	0.250	0.500	0.333	0.915	0.920	0.920	0.000	0.000	0.333	0.735	0.665	0.711

Table 3: Explanation efectiveness measured based on the precision (Pr), recall (Re) and weighted average f1-score (F1) variation for PPI and GDA prediction.

		PPI Prediction						**GDA Prediction**					
		RDF2Vec			OWL2Vec*			RDF2Vec			OWL2Vec*		
		MLP	XGB	RF	MLP	XGB	RF	MLP	XGB	RF	MLP	XGB	RF
w/o necessary	ΔPr	-0.157	-0.109	-0.106	-0.095	-0.099	-0.089	-0.291	-0.296	-0.326	-0.265	-0.332	-0.269
	ΔRe	-0.137	-0.120	-0.153	-0.145	-0.131	-0.129	-0.329	-0.220	-0.277	-0.353	-0.208	-0.329
	ΔF1	-0.148	-0.113	-0.125	-0.117	-0.113	-0.107	-0.264	-0.225	-0.273	-0.270	-0.256	-0.260
only sufficient	ΔPr	0.932	1.000	0.973	0.981	1.000	0.988	0.957	0.969	0.893	0.921	0.986	0.917
	ΔRe	0.959	1.000	0.888	0.927	1.000	0.942	0.737	0.905	0.779	0.777	0.993	0.869
	ΔF1	0.950	1.000	0.945	0.954	1.000	0.967	0.898	0.964	0.896	0.885	0.993	0.925

Table 4: Explanation average length (Avg) and standard deviation (Std) for PPI prediction.

		RDF2Vec		OWL2Vec*	
		Avg	Std	Avg	Std
sufficient	MLP	5.6	3.9	5.3	3.5
	XGB	6.2	3.9	6.3	4.1
	RF	5.6	3.7	5.9	3.7
necessary	MLP	0.4	1.1	0.3	1.0
	XGB	0.4	1.1	0.3	1.0
	RF	0.4	1.3	0.3	1.1

Table 5: Explanation average length (Avg) and standard deviation (Std) for GDA prediction.

		RDF2Vev		OWL2Vec*	
		Avg	Std	Avg	Std
sufficient	MLP	5.6	7.1	5.5	7.8
	XGB	6.0	8.3	6.0	9.4
	RF	5.6	7.7	5.7	8.6
necessary	MLP	0.6	1.5	0.6	1.1
	XGB	0.5	1.3	0.5	1.2
	RF	0.7	1.7	0.7	1.4

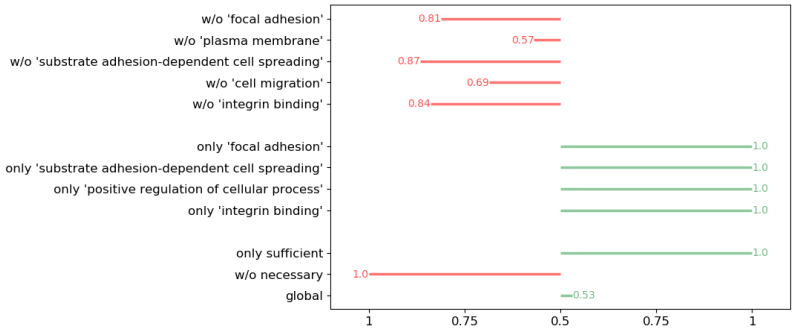
The first pair in our analysis consists of paxillin[2] and integrin α-4[3]. There is strong evidence for their interaction (Han et al. 2001) since integrin α-4 binds tightly to paxillin, leading to increased cell migration and an altered cytoskeletal organization that results in reduced cell spreading. Our explanations identify several aspects that are necessary and/or sufficient to explain the interaction and that strongly correlate with the known evidence: focal adhesion, substrate adhesion-dependent cell spreading, cell migration and integrin binding.

[2]https://www.uniprot.org/uniprot/P49023

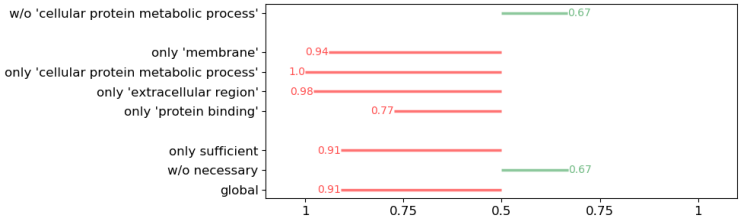
[3]https://www.uniprot.org/uniprot/P13612

Table 6: Explanations of PPI prediction models for four randomly selected pairs. For each pair, we provide a bar chart using different sets of disjoint common ancestors to represent the pair. On the x-axis, each bar represents the likelihood returned by the MLP model of the predicted class being correct. Classes are represented by colors (red for class 0 and green for class 1).

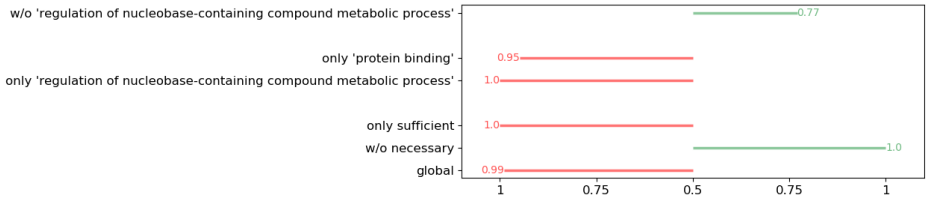
Paxillin – Integrin α-4
(True positive)



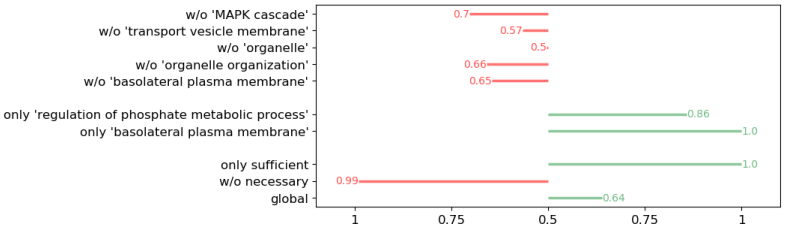
Pulmonary surfactant-associated protein B – Granulocyte-macrophage colony-stimulating factor receptor subunit α
(False negative)



Proline-rich 5-like – Guanine nucleotide-binding 3-like
(True Negative)



Protransforming growth factor α – Disks large homolog 2
(False Positive)



The proteins Pulmonary surfactant-associated protein B[4] and ganulocyte-macrophage colony-stimulating factor receptor subunit α[5] make up the second pair. Although the proteins share some necessary and/or sufficient semantic aspects, they are very general; therefore, the model does not predict the interaction. However, according to the literature, they are likely involved in the same pulmonary disease (Trapnell, Whitsett, and Nakata 2003). Both proteins are poorly described under the GO, which can explain why the relation prediction model fails.

The third pair includes the proline-rich 5-like protein[6] and the guanine nucleotide-binding 3-like protein[7]. The model

[4]https://www.uniprot.org/uniprot/P07988
[5]https://www.uniprot.org/uniprot/P15509
[6]https://www.uniprot.org/uniprot/Q6MZQ0
[7]https://www.uniprot.org/uniprot/Q9NVN8

predicts this as a negative pair, and the explanations confirm this, with the removal of the necessary shared semantic aspect resulting in a positive prediction. No interaction is known between these two proteins.

The Protransforming growth factor (TGF) α[8] and the Disks large homolog 2 (Dlg2)[9] compose the last pair and correspond to a false positive. The explanations highlight their participation in the MAPK cascade (central signaling pathways that regulate a wide variety of stimulated cellular processes, including proliferation, differentiation, apoptosis and stress response) as well as their co-location in the basolateral plasma membrane. It is intriguing to note that although there is no known interaction between these proteins, there is evidence of an interaction between highly similar proteins: TGF-β is regulated by Dlg5 and both proteins activate the MAPK cascade (Sezaki et al. 2013). We hypothesize this is not in fact a true negative pair but a still unknown PPI erroneously used as a negative example through random negative sampling.

6 Conclusion

Existing KG embedding methods are not explainable, which hinders their application in complex and critical domains. This is especially challenging in relation prediction, where understanding which KG semantic aspects are more relevant for a relationship between two KG entities can provide insightful knowledge about its mechanisms and help distinguish meaningful predictions from spurious correlations.

To address this challenge, we propose SEEK, a novel approach for learning and explaining representations of KG entity pairs based on their shared semantic space for relation prediction. Its explanatory mechanism is based on generating perturbed representations to identify the relevant semantic aspects of the KG that explain a relation; and since it does not require retraining of representations, it is particularly efficient. We evaluate SEEK on protein-protein interaction prediction and gene-disease association prediction, two complex and core tasks in the biomedical domain. SEEK clearly outperforms state-of-the-art learning representation methods in performance, while generating explanations that can identify critical factors driving biological phenomena.

In future work, we will conduct user studies with biomedical domain experts to evaluate SEEK explanations and also improve explanations by investigating the minimal set of shared semantic aspects required to adequately explain a relation.

Acknowledgments

C.P., S.S., and R.T.S. are funded by FCT, Portugal, through LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020). R.T.S. acknowledges the FCT PhD grant (ref. SFRH/BD/145377/2019). This work was also partially supported by the KATY project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017453, and by HfPT: Health from Portugal under the Portuguese Plano de Recuperação e Resiliência.

[8]https://www.uniprot.org/uniprot/P01135

[9]https://www.uniprot.org/uniprot/Q15700

References

Alshahrani, M.; Khan, M. A.; Maddouri, O.; Kinjo, A. R.; Queralt-Rosinach, N.; and Hoehndorf, R. 2017. Neuro-symbolic representation learning on biological knowledge graphs. *Bioinformatics* 33(17):2723–2730.

Asif, M.; Martiniano, H. F. M. C. M.; Vicente, A. M.; and Couto, F. M. 2018. Identifying disease genes using machine learning and gene functional similarities, assessed through gene ontology. *PLOS ONE* 13(12):1–15.

Barredo Arrieta, A.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; Garcia, S.; Gil-Lopez, S.; Molina, D.; Benjamins, R.; Chatila, R.; and Herrera, F. 2020. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion* 58:82–115.

Betz, P.; Meilicke, C.; and Stuckenschmidt, H. 2022. Adversarial explanations for knowledge graph embeddings. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2820–2826. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Bordes, A.; Usunier, N.; Garcia-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of NIPS 2013*, 2787–2795. Red Hook, NY, USA: Curran Associates Inc.

Breiman, L. 2001. Random forests. *Machine learning* 45(1):5–32.

Celebi, R.; Uyar, H.; Yasar, E.; Gumus, O.; Dikenelli, O.; and Dumontier, M. 2019. Evaluation of knowledge graph embedding approaches for drug-drug interaction prediction in realistic settings. *BMC bioinformatics* 20(1):1–14.

Chari, S.; Gruen, D. M.; Seneviratne, O.; and McGuinness, D. L. 2020. Foundations of explainable knowledge-enabled systems. In *Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges*. IOS Press. 23–48.

Chen, T., and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. New York, NY, USA: Association for Computing Machinery.

Chen, J.; Hu, P.; Jimenez-Ruiz, E.; Holter, O. M.; Antonyrajah, D.; and Horrocks, I. 2021. OWL2Vec*: Embedding of OWL ontologies. *Machine Learning* 1–33.

Chen, K.-H.; Wang, T.-F.; and Hu, Y.-J. 2019. Protein-protein interaction prediction using a hybrid feature representation and a stacked generalization scheme. *BMC Bioinformatics* 20(1):308.

Consortium, G. 2021. The gene ontology resource: enriching a GOld mine. *Nucleic Acids Research* 49(D1):D325–D334.

Couto, F. M., and Silva, M. J. 2011. Disjunctive shared

information between ontology concepts: application to gene ontology. *Journal of biomedical semantics* 2:1–16.

Ehrlinger, L., and Wöß, W. 2016. Towards a definition of knowledge graphs. *SEMANTiCS (Posters, Demos, SuCCESS)* 48(1-4):2.

Eilbeck, K.; Quinlan, A.; and Yandell, M. 2017. Settling the score: variant prioritization and mendelian disease. *Nature Reviews Genetics* 18(10):599–612.

Gad-Elrab, M. H.; Stepanova, D.; Tran, T.-K.; Adel, H.; and Weikum, G. 2020. Excut: Explainable embedding-based clustering over knowledge graphs. In *The Semantic Web–ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part I*, 218–237. Springer.

Gilpin, L. H.; Bau, D.; Yuan, B. Z.; Bajwa, A.; Specter, M.; and Kagal, L. 2018. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, 80–89. IEEE.

Grau, B. C.; Horrocks, I.; Motik, B.; Parsia, B.; Patel-Schneider, P.; and Sattler, U. 2008. OWL 2: The next step for OWL. *Journal of Web Semantics* 6(4):309–322.

Guidotti, R.; Monreale, A.; Ruggieri, S.; Turini, F.; Giannotti, F.; and Pedreschi, D. 2018. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)* 51(5):1–42.

Han, J.; Liu, S.; Rose, D. M.; Schlaepfer, D. D.; McDonald, H.; and Ginsberg, M. H. 2001. Phosphorylation of the integrin alpha-4 cytoplasmic domain regulates paxillin binding. *Journal of Biological Chemistry* 276(44):40903–40909.

Hoehndorf, R.; Schofield, P. N.; and Gkoutos, G. V. 2011. PhenomeNET: a whole-phenome approach to disease gene discovery. *Nucleic Acids Research* 39(18):e119.

Hogan, A.; Blomqvist, E.; Cochez, M.; d'Amato, C.; Melo, G. d.; Gutierrez, C.; Kirrane, S.; Gayo, J. E. L.; Navigli, R.; Neumaier, S.; et al. 2021. Knowledge graphs. *ACM Computing Surveys (CSUR)* 54(4):1–37.

Holzinger, A.; Biemann, C.; Pattichis, C. S.; and Kell, D. B. 2017. What do we need to build explainable AI systems for the medical domain? *arXiv preprint arXiv:1712.09923*.

Hu, L.; Wang, X.; Huang, Y.-A.; Hu, P.; and You, Z.-H. 2021. A survey on computational models for predicting protein–protein interactions. *Briefings in Bioinformatics* 22(5):bbab036.

Huntley, R. P.; Sawford, T.; Mutowo-Meullenet, P.; Shypitsyna, A.; Bonilla, C.; Martin, M. J.; and O'donovan, C. 2014. The GOA database: gene ontology annotation updates for 2015. *Nucleic Acids Research* 43(D1):D1057–D1063.

Ieremie, I.; Ewing, R. M.; and Niranjan, M. 2022. TransformerGO: predicting protein–protein interactions by modelling the attention between sets of gene ontology terms. *Bioinformatics*.

Kulmanov, M.; Liu-Wei, W.; Yan, Y.; and Hoehndorf, R. 2019. EL embeddings: geometric construction of models for the description logic EL++. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*.

Kulmanov, M.; Smaili, F. Z.; Gao, X.; and Hoehndorf, R. 2021. Semantic similarity and machine learning with ontologies. *Briefings in Bioinformatics* 22(4):bbaa199.

Köhler, S.; Gargano, M.; Matentzoglu, N.; Carmody, L. C.; Lewis-Smith, D.; Vasilevsky, N. A.; and Danis, D. e. a. 2020. The Human Phenotype Ontology in 2021. *Nucleic Acids Research* 49(D1):D1207–D1217.

Li, D.; Li, D.; Wang, C.; and Chen, Y. 2021. Network embedding method based on semantic information. In *2021 3rd International Conference on Advanced Information Science and System (AISS 2021)*, 1–6.

Mei, Y.; Chen, Q.; Lensen, A.; Xue, B.; and Zhang, M. 2022. Explainable artificial intelligence by genetic programming: A survey. *IEEE Transactions on Evolutionary Computation* 1–1.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Miller, G. A. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review* 63(2):81.

Mjolsness, E., and DeCoste, D. 2001. Machine learning for science: state of the art and future prospects. *science* 293(5537):2051–2055.

Mohamed, S. K.; Nounu, A.; and Nováček, V. 2021. Biological applications of knowledge graph embedding models. *Briefings in Bioinformatics* 22(2):1679–1693.

Nicholson, D. N., and Greene, C. S. 2020. Constructing knowledge graphs and their biomedical applications. *Computational and structural biotechnology journal* 18:1414–1428.

Nunes, S.; Sousa, R. T.; and Pesquita, C. 2021. Predicting gene-disease associations with knowledge graph embeddings over multiple ontologies. In *ISMB Annual Meeting - Bio-Ontologies*.

Palmonari, M., and Minervini, P. 2020. Knowledge graph embeddings and explainable AI. *Knowledge Graphs for Explainable Artificial Intelligence: Foundations, Applications and Challenges* 47:49.

Pezeshkpour, P.; Tian, Y.; and Singh, S. 2019. Investigating robustness and interpretability of link prediction via adversarial modifications. *NAACL-HLT*.

Piñero, J.; Ramírez-Anguita, J. M.; Saüch-Pitarch, J.; Ronzano, F.; Centeno, E.; Sanz, F.; and Furlong, L. I. 2019. The DisGeNET knowledge platform for disease genomics: 2019 update. *Nucleic Acids Research* 48(D1):D845–D855.

Portisch, J.; Heist, N.; and Paulheim, H. 2022. Knowledge graph embedding for data mining vs. knowledge graph embedding for link prediction–two sides of the same coin? *Semantic Web* 13(Preprint):1–24.

Ristoski, P., and Paulheim, H. 2016. RDF2Vec: RDF graph embeddings for data mining. In *International Semantic Web Conference*, 498–514. Springer.

Roscher, R.; Bohn, B.; Duarte, M. F.; and Garcke, J. 2020. Explainable machine learning for scientific insights and discoveries. *Ieee Access* 8:42200–42216.

Rossi, A.; Firmani, D.; Merialdo, P.; and Teofili, T. 2022a. Explaining link prediction systems based on knowledge graph embeddings. In *Proceedings of the 2022 International Conference on Management of Data*, 2062–2075.

Rossi, A.; Firmani, D.; Merialdo, P.; and Teofili, T. 2022b. Kelpie: an explainability framework for embedding-based link prediction models. *Proceedings of the VLDB Endowment* 15(12):3566–3569.

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1986. Learning representations by back-propagating errors. *Nature* 323(6088):533–536.

Sezaki, T.; Tomiyama, L.; Kimura, Y.; Ueda, K.; and Kioka, N. 2013. Dlg5 interacts with the TGF-beta receptor and promotes its degradation. *FEBS Letters* 587(11):1624–1629.

Smaili, F. Z.; Gao, X.; and Hoehndorf, R. 2019. OPA2Vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics* 35(12):2133–2140.

Sousa, R. T.; Silva, S.; and Pesquita, C. 2020. Evolving knowledge graph similarity for supervised learning in complex biomedical domains. *BMC Bioinformatics* 21(1):6.

Sousa, R. T.; Silva, S.; and Pesquita, C. 2021. evoKGsim+: a framework for tailoring knowledge graph-based similarity for supervised learning. In *ESWC 2021 Poster and Demo Track*.

Staab, S., and Studer, R. 2010. *Handbook on ontologies*. Springer-Verlag.

Szklarczyk, D.; Gable, A. L.; Nastou, K. C.; Lyon, D.; Kirsch, R.; Pyysalo, S.; Doncheva, N. T.; Legeay, M.; Fang, T.; Bork, P.; Jensen, L. J.; and von Mering, C. 2020. The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. *Nucleic Acids Research* 49(D1):D605–D612.

Trapnell, B. C.; Whitsett, J. A.; and Nakata, K. 2003. Pulmonary alveolar proteinosis. *New England Journal of Medicine* 349(26):2527–2539.

Van der Maaten, L., and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9(11).

Wang, Z.; Zhang, J.; Feng, J.; and Chen, Z. 2014. Knowledge Graph Embedding by Translating on Hyperplanes. In *Proceedings of the 28th AAAI Conference on Artificial Intelligence*, 1112–1119. AAAI Press.

Wang, Q.; Mao, Z.; Wang, B.; and Guo, L. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29(12):2724–2743.

Watson, D. S.; Gultchin, L.; Taly, A.; and Floridi, L. 2021. Local explanations via necessity and sufficiency: Unifying theory and practice. In *Uncertainty in Artificial Intelligence*, 1382–1392. PMLR.

Xiong, B.; Potyka, N.; Tran, T.-K.; Nayyeri, M.; and Staab, S. 2022. Faithful Embeddings for EL++ Knowledge Bases. In *International Semantic Web Conference*, 22–38. Springer.

Yang, B.; tau Yih, W.; He, X.; Gao, J.; and Deng, L. 2015. Embedding entities and relations for learning and inference in knowledge bases.

Zhang, S.-B., and Tang, Q.-R. 2016. Protein–protein interaction inference based on semantic similarity of gene ontology terms. *Journal of Theoretical Biology* 401:30–37.

Appendix C

Biomedical Knowledge Graph Embeddings with Negative Statements

Biomedical Knowledge Graph Embeddings with Negative Statements

Rita T. Sousa[1 [0000−0002−7241−8970]], Sara Silva[1 [0000−0001−8223−4799]], Heiko Paulheim[2 [0000−0003−4386−8195]], and Catia Pesquita[1 [0000−0002−1847−9393]]

[1] LASIGE, Faculdade de Ciências da Universidade de Lisboa, Portugal
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.pt
[2] Data and Web Science Group, Universität Mannheim, Germany
heiko.paulheim@uni-mannheim.de

Abstract. A knowledge graph is a powerful representation of real-world entities and their relations. The vast majority of these relations are defined as positive statements, but the importance of negative statements is increasingly recognized, especially under an Open World Assumption. Explicitly considering negative statements has been shown to improve performance on tasks such as entity summarization and question answering or domain-specific tasks such as protein function prediction. However, no attention has been given to the exploration of negative statements by knowledge graph embedding approaches despite the potential of negative statements to produce more accurate representations of entities in a knowledge graph.

We propose a novel approach, TrueWalks, to incorporate negative statements into the knowledge graph representation learning process. In particular, we present a novel walk-generation method that is able to not only differentiate between positive and negative statements but also take into account the semantic implications of negation in ontology-rich knowledge graphs. This is of particular importance for applications in the biomedical domain, where the inadequacy of embedding approaches regarding negative statements at the ontology level has been identified as a crucial limitation.

We evaluate TrueWalks in ontology-rich biomedical knowledge graphs in two different predictive tasks based on KG embeddings: protein-protein interaction prediction and gene-disease association prediction. We conduct an extensive analysis over established benchmarks and demonstrate that our method is able to improve the performance of knowledge graph embeddings on all tasks.

Keywords: Knowledge Graph · Knowledge Graph Embedding · Negative Statements · Biomedical Applications.

1 Introduction

Knowledge Graphs (KGs) represent facts about real-world entities and their relations and have been extensively used to support a range of applications from question-answering and recommendation systems to machine learning and analytics [17]. KGs have taken to the forefront of biomedical data through their ability to describe and interlink information about biomedical entities such as genes, proteins, diseases and patients, structured

according to biomedical ontologies. This supports the analysis and interpretation of biological data, for instance, through the use of semantic similarity measures [32]. More recently, a spate of KG embedding methods [42] have emerged in this space and have been successfully employed in a number of biomedical applications [28]. The impact of KG embeddings in biomedical analytics is expected to increase in tandem with the growing volume and complexity of biomedical data. However, this success relies on the expectation that KG embeddings are semantically meaningful representations of the underlying biomedical entities.

Regardless of their domain, the vast majority of KG facts are represented as positive statements, e.g. $(\textit{hemoglobin}, \textit{hasFunction}, \textit{oxygen transport})$. Under a Closed World Assumption, negative statements are not required, since any missing fact can be assumed as a negative. However, real-world KGs reside under the Open World Assumption where non-stated negative facts are formally indistinguishable from missing or unknown facts, which can have important implications across a variety of tasks.

The importance of negative statements is increasingly recognized [2,10]. For example, in the biomedical domain, the knowledge that a patient does not exhibit a given symptom or a protein does not perform a specific function is crucial for both clinical decision-making and biomedical insight. While ontologies are able to express negation and the enrichment of KGs with interesting negative statements is gaining traction, existing KG embedding methods are not able to adequately utilize them [21], which ultimately results in less accurate representations of entities.

We propose True Walks, to the best of our knowledge, the first-ever approach that is able to incorporate negative statements into the KG embedding learning process. This is fundamentally different from other KG embedding methods, which produce negative statements by negative random sampling strategies to train representations that bring the representations of nodes that are linked closer, while distancing them from the negative examples. TrueWalks uses explicit negative statements to produce entity representations that take into account both existing attributes and lacking attributes. For example, for the negative statement $(\textit{Bruce Willis}, \textit{NOT birthPlace}, \textit{U.S.})$, our representation would be able to capture the similarity between Bruce Willis and Ryan Gosling, since neither was born in the U.S (see Figure 1). The explicit declaration of negative statements such as these is an important aspect of more accurate representations, especially when they capture unexpected negative statements (i.e., most people would expect that both actors are U.S. born). Using TrueWalks, Bruce Willis and Ryan Gosling would be similar not just because they are both actors but also because neither was born in the U.S.

True Walks generates walks that can distinguish between positive and negative statements and consider the semantic implications of negation in KGs that are rich in ontological information, particularly in regard to inheritance. This is of particular importance for applications in the biomedical domain, where the inadequacy of embedding approaches regarding negative statements has been identified as a crucial limitation [21]. We demonstrate that the resulting embeddings can be employed to determine semantic similarity or as features for relation prediction. We evaluate the effectiveness of our approach in two different tasks, protein-protein interaction prediction and gene-disease association prediction, and show that our method improves performance over state-of-the-art embedding methods and popular semantic similarity measures.

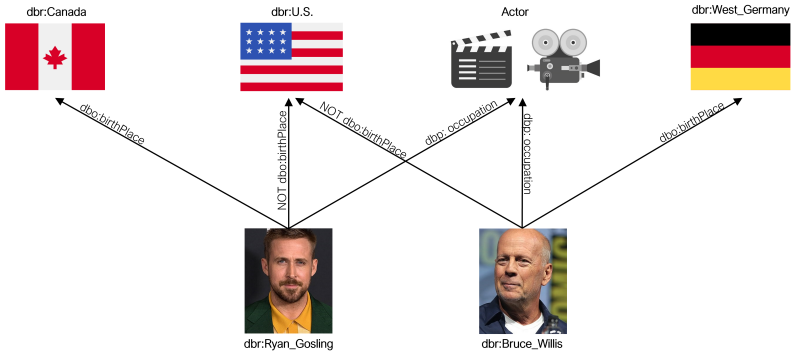


Fig. 1. A DBPedia example motivating the negative statements problem. The author of Bruce Willis' picture is Gage Skidmore.

Our contributions are as follows:

- We propose TrueWalks, a novel method to generate random walks on KGs that are aware of negative statements and results in the first KG embedding approach that considers negative statements.
- We develop extensions of popular path-based KG embedding methods implementing the TrueWalks approach.
- We enrich existing KGs with negative statements and propose benchmark datasets for two popular biomedical KG applications: protein-protein interaction (PPI) prediction and gene-disease association (GDA) prediction.
- We report experimental results that demonstrate the superior performance of TrueWalks when compared to state-of-the-art KG embedding methods.

2 Related Work

2.1 Exploring Negative Statements

Approaches to enrich existing KGs with interesting negative statements have been proposed both for general-purpose KGs such as Wikidata [3] and for domain-specific ones such as the Gene Ontology (GO) [11,44]. Exploring negative statements has been demonstrated to improve the performance of various applications. [2] developed a method to enrich Wikidata with interesting negative statements and its usage improved the performance on entity summarization and decision-making tasks. [44] have designed a method to enrich the GO [14] with relevant negative statements indicating that a protein does not perform a given function and demonstrated that a balance between positive and negative annotations supports a more reasonable evaluation of protein function prediction methods. Similarly, [11] enriched the GO with negative statements and demonstrated an associated increase in protein function prediction performance. The relevance of negative annotations has also been recognized in the prediction of gene-phenotype

associations in the context of the Human-Phenotype Ontology (HP) [22], but the topic remains unexplored [25]. It should be highlighted that KG embedding methods have not been employed in any of these approaches to explore negative statements.

2.2 Knowledge Graph Embeddings

KG embedding methods map entities and their relations expressed in a KG into a lower-dimensional space while preserving the underlying structure of the KG and other semantic information [42]. These entity and relation embedding vectors can then be applied to various KG applications such as link prediction, entity typing, or triple classification. In the biomedical domain, KG embeddings have been used in machine learning-based applications in which they are used as input in classification tasks or to predict relations between biomedical entities. [21] provides an overview of KG embedding-based approaches for biomedical applications.

Translational models, which rely on distance-based scoring functions, are some of the most widely employed KG embedding methods. A popular method, TransE [6], assumes that if a relation holds between two entities, the vector of the head entity plus the relation vector should be close to the vector of the tail entity in the vector space. TransE has the disadvantage of not handling one-to-many and many-to-many relationships well. To address this issue, TransH [43] introduces a relation-specific hyperplane for each relation and projects the head and tail entities into the hyperplane. TransR [23] builds entity and relation embeddings in separate entity space and relation spaces.

Semantic matching approaches are also well-known and use similarity-based scoring functions to capture the latent semantics of entities and relations in their vector space representations. For instance, DistMult [48] employs tensor factorization to embed entities as vectors and relations as diagonal matrices.

2.3 Walk-Based Embeddings

More recently, random walk-based KG embedding approaches have emerged. These approaches are built upon two main steps: (i) producing entity sequences from walks in the graph to produce a corpus of sequences that is akin to a corpus of word sequences or sentences; (2) using those sequences as input to a neural language model [27] that learns a latent low-dimensional representation of each entity within the corpus of sequences.

DeepWalk [31] first samples a set of paths from the input graph using uniform random walks. Then it uses those paths to train a skip-gram model, originally proposed by the word2vec approach for word embeddings [27]. Node2vec [16] introduces a different biased strategy for generating random walks and exploring diverse neighborhoods. The biased random walk strategy is controlled by two parameters: the likelihood of visiting immediate neighbors (breadth-first search behavior), and the likelihood of visiting entities that are at increasing distances (depth-first search behavior). Neither DeepWalk nor node2vec take into account the direction or type of the edges. Metapath2vec [8] proposes random walks driven by metapaths that define the node type order by which the random walker explores the graph. RDF2Vec [35] is inspired by the node2vec strategy but it considers both edge direction and type making it particularly suited to KGs.

OWL2Vec* [7] was designed to learn ontology embeddings and it also employs direct walks on the graph to learn graph structure.

2.4 Tailoring Knowledge Graph Embeddings

Recent KG embedding approaches aim to tailor representations by considering different semantic, structural or lexical aspects of a KG and its underlying ontology. Approaches such as EL [20] and BoxEL [45] embeddings are geometric approaches that account for the logical structure of the ontology (e.g., intersection, conjunction, existential quantifiers). OWL2Vec* [7] and OPA2Vec [37] take into consideration the lexical portion of the KG (i.e., labels of entities) when generating graph walks or triples. OPA2Vec also offers the option of using a pre-trained language model to bootstrap the KG embedding. Closer to our approach, OLW2Vec* contemplates the declaration of inverse axioms to enable reverse path traversal, however, this option was found lacking for the biomedical ontology GO. Finally, different approaches have been proposed to train embeddings that are aware of the order of entities in a path, such as [51] and [34], which extend TransE and RDF2Vec, respectively.

3 Methods

3.1 Problem Formulation

In this work, we address the task of learning a relation between two KG entities (which can belong to the same or different KGs) when the relation itself is not encoded in the KG. We employ two distinct approaches: (1) using the KG embeddings of each entity

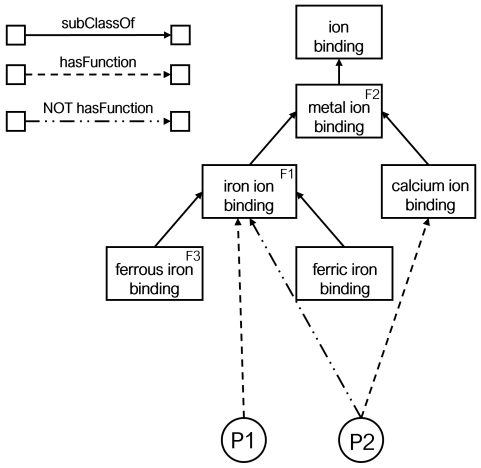


Fig. 2. A GO KG subgraph motivating the *reverse inheritance* problem.

as features for a machine learning algorithm and (2) comparing the KG embeddings directly through a similarity metric.

We target ontology-rich KGs that use an ontology to provide rich descriptions of real-world entities instead of focusing on describing relations between entities themselves. These KGs are common in the biomedical domain. As a result, the KG's richness lies in the TBox, with a comparatively less complex ABox, since entities have no links between them. We focus on Web Ontology Language (OWL) [15] ontologies since biomedical ontologies are typically developed in OWL or have an OWL version.

Biomedical entities in a KG are typically described through positive statements that link them to an ontology. For instance, to state that a protein P performs a function F described under the GO, a KG can declare the axiom $P \sqsubseteq \exists hasFunction.F$. However, the knowledge that a given protein does not perform a function can also be relevant, especially to declare that a given protein does not have an activity typical of its homologs [12]. Likewise, the knowledge that a given disease does not exhibit a particular phenotype is also decisive in understanding the relations between diseases and genes [25]. We consider the definition of grounded negative statements proposed by [2] as $\neg(s, p, o)$ which is satisfied if $(s, p, o) \notin KG$ and expressed as a *NegativeObjectPropertyAssertion*[3]. Similar to what was done in [2], we do not have a negative object property assertion for every missing triple. Negative statements are only included if there is clear evidence that a triple does not exist in the domain being captured. Taking the protein example, negative object property assertions only exist when it has been demonstrated that a protein does not perform a particular function.

An essential difference between a positive and a negative statement of this kind is related to the implied inheritance of properties exhibited by the superclasses or subclasses of the assigned class. Let us consider that $(P_1, \textit{hasFunction}, F_1)$ and $(F_1, \textit{subClassOf}, F_2)$. This implies that $(P_1, \textit{hasFunction}, F_2)$, since an individual with a class assignment also belongs to all superclasses of the given class, e.g., a protein that performs *iron ion binding* also performs *metal ion binding* (see Figure 2). This implication is easily captured by directed walk generation methods that explore the declared subclass axioms in an OWL ontology. However, when we have a negative statement, such as $\neg(P_2, \textit{hasFunction}, F_1)$, it does not imply that $\neg(P_2, \textit{hasFunction}, F_2)$. There are no guarantees that a protein that does not perform *iron ion binding* also does not perform *metal ion binding*, since it can very well, for instance, perform *calcium ion binding*. However, for $(F_3, \textit{subClassOf}, F_1)$ the negative statement $\neg(P_2, \textit{hasFunction}, F_1)$ implies that $\neg(P_2, \textit{hasFunction}, F_3)$, as a protein that does not perform *iron ion binding* also does not perform *ferric iron binding* nor *ferrous iron binding*. Therefore, we need to be able to declare that protein P_1 performs both functions F_1 and F_3, but that P_2 performs F_1 but not F_3. Since OWL ontologies typically declare subclass axioms, there is no opportunity for typical KG embedding methods to explore the reverse paths that would more accurately represent a negative statement.

The problem we tackle is then two-fold: how can the *reverse inheritance* implied by negative statements be adequately explored by walk-based KG embedding methods, and how can these methods distinguish between negative and positive statements.

[3] https://www.w3.org/TR/owl2-syntax/#Negative_Object_Property_Assertions

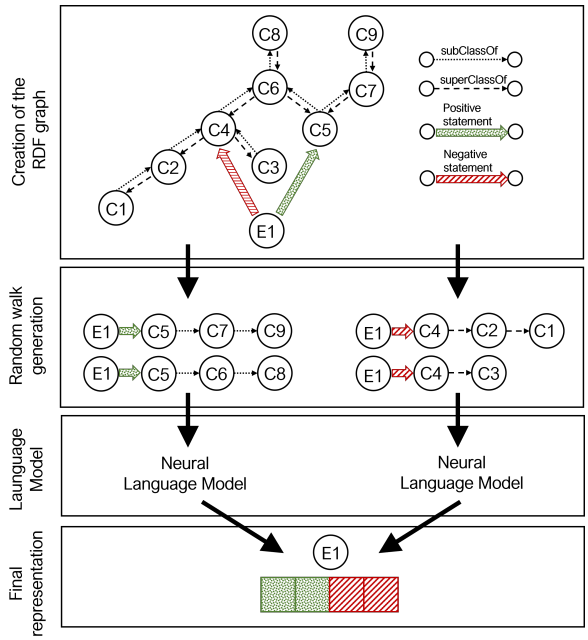


Fig. 3. Overview of the TrueWalks method with the four main steps: (i) creation of the RDF graph, (ii) random walk generation with negative statements; (iii) neural language models, and (iv) final representation.

3.2 Overview

An overview of TrueWalks, the method we propose, is shown in Figure 3. The first step is the transformation of the KG into an RDF Graph. Next, our novel random walk generation strategy that is aware of positive and negative statements is applied to the graph to produce a set of entity sequences. The positive and negative entity walks are fed to neural language models to learn a dual latent representation of the entities. TrueWalks has two variants: one that employs the classical skip-gram model to learn the embeddings (TrueWalks), and one that employs a variation of skip-gram that is aware of the order of entities in the walk (TrueWalksOA, i.e. order-aware).

3.3 Creation of the RDF Graph

The first step is the conversion of an ontology-rich KG into an RDF graph. This is a directed, labeled graph, where the edges represent the named relations between two resources or entities, represented by the graph nodes[4]. We perform the transformation according to the *OWL to RDF Graph Mapping* guidelines defined by the W3C[5]. Simple

[4]https://www.w3.org/RDF/

[5]https://www.w3.org/TR/owl2-mapping-to-rdf/

axioms can be directly transformed into RDF triples, such as subsumption axioms for atomic entities or data and annotation properties associated with an entity. Axioms involving complex class expressions are transformed into multiple triples which typically require blank nodes.

Let us consider the following existential restriction of the class *obo:GO_0034708 (methyltransferase complex)* that encodes the fact that a methyltransferase complex is part of at least one intracellular anatomical structure:

ObjectSomeValuesFrom(obo:BFO_0000050 (part of),
obo:GO_0005622 (intracellular anatomical structure))

Its conversion to RDF results in three triples:

(obo:GO_0034708, rdfs:subClassOf, _:x)
(_:x, owl:someValuesFrom,obo:GO_0005622)
(_:x, owl:onProperty,obo:BFO_0000050)

where _:x denotes a blank node.

3.4 Random Walk Generation with Negative Statements

The next step is to generate the graph walks that will make up the corpus (see Algorithm 1). For a given graph $G = (V, E)$ where E is the set of edges and V is the set of vertices, for each vertex $v_r \in V_r$, where V_r is the subset of individuals for which we want to learn representations, we generate up to w graph walks of maximum depth d rooted in vertex v_r. We employ a depth-first search algorithm, extending on the basic approach in [35]. At the first iteration, we can find either a positive or negative statement. From then on, walks are biased: a positive statement implies that whenever a subclass edge is found it is traversed from subclass to superclass, whereas a negative statement results in a traversal of subclass edges in the opposite direction (see also Figure 3). This generates paths that follow the pattern $v_r \rightarrow e_{1i} \rightarrow v_{1i} \rightarrow e_{2i}$. The set of walks is split in two, negative statement walks and positive statement walks. This will allow the learning of separate latent representations, one that captures the positive aspect and one that captures the negative aspect.

An important aspect of our approach is that, since OWL is converted into an RDF graph for walk-based KG embedding methods, a negative statement declared using a simple object property assertion (e.g. *notHasFunction*) could result in the less accurate path: *Protein P* $\rightarrow$ *notHasFunction* $\rightarrow$ *iron ion binding* $\rightarrow$ *subClassOf* $\rightarrow$ *ion binding*. Moreover, random walks directly over the *NegativeObjectPropertyAssertion*, since it is decomposed into multiple triples using blank nodes, would also result in inaccurate paths. However, our algorithm produces more accurate paths, e.g.: *Protein P* $\rightarrow$ *notHasFunction* $\rightarrow$ *iron ion binding* $\rightarrow$ *superClassOf* $\rightarrow$ *ferric iron binding* by adequately processing the *NegativeObjectPropertyAssertion*.

3.5 Neural Language Models

We employ two alternative approaches to learn a latent representation of the individuals in the KG. For the first approach, we use the skip-gram model [27], which predicts the context (neighbor entities) based on a target word, or in our case a target entity.

Algorithm 1 Walk generation for one entity using TrueWalks. The function GET NON VISITED NEIGHBOURS(status) is used to generate the random walks using a depth-first search. It gets the neighbors of a given node that have not yet been visited in previous iterations. If the status is negative (which means that the first step in the walk was made with a negative statement), the neighbors will include all the non-visited neighbors except those connected through subclass statements, and if the status is positive, it will include all the neighbors except those connected through superclass statements.

$d \leftarrow max\_depth\_walks$
$w \leftarrow max\_number\_of\_walks$
$ent \leftarrow root\_entity$
function GET TRUEWALKS(ent)
  $pos\_walks \leftarrow$ GET RANDOM WALKS$(ent, positive)$
  $neg\_walks \leftarrow$ GET RANDOM WALKS$(ent, negative)$
  return pos_walks, neg_walks
function GET RANDOM WALKS($ent, status$)
  while $len(walks) < w$ **do**
    $walk \leftarrow ent$
    $depth \leftarrow 1$
    while $depth < d$ **do**
      $last \leftarrow len(walk) == d$
      $e, v \leftarrow$ GET NEIGHBOR$(walk, status, last)$
      if $e, v == None$ **then**
        break
      $walk$.append(e, v)
      $depth++$
    $walks$.append$(walk)$
  return $walks$
function GET NEIGHBOR($walk, status, last$)
  $n \leftarrow$ GET NON VISITED NEIGHBORS$(status)$
  if $len(n) == 0$ & $len(walk) > 2$ **then**
    $e, v \leftarrow walk[-2], walk[-1]$
    ADD VISITED NEIGHBORS$(e, v, len(walk) - 2, status)$
    return $None$
  $e, v \leftarrow n[rand()]$
  if $last$ **then**
    ADD VISITED NEIGHBORS$(e, v, len(walk), status)$
  return e, v

Let $f : E \rightarrow \mathbb{R}^d$ be the mapping function from entities to the latent representations we will be learning, where d is the number of dimensions of the representation (f is then a matrix $|E| \times d$). Given a context window c, and a sequence of entities $e_1, e_2, e_3, ..., e_L$, the objective of the skip-gram model is to maximize the average log probability p:

$$\frac{1}{L} \sum_{l=1}^{L} \log p(e_{l+c}|e_l) \tag{1}$$

where $p(e_{l+c}|e_l)$ is calculated using the softmax function:

$$p(e_{l+c}|e_l) = \frac{\exp(f(e_{l+c}) \cdot f(e_l))}{\sum_{e=1}^{E} \exp(f(e) \cdot f(e_l))} \quad (2)$$

where $f(e)$ is the vector of the entity e.

To improve computation time, we employ a negative sampling approach based in [27] that minimizes the number of comparisons required to distinguish the target entity, by taking samples from a noise distribution using logistic regression, where there are k negative samples for each entity.

The second approach is the structured skip-gram model [24], a variation of skip-gram that is sensitive to the order of words, or in our case, entities in the graph walks. The critical distinction of this approach is that, instead of using a single matrix f, it creates $c \times 2$ matrices, $f_{-c}, ..., f_{-2}, f_{-1}, f_1, ..., f_c$, each dedicated to predicting a specific relative position to the entity. To make a prediction $p(e_{l+c}|e_l)$, the method selects the appropriate matrix f_l.

The neural language models are applied separately to the positive and negative walks, producing two representations for each entity.

3.6 Final Representations

The two representations of each entity need to be combined to produce a final representation. Different vector operations can, in principle, be employed, such as the Hadamard product or the L1-norm. However, especially since we will employ these vectors as inputs for machine learning methods, we would like to create a feature space that allows the distinction between the negative and positive representations, motivating us to use a simple concatenation of vectors.

4 Experiments

We evaluate our novel approach on two biomedical tasks: protein-protein interaction (PPI) prediction and gene-disease association (GDA) prediction[39]. These two challenges have significant implications for understanding the underlying mechanisms of biological processes and disease states.

Both tasks are modeled as relation prediction tasks. For PPI prediction, we employ TrueWalks embeddings both as features for a supervised learning algorithm and directly for similarity-based prediction. For GDA prediction, since embeddings for genes and diseases are learned over two different KGs, we focus only on supervised learning. We employ a Random Forest algorithm across all classification experiments with the same parameters (see the supplementary file for details).

4.1 Data

Our method takes as input an ontology file, instance annotation file and a list of instance pairs. We construct the knowledge graph (KG) using the RDFlib package [5], which

Table 1. Statistics for each KG regarding classes, instances, nodes, edges, positive and negative statements.

	GO_{PPI}	GO_{GDA}	HP_{GDA}
Classes	50918	50918	17060
Literals and blank nodes	532373	532373	442246
Edges	1425102	1425102	1082859
Instances	440	755	162
Positive statements	7364	10631	4197
Negative statements	8579	8966	225

parses the ontology file in OWL format and processes the annotation file to add edges to the RDFlib graph. The annotation file contains both positive and negative statements which are used to create the edges in the graph.

Protein-Protein Interaction Prediction Predicting protein-protein interactions is a fundamental task in molecular biology that can explore both sequence and functional information [18]. Given the high cost of experimentally determining PPI, computational methods have been proposed as a solution to the problem of finding protein pairs that are likely to interact and thus provide a selection of good candidates for experimental analysis. In recent years, a number of approaches for PPI prediction based on functional information as described by the GO have been proposed [50,20,37,38,21]. The GO contains over 50000 classes that describe proteins or genes according to the molecular functions they perform, the biological processes they are involved in, and the cellular components where they act.

The GO KG is built by integrating three sources: the GO itself [14], the Gene Ontology Annotation (GOA) data [13], and negative GO annotations [44] (details on the KG building method and data sources are available in the supplementary file). A GO annotation associates a Uniprot protein identifier with a GO class that describes it. We downloaded the GO annotations corresponding to positive statements from the GOA database for human species. For each protein P in the PPI dataset and each of its association statements to a function F in GOA, we add the assertion $(P, \mathit{hasFunction}, F)$. We employ the negative GO associations produced in [44], which were derived from expert-curated annotations of protein families on phylogenetic trees. For each protein P in the PPI dataset and each of its association statements to a function F in the negative GO associations dataset, we add a negative object property assertion. To do so, we use metamodeling (more specifically, punning [6]) and represent each ontology class as both a class and an individual. This situation translates into using the same IRI. Then, we use a negative object property assertion to state that the individual representing a biomedical entity is not connected by the object property expression to the individual representing an ontology class. Table 1 presents the GO KG statistics.

The target relations to predict are extracted from the STRING database [40]. We considered the following criteria to select protein pairs: (i) protein interactions must be

[6] https://www.w3.org/TR/owl2-new-features/#F12:_Punning

extracted from curated databases or experimentally determined (as opposed to computationally determined); (ii) interactions must have a confidence score above 0.950 to retain only high confidence interaction; (iii) each protein must have at least one positive GO association and one negative GO association. The PPI dataset contains 440 proteins, 1024 interacting protein pairs, and another 1024 pairs generated by random negative sampling over the same set of proteins.

Gene-Disease Association Prediction Predicting the relation between genes and diseases is essential to understand disease mechanisms and identify potential biomarkers or therapeutic targets [9]. However, validating these associations in the wet lab is expensive and time-consuming, which fostered the development of computational approaches to identify the most promising associations to be further validated. Many of these explore biomedical ontologies and KGs [41,49,36,4,26] and some recent approaches even apply KG embedding methods such as DeepWalk [1] or OPA2Vec [37,30].

For GDA prediction, we have used the GO KG, the Human Phenotype Ontology (HP) KG (created from the HP file and HP annotations files), and a GDA dataset. Two different ontologies are used to describe each type of entity. Diseases are described under the HP and genes under the GO. We built GO KG in the same fashion as in the PPI experiment, but instead of having proteins linked to GO classes, we have genes associated with GO classes. Regarding HP KG, HP [22] describes phenotypic abnormalities found in human hereditary diseases. The HP annotations link a disease to a specific class in the HP through both positive and negative statements.

The target relations to predict are extracted from DisGeNET [33], adapting the approach described in [30] to consider the following criterion: each gene (or disease) must have at least one positive GO (or HP) association and one negative GO (or HP) association. This resulted in 755 genes, 162 diseases, and 107 gene-disease relations. To create a balanced dataset, we sampled random negative examples over the same genes and diseases. Table 1 describes the created KGs.

4.2 Results and Discussion

We compare TrueWalks against ten state-of-the-art KG embedding methods: TransE, TransH, TransR, ComplEx, distMult, DeepWalk, node2vec, metapath2vec, OWL2Vec* and RDF2Vec. TransE, TransH and TransR are representative methods of translational models. ComplEx and distMult are semantic matching methods. They represent a bottom-line baseline with well-known KG embedding methods. DeepWalk and node2vec are undirected random walk-based methods, and OWL2Vec* and RDF2Vec are directed walk-based methods. These methods represent a closer approach to ours, providing a potentially stronger baseline. Each method is run with two different KGs, one with only positive statements and one with both positive and negative statements. In this second KG, we declare the negative statements as an object property, so positive and negative statements appear as two distinct relation types. The size of all the embeddings is 200 dimensions across all experiments (details on parameters can be found in the supplementary file), with TrueWalks generating two 100-dimensional vectors, one for the positive statement-based representation and one for the negative, which are concatenated to produce the final 200-dimensional representation.

Relation Prediction using Machine Learning To predict the relation between a pair of entities e_1 and e_2 using machine learning, we take their vector representations and combine them using the binary Hadamard operator to represent the pair: $r(e_1, e_2) = v_{e_1} \times v_{e_2}$. The pair representations are then fed into a Random Forest algorithm for training using Monte Carlo cross-validation (MCCV) [46]. MCCV is a variation of traditional k-fold cross-validation in which the process of dividing the data into training and testing sets (with β being the proportion of the dataset to include in the test split) is repeated M times. Our experiments use MCCV with $M = 30$ and $\beta = 0.3$. For each run, the predictive performance is evaluated based on recall, precision and weighted average F-measure. Statistically significant differences between TrueWalks and the other methods are determined using the non-parametric Wilcoxon test at $p < 0.05$.

Table 2 reports the median scores for both PPI and GDA prediction. The top half contains the results of the first experiment where we compare state-of-the-art methods using only the positive statements to TrueWalks (at the bottom) which uses both types. The results reveal that the performance of TrueWalks is significantly better than the

Table 2. Median precision, recall, and F-measure (weighted average F-measure) for PPI and GDA prediction. TrueWalks performance values are italicized/underlined when improvements are statistically significant with p-value < 0.05 for the Wilcoxon test against the positive (Pos)/positive and negative (Pos+Neg) variants of other methods. The best results are in bold.

	Method	PPI Prediction			GDA Prediction		
		Precision	Recall	F-measure	Precision	Recall	F-measure
Pos	TransE	0.553	0.546	0.554	0.533	0.538	0.531
	TransH	0.566	0.562	0.566	0.556	0.563	0.548
	TransR	0.620	0.607	0.616	0.594	0.600	0.592
	ComplEx	0.680	0.659	0.679	0.597	0.625	0.598
	distMult	0.765	0.737	0.754	0.585	0.600	0.575
	DeepWalk	0.813	0.836	0.822	0.618	0.646	0.629
	node2vec	0.826	0.741	0.794	0.643	0.616	0.644
	metapath2vec	0.562	0.563	0.561	0.554	0.531	0.549
	OWL2Vec*	0.833	0.806	0.823	0.652	0.656	0.646
	RDF2Vec	0.831	0.826	0.828	0.623	0.625	0.615
Pos+Neg	TransE	0.584	0.582	0.585	0.597	0.585	0.586
	TransH	0.573	0.572	0.570	0.563	0.554	0.554
	TransR	0.722	0.678	0.704	0.633	0.625	0.630
	ComplEx	0.750	0.720	0.740	0.549	0.545	0.545
	distMult	0.813	0.740	0.784	0.530	0.523	0.534
	DeepWalk	0.843	0.834	0.841	0.615	0.646	0.630
	node2vec	0.847	0.734	0.798	0.614	0.594	0.621
	metapath2vec	0.557	0.569	0.558	0.527	0.531	0.522
	OWL2Vec*	0.860	0.812	0.840	0.654	0.600	0.645
	RDF2Vec	0.847	**0.844**	0.845	0.625	**0.661**	0.630
	TrueWalks	***0.870***	0.817	*0.846*	***0.667***	0.625	***0.661***
	TrueWalksOA	*0.868*	0.836	***0.858***	*0.661*	0.616	*0.654*

other methods, improving both precision and F-measure. An improvement in precision, which is not always accompanied by an increase in recall, confirms the hypothesis that embeddings that consider negative statements produce more accurate representations of entities, which allows a better distinction of true positives from false positives.

A second experiment employs a KG with both negative and positive statements for all methods. Our method can accurately distinguish between positive statements and negative statements, as discussed in subsection 3.4. For the remaining embedding methods, we declare the negative statements as an object property so that these methods distinguish positive and negative statements as two distinct types of relation. This experiment allows us to test whether TrueWalks, which takes into account the positive or negative status of a statement, can improve the performance of methods that handle all statements equally regardless of status.

The bottom half of Table 2 shows that both variants of TrueWalks improve on precision and F-measure for both tasks when compared with the state-of-the-art methods using both positive and negative statements. This experiment further shows that the added information given by negative statements generally improves the performance of most KG embedding methods. However, no method surpasses TrueWalks, likely due to its ability to consider the semantic implications of inheritance and walk direction, especially when combined with the order-aware model.

Comparing the two variants of TrueWalks demonstrates that order awareness does not improve performance in most cases. However, TrueWalksOA improves on precision and F-measure for all other state-of-the-art methods. These results are not unexpected since the same effect was observed in other order-aware embedding methods [34].

Regarding the statistical tests, TrueWalks performance values are italicized/underlined in Table 2 when improvements over all other methods are statistically significant, except when comparing TrueWalks with OWL2Vec* for GDA, since in this particular case the improvement is not statistically significant.

Relation Prediction using Semantic Similarity We also evaluate all methods in PPI prediction using KG embedding-based semantic similarity, computed as the cosine similarity between the vectors of each protein in a pair. Adopting the methodology employed by [20] and [45], for each positive pair e_1 and e_2 in the dataset, we compute the similarity between e_1 and all other entities and identify the rank of e_2. The performance was measured using recall at rank 10[7], recall at rank 100, mean rank, and the area under the ROC curve (Table 3). Results show that TrueWalksOA achieves the top performance across all metrics, but TrueWalks is bested by RDF2Vec on all metrics except Hits@10, by OWL2Vec* on Hits@100 and by node2vec on Hits@10.

To better understand these results, we plotted the distribution of similarity values for positive and negative pairs in Figure 4. There is a smaller overlap between negative and positive pairs similarities for TrueWalksOA, which indicates that considering both the status of the function assignments and the order of entities in the random walks

[7] Since we compute the similarity score for all possible pairs to simulate a more realistic scenario where a user is presented with a ranked list of candidate interactions, the task is several degrees more difficult to performant and all KG embedding methods have a recall score of 0 at rank 1. As a result, we have excluded the results for this metric from our analysis.

Table 3. Hits@10, Hits@100, mean rank, and ROC-AUC for PPI prediction using cosine similarity obtained with different methods. In bold, the best value for each metric.

	Method	Hits@10	Hits@100	MeanRank	AUC
Pos	TransE	0.013	0.125	103.934	0.538
	TransH	0.013	0.134	102.703	0.543
	TransR	0.037	0.196	81.916	0.636
	ComplEx	0.080	0.261	64.558	0.689
	distMult	0.112	0.340	46.512	0.803
	DeepWalk	0.125	0.380	35.406	0.847
	node2vec	0.163	0.375	37.275	0.827
	metapath2vec	0.017	0.151	98.445	0.558
	OWL2Vec*	0.152	0.386	33.192	0.860
	RDF2Vec	0.133	0.391	32.419	0.870
Pos + Neg	TransE	0.022	0.161	94.809	0.576
	TransR	0.100	0.274	60.120	0.732
	TransH	0.025	0.174	91.553	0.594
	ComplEx	0.132	0.334	45.268	0.805
	distMult	0.149	0.378	35.351	0.853
	DeepWalk	0.148	0.383	35.365	0.849
	node2vec	**0.166**	0.389	34.305	0.840
	metapath2vec	0.020	0.165	93.374	0.578
	OWL2Vec*	0.160	0.397	32.234	0.869
	RDF2Vec	0.155	0.401	30.281	0.879
TrueWalks		0.161	0.392	32.089	0.869
TrueWalksOA		**0.166**	**0.407**	**28.128**	**0.889**

results in embeddings that are more meaningful semantic representations of proteins. Furthermore, the cosine similarity for negative pairs is consistently lower when using both variants of TrueWalks, which supports that the contribution of negative statement-based embeddings is working towards filtering out false positives.

5 Conclusion

Knowledge graph embeddings are increasingly used in biomedical applications such as the prediction of protein–protein interactions, gene-disease associations, drug-target relations and drug-drug interactions [28]. Our novel approach, TrueWalks, was motivated by the fact that existing knowledge graph embedding methods are ill-equipped to handle negative statements, despite their recognized importance in biomedical machine learning tasks [21]. TrueWalks incorporates a novel walk-generation method that distinguishes between positive and negative statements and considers the semantic implications of negation in ontology-rich knowledge graphs. It generates two separate embeddings, one for each type of statement, enabling a dual representation of entities that can be explored by downstream ML, focusing both on features entities have and those they lack. TrueWalks outperforms representative and state-of-the-art knowledge

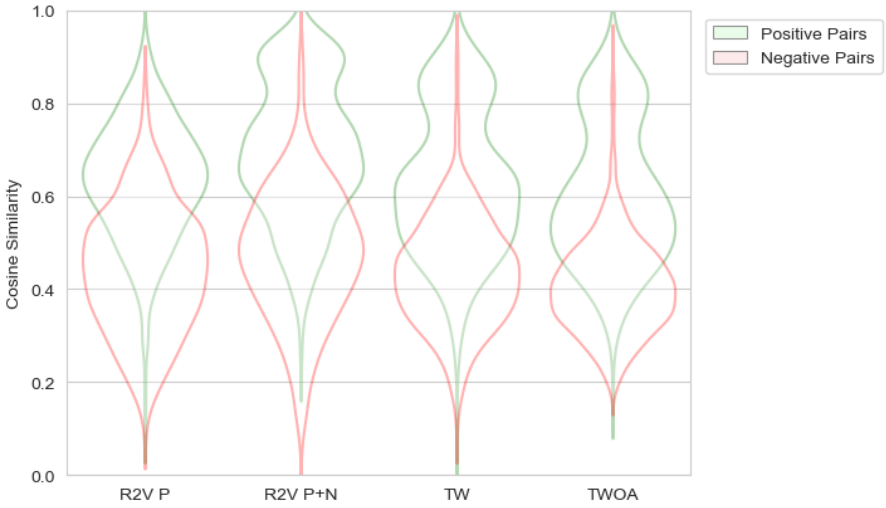


Fig. 4. Violin plot with embedding similarity obtained with RDF2Vec with positive statements (R2V P), RDF2Vec with both positive and negative statements (R2V P+N), TrueWalks (TW), and TrueWalksOA (TWOA).

graph embedding approaches in the prediction of protein-protein interactions and gene-disease associations.

We expect TrueWalks to be generalizable to other biomedical applications where negative statements play a decisive role, such as predicting disease-related phenotypes [47] or performing differential diagnosis [19]. In future work, we would also like to explore counter-fitting approaches, such as those proposed for language embeddings [29], to consider how opposite statements can impact the dissimilarity of entities.

Supplemental Material Statement: The source code for True Walks is available on GitHub (`https://github.com/liseda-lab/TrueWalks`). All datasets are available on Zenodo (`https://doi.org/10.5281/zenodo.7709195`). A supplementary file contains the links to the data sources, the parameters for the KG embedding methods and ML models, and the results of the statistical tests.

Acknowledgements C.P., S.S., and R.T.S. are funded by FCT, Portugal, through LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020). R.T.S. acknowledges the FCT PhD grant (ref. SFRH/BD/145377/2019). This work was also partially supported by the KATY project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017453, and in part by projeto 41, HfPT: Health from Portugal, funded by the Portuguese Plano de Recuperação e Resiliência. The authors are grateful to Lina Aveiro and Carlota Cardoso for the fruitful discussions that inspired this work.

References

1. Alshahrani, M., Khan, M.A., Maddouri, O., Kinjo, A.R., Queralt-Rosinach, N., Hoehndorf, R.: Neuro-symbolic representation learning on biological knowledge graphs. Bioinformatics **33**(17), 2723–2730 (2017)
2. Arnaout, H., Razniewski, S., Weikum, G., Pan, J.Z.: Negative statements considered useful. Journal of Web Semantics **71**, 100661 (2021), publisher: Elsevier
3. Arnaout, H., Razniewski, S., Weikum, G., Pan, J.Z.: Wikinegata: a knowledge base with interesting negative statements. Proceedings of the VLDB Endowment **14**(12), 2807–2810 (2021), publisher: VLDB Endowment Inc.
4. Asif, M., Martiniano, H., Couto, F.: Identifying disease genes using machine learning and gene functional similarities, assessed through Gene Ontology. PLOS ONE **13**, e0208626 (12 2018)
5. Boettiger, C.: rdflib: A high level wrapper around the redland package for common rdf applications (2018)
6. Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Proceedings of NIPS 2013. p. 2787–2795. Curran Associates Inc., Red Hook, NY, USA (2013)
7. Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O.M., Antonyrajah, D., Horrocks, I.: OWL2Vec*: Embedding of OWL ontologies. Machine Learning pp. 1–33 (2021)
8. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 135–144 (2017)
9. Eilbeck, K., Quinlan, A., Yandell, M.: Settling the score: variant prioritization and mendelian disease. Nature Reviews Genetics **18**(10), 599–612 (2017)
10. Flouris, G., Huang, Z., Pan, J.Z., Plexousakis, D., Wache, H.: Inconsistencies, negations and changes in ontologies. In: Proceedings of the 21st National Conference on Artificial Intelligence-Volume 2. pp. 1295–1300 (2006)
11. Fu, G., Wang, J., Yang, B., Yu, G.: NegGOA: negative GO annotations selection using ontology structure. Bioinformatics **32**(19), 2996–3004 (06 2016)
12. Gaudet, P., Dessimoz, C.: Gene Ontology: pitfalls, biases, and remedies. In: The Gene Ontology Handbook, pp. 189–205. Humana Press, New York, NY (2017)
13. GO Consortium: The Gene Ontology resource: enriching a GOld mine. Nucleic Acids Research **49**(D1), D325–D334 (2021)
14. GO Consortium: The Gene Ontology Resource: 20 years and still GOing strong. Nucleic Acids Research **47**(D1), D330–D338 (11 2018)
15. Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., Sattler, U.: OWL 2: The next step for OWL. Journal of Web Semantics **6**(4), 309–322 (2008)
16. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 855–864 (2016)
17. Hogan, A., Blomqvist, E., Cochez, M., d'Amato, C., Melo, G.d., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., et al.: Knowledge graphs. ACM Computing Surveys (CSUR) **54**(4), 1–37 (2021)
18. Hu, L., Wang, X., Huang, Y.A., Hu, P., You, Z.H.: A survey on computational models for predicting protein–protein interactions. Briefings in Bioinformatics **22**(5), bbab036 (2021)
19. Köhler, S., Øien, N.C., Buske, O.J., Groza, T., Jacobsen, J.O., McNamara, C., Vasilevsky, N., Carmody, L.C., Gourdine, J., Gargano, M., et al.: Encoding clinical data with the Human Phenotype Ontology for computational differential diagnostics. Current Protocols in Human Genetics **103**(1), e92 (2019)

20. Kulmanov, M., Liu-Wei, W., Yan, Y., Hoehndorf, R.: EL embeddings: geometric construction of models for the description logic EL++. Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (2019)
21. Kulmanov, M., Smaili, F.Z., Gao, X., Hoehndorf, R.: Semantic similarity and machine learning with ontologies. Briefings in Bioinformatics **22**(4), bbaa199 (2021)
22. Köhler, S., Gargano, M., Matentzoglu, N., Carmody, L.C., Lewis-Smith, D., Vasilevsky, N.A., Danis, D.e.a.: The Human Phenotype Ontology in 2021. Nucleic Acids Research **49**(D1), D1207–D1217 (12 2020)
23. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. Proceedings of the AAAI conference on artificial intelligence **29**(1) (2015)
24. Ling, W., Dyer, C., Black, A.W., Trancoso, I.: Two/too simple adaptations of word2vec for syntax problems. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1299–1304 (2015)
25. Liu, L., Zhu, S.: Computational methods for prediction of human protein-phenotype associations: A review. Phenomics **1**(4), 171–185 (2021)
26. Luo, P., Xiao, Q., Wei, P.J., Liao, B., Wu, F.X.: Identifying disease-gene associations with graph-regularized manifold learning. Frontiers in Genetics **10** (2019)
27. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
28. Mohamed, S.K., Nounu, A., Nováček, V.: Biological applications of knowledge graph embedding models. Briefings in Bioinformatics **22**(2), 1679–1693 (2021)
29. Mrksic, N., Séaghdha, D.Ó., Thomson, B., Gasic, M., Rojas-Barahona, L.M., Su, P.H., Vandyke, D., Wen, T.H., Young, S.J.: Counter-fitting word vectors to linguistic constraints. In: HLT-NAACL (2016)
30. Nunes, S., Sousa, R.T., Pesquita, C.: Predicting gene-disease associations with knowledge graph embeddings over multiple ontologies. In: ISMB Annual Meeting - Bio-Ontologies (2021)
31. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 701–710 (2014)
32. Pesquita, C., Faria, D., Falcao, A.O., Lord, P., Couto, F.M.: Semantic similarity in biomedical ontologies. PLoS Computational Biology **5**(7), e1000443 (2009)
33. Piñero, J., Ramírez-Anguita, J.M., Saüch-Pitarch, J., Ronzano, F., Centeno, E., Sanz, F., Furlong, L.I.: The DisGeNET knowledge platform for disease genomics: 2019 update. Nucleic Acids Research **48**(D1), D845–D855 (11 2019)
34. Portisch, J., Paulheim, H.: Putting RDF2Vec in order. In: CEUR Workshop Proceedings. vol. 2980, pp. 1–5. RWTH (2021)
35. Ristoski, P., Paulheim, H.: RDF2Vec: RDF graph embeddings for data mining. In: International Semantic Web Conference. pp. 498–514. Springer (2016)
36. Robinson, P., Köhler, S., Oellrich, A., Genetics, S., Wang, K., Mungall, C., Lewis, S., Washington, N., Bauer, S., Seelow, D., Krawitz, P., Gilissen, C., Haendel, M., Smedley, D.: Improved exome prioritization of disease genes through cross-species phenotype comparison. PCR Methods and Applications **24**(2), 340–348 (Feb 2014)
37. Smaili, F.Z., Gao, X., Hoehndorf, R.: OPA2Vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. Bioinformatics **35**(12), 2133–2140 (2019)
38. Sousa, R.T., Silva, S., Pesquita, C.: Evolving knowledge graph similarity for supervised learning in complex biomedical domains. BMC Bioinformatics **21**(1), 1–19 (2020)

39. Sousa, R.T., Silva, S., Pesquita, C.: Benchmark datasets for biomedical knowledge graphs with negative statements (2023)
40. Szklarczyk, D., Gable, A.L., Nastou, K.C., Lyon, D., Kirsch, R., Pyysalo, S., Doncheva, N.T., Legeay, M., Fang, T., Bork, P., Jensen, L.J., von Mering, C.: The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. Nucleic Acids Research **49**(D1), D605–D612 (11 2020)
41. Vanunu, O., Magger, O., Ruppin, E., Shlomi, T., Sharan, R.: Associating genes and protein complexes with disease via network propagation. PLoS Computational Biology **6** (2010)
42. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering **29**(12), 2724–2743 (2017)
43. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge Graph Embedding by Translating on Hyperplanes. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence. pp. 1112–1119. AAAI Press (2014)
44. Warwick Vesztrocy, A., Dessimoz, C.: Benchmarking Gene Ontology function predictions using negative annotations. Bioinformatics **36**(Supplement_1), i210–i218 (07 2020)
45. Xiong, B., Potyka, N., Tran, T.K., Nayyeri, M., Staab, S.: Faithful Embeddings for EL++ Knowledge Bases. In: International Semantic Web Conference. pp. 22–38. Springer (2022)
46. Xu, Q.S., Liang, Y.Z.: Monte Carlo cross validation. Chemometrics and Intelligent Laboratory Systems **56**(1), 1–11 (2001)
47. Xue, H., Peng, J., Shang, X.: Predicting disease-related phenotypes using an integrated phenotype similarity measurement based on HPO. BMC Systems Biology **13**(2), 1–12 (2019)
48. Yang, B., tau Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases (2015)
49. Zakeri, P., Simm, J., Arany, A., ElShal, S., Moreau, Y.: Gene prioritization using bayesian matrix factorization with genomic and phenotypic side information. Bioinformatics **34**, i447 – i456 (2018)
50. Zhang, S.B., Tang, Q.R.: Protein–protein interaction inference based on semantic similarity of Gene Ontology terms. Journal of Theoretical Biology **401**, 30–37 (2016)
51. Zhu, Y., Liu, H., Wu, Z., Song, Y., Zhang, T.: Representation learning with ordered relation paths for knowledge graph completion. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 2662–2671 (2019)

Appendix D

Benchmark datasets for biomedical knowledge graphs with negative statements

Benchmark datasets for biomedical knowledge graphs with negative statements

Rita T. Sousa, Sara Silva, and Catia Pesquita

LASIGE, Faculdade de Ciências da Universidade de Lisboa
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.pt

Abstract. Knowledge graphs represent facts about real-world entities. Most of these facts are defined as positive statements. The negative statements are scarce but highly relevant under the open-world assumption. Furthermore, they have been demonstrated to improve the performance of several applications, namely in the biomedical domain. However, no benchmark dataset supports the evaluation of the methods that consider these negative statements.
We present a collection of datasets for three relation prediction tasks - protein-protein interaction prediction, gene-disease association prediction and disease prediction - that aim at circumventing the difficulties in building benchmarks for knowledge graphs with negative statements. These datasets include data from two successful biomedical ontologies, Gene Ontology and Human Phenotype Ontology, enriched with negative statements.
We also generate knowledge graph embeddings for each dataset with two popular path-based methods and evaluate the performance in each task. The results show that the negative statements can improve the performance of knowledge graph embeddings.

Keywords: Biomedical Knowledge Graphs · Biomedical Ontologies · Gene Ontology · Human Phenotype Ontology · Negative Statements · Protein-Protein Interaction Prediction · Gene-Disease Association Prediction · Disease Prediction

1 Introduction

Knowledge Graphs (KGs) have been used to represent knowledge about real-world entities and their relationships. Most KGs use ontologies as a backbone to describe entities through ontology-based annotation, which associates an entity with a class. These annotations are commonly represented as positive statements establishing that an ontology class describes an entity. For example, in the biomedical domain, positive statements express that a protein $P1$ performs *intracellular_galactose_homeostasis* as defined in the Gene Ontology (GO) [6]. Negative statements are extremely rare but can be used to declare that a given protein $P2$ does not perform *glucose_homeostasis* (Figure 1).

The lack of negative statements is a significant issue because KGs operate under the open-world assumption. Therefore, this lack of information can lead

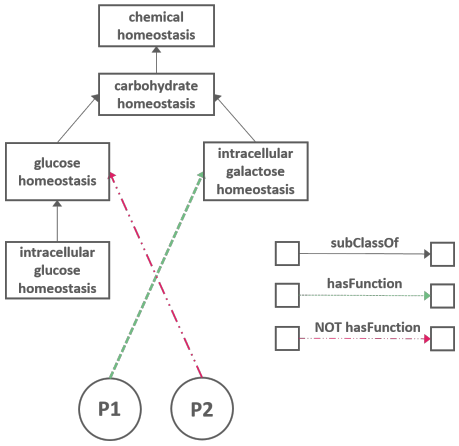


Fig. 1: A GO KG subgraph with positive and negative statements describing two proteins.

to confusion regarding whether the absence of a positive statement is due to a lack of knowledge or the actual absence of the relationship. Moreover, the importance of negative statements to produce more accurate representations of entities in a KG [8, 4] and improving performance in different applications [3, 15] is increasingly recognized in the biomedical domain.

While there have been attempts to enhance current KGs with interesting negative statements, to the best of our knowledge, no benchmark datasets have been established to evaluate learning tasks over those KGs. With this in mind, we enrich existing biomedical KGs with negative statements and propose a collection of datasets for different biomedical tasks of relation prediction. The biomedical domain was selected because biomedical KGs are usually back-boned by biomedical ontologies that can express negation. Additionally, negative statements have been considered relevant for different biomedical applications [7]. Our datasets are grouped according to the task: protein-protein interaction (PPI) prediction, gene-disease association (GDA) prediction and disease prediction. Regarding the KGs, we enrich two successful biomedical ontologies: GO which covers distinct semantic aspects of gene products' function, and Human Phenotype Ontology (HP) which describes the universe of concepts related to phenotypic abnormalities found in human hereditary diseases.

2 Related Work

Several approaches to enriching existing KGs with interesting negative statements have been proposed. Arnaout *et al.* [1] proposed a method to enrich Wikidata by including interesting negative statements, which led to improvements in tasks involving entity summarization and decision-making.

In the biomedical domain, several approaches tackle the lack of negative statements in biomedical ontologies, such as GO. The number of functions that a protein does not have is larger than the number of functions it has. Therefore, the number of negative statements describing proteins in the GO should be several orders of magnitude greater than the number of positive statements. Youngs *et al.* [17] designed two algorithms to predict negative statements for GO and populate the NoGo database, one based on empirical conditional probability and the other on topic modeling applied to genes and annotation. Fu *et al.* [3] introduced NegGOA, a new method to enrich the GO with relevant negative statements indicating that a protein does not perform a given function. This method exploits the GO by using hierarchical semantic similarity between GO terms. The enriched GO was used for protein function prediction. Later, Vesztrocy *et al.* [15] presented a benchmark based on a balanced test set of positive and negative statements. The negative statements are generated from expert-curated annotations of protein families on phylogenetic trees. The results of this work demonstrated that negative statements improve protein function prediction. Regarding the HP, although the importance of negative statements in gene-phenotype prediction is recognized, the enrichment with negative statements has yet to be investigated [8].

3 Building the Datasets

We present a collection of datasets that work over two enriched KGs for three relation prediction tasks: PPI prediction, GDA prediction, and disease prediction. Each benchmark dataset comprises several pairs of biomedical entities (or instances) that can be of the same type (protein-protein) or distinct types (gene-disease and disease-patient) with the respective label (1 for the positive pairs and zero for the negative pairs). Tables 1 and 2 show the KGs' and datasets' statistics for each task. Since for GDA prediction and disease prediction, the target relation happens between two types of instances (genes and diseases for GDA prediction and diseases and patients for disease prediction), the instance numbers in Table 2 appear separately. Moreover, in the case of PPI prediction, we exclusively employ the GO KG that has been subjected to a negative statement enrichment approach. However, when it comes to GDA prediction and disease prediction, we rely on the HP KG, which lacks a negative statement enrichment approach, resulting in a significant imbalance between the number of positive and negative statements.

To build these datasets, we adopt three main steps. The first one consists of enriching the KGs. The KG is constructed using the owlready2 package[1], which parses the ontology file in OWL format and processes the annotation file. The annotation file contains positive and negative statements used to describe entities. We use the guidelines established by the W3C[2] to define the negative statements

[1] https://owlready2.readthedocs.io/en/v0.37/
[2] https://www.w3.org/TR/owl2-mapping-to-rdf/

Table 1: Statistics for each ontology regarding classes, nodes, edges.

	GO	HP
Classes	50918	17060
Literals and blank nodes	532373	442246
Edges	1425102	1082859

Table 2: Statistics for each task's dataset regarding the number of instances, pairs, positive and negative statements.

	PPI prediction	GDA prediction	Disease Prediction
Instances	440	174 + 107	1033 + 660
Positive Pairs	1024	107	660
Negative Pairs	1024	107	681120
Positive statements	7364	14828	38130
Negative statements	8579	9191	179

```
<owl:NamedIndividual rdf:about="http://purl.obolibrary.org/obo/GO_0048268">
    <rdf:type rdf:resource="http://purl.obolibrary.org/obo/GO_0048268"/>
</owl:NamedIndividual>
<rdf:Description>
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#NegativePropertyAssertion"/>
    <owl:sourceIndividual rdf:resource="http://purl.obolibrary.org/obo/GO_0048268"/>
    <owl:assertionProperty rdf:resource="http://purl.obolibrary.org/obo#has_function"/>
    <owl:targetIndividual rdf:resource="https://www.uniprot.org/uniprotkb/Q9BY11"/>
</rdf:Description>
```

Fig. 2: Example of how the negative statements are defined in the OWL file.

as negative object property assertions[3]. To do so, we use metamodeling and represent each ontology class as a class and an individual. This situation translates into using the same IRI. Then, we use a negative object property assertion to state that the individual representing a biomedical entity is not connected by the object property expression to the individual representing an ontology class, as depicted in Figure 2. The second step consists of extracting pairs of entities from bioinformatic databases. The third step involves selecting the pairs containing KG entities that are well described with positive and negative statements.

The following subsections describe in more detail the KGs as the characteristics of each task.

3.1 Biomedical Knowledge Graphs

Two KGs back-boned by biomedical ontologies are used: the GO KG and the HP KG. Table 1 shows the statistics for each ontology.

[3] https://www.w3.org/TR/owl2-syntax/#Negative_Object_Property_Assertions

The GO is used to describe gene products (proteins or genes) according to the molecular functions they perform, the biological processes they are involved in, and the cellular components where they act. The GO KG is built by integrating three sources: the GO[4] itself, the GO Annotation data[5] [5], and negative GO associations produced in [15][6].

A GO annotation links a specific gene product with a particular GO class. The majority of GO annotation data corresponds to positive statements. However, the GO annotation has the qualifier 'NOT' for a few cases, meaning that a gene product has been proven not to carry out a specific function. The annotations that possess this qualifier were added as negative statements. In addition to these negative statements, the GO KG was also enriched with negative statements derived from expert-curated annotations of protein families on phylogenetic trees. The idea is that, if no evidence exists to suggest otherwise, gene function is maintained over time through evolution. Therefore, after expert curators have annotated ancestral states in gene phylogenies with GO classes, they check if the annotations are propagated down the phylogeny. When there is evidence that the function is absent in a specific sub-tree, a negative statement is added to that protein. These enriched negative statements were filtered so there were no contradictions with the GO annotation data.

HP characterizes phenotypic abnormalities discovered in human hereditary diseases according to five semantic aspects: phenotypic abnormalities, mode of inheritance, clinical course, clinical modifier and frequency. HP annotations can link diseases, patients or genes to HP classes via positive and negative statements. The construction of HP KG[7] is similar to that of the GO KG. A negative annotation from HP that includes 'NOT' indicates that a disease does not cause that phenotype, so they are included as negative statements.

3.2 Protein-Protein Interaction Prediction Dataset

Predicting PPIs is a fundamental task in molecular biology for understanding biological systems. Given the high cost of experimentally determining PPI, many computational approaches for PPI prediction based on available functional information described by the GO [6] have been proposed to find protein pairs likely to interact and thus provide a selection of good candidates for experimental analysis. Therefore, the GO KG is used to describe the proteins of the dataset.

The positive examples are extracted from the STRING [13] database. Our selection of protein pairs was based on the following criteria: (i) interactions between proteins had to be curated or experimentally determined rather than

[4] The GO was downloaded on September 2021. It is available at http://release.geneontology.org/2021-09-01/ontology/index.html

[5] The GO positive annotations were downloaded on January 2021. It is available at http://release.geneontology.org/2021-01-01/annotations/index.html.

[6] The negative annotations were downloaded from https://lab.dessimoz.org/20_not

[7] The HP was downloaded on October 2022, while the HP annotations were downloaded on November 2021. A link to these versions is no longer available.

computationally determined; (ii) interactions needed to have a confidence score above 0.950 to ensure high confidence; (iii) each protein must have at least one positive statement for a GO class and one negative statement for another GO class. The negative examples are generated by random negative sampling over the set of proteins of the positive examples.

3.3 Gene-Disease Association Prediction Dataset

Knowing which genes are associated with a specific disease is crucial to understanding the disease mechanisms and recognising potential biomarkers or therapeutic targets. However, once again, validating these associations in the wet lab is expensive and time-consuming. This has prompted the evolution of computational methods to identify the most promising associations to be further validated.

The two KGs are used for the GDA prediction task dataset. GO KG describes the genes, and HP KG describes the diseases. The target relations to predict are extracted from DisGeNET [11]. Adapting the approach described in [10], we considered the following criteria to select gene-disease pairs: (i) each gene must have at least one positive statement for a GO class and one negative statement for another GO class; (ii) each disease must have at least one positive statement for an HP class and one negative statement for an HP class. We sampled random negative examples of the same genes and diseases to create a balanced dataset.

3.4 Disease Prediction Datasets

Since human diseases are a complex phenomenon, disease prediction is an essential but still complicated task that must be executed accurately and efficiently. Therefore, using computational methods to help physicians prioritize diseases is highly advantageous.

The dataset to predict if a synthetic patient has been diagnosed with a specific disease is generated by adapting the methodology proposed in [9]. Thirty-three mendelian diseases for which they knew the penetrance of each phenotype are selected. Penetrance indicates the likelihood that a patient suffering from a specific disease will exhibit a particular phenotype. For each of these 33 diseases, 20 synthetic patients diagnosed with that disease are created. The patients' positive annotation is determined by the disease's penetrance and the patient's gender. The gender is defined randomly with an equal likelihood for both genders. For example, the 'Aarskog-Scott syndrome' is annotated with the phenotype 'Ptosis' with a penetrance of 0.5061, meaning that approximately half of the synthetic patients diagnosed with that disease will have a positive statement for this phenotype. The negation of phenotypes does not have a penetrance associated, so synthetic patients inherit the negative phenotypes related to the disease. For example, since the disease 'Aarskog-Scott syndrome' is annotated with 'NOT Decreased Fertility', each patient will have a negative statement for this phenotype. Furthermore, 1000 diseases were randomly chosen to add complexity to the task. These diseases are annotated with positive and negative statements.

Table 3: Statistics for each noise version regarding the number of positive and negative statements.

Noise	Positive Statements	Negative Statements
0.1	40592	216
0.2	37195	214
0.4	38242	192

Random annotations can also be added to patients to emulate a more realistic situation where a patient is associated with phenotypes unrelated to the patient's disease. In addition to the disease prediction dataset, we present three versions with random annotations. The number of random annotations is defined by a percentage Noi (Noi=[0, 0.1,0.2,0.4]) concerning a given patient's total number of annotations. For example, if Noi=0.5, half of the full annotations of a given patient are added. Table 3 shows the number of positive and negative statements for each noise version.

4 Validation of the Datasets

KG embedding methods [14] have been successfully employed in several biomedical applications [14]. Since these methods map KGs into low-dimensional spaces, they have emerged as a popular way to generate features for machine learning tasks. Therefore, we use two KG embedding methods to evaluate our datasets - RDF2Vec [12] and OWL2Vec* [2]. RDF2Vec is a path-based method that generates random walks in the KG that constitutes the corpus of word sequences given as input to a neural language model. OWL2Vec* was designed to learn ontology embeddings and it also employs direct walks on the graph to learn graph structure. These embedding methods generate representations of the biomedical entities that are combined using the binary Hadamard operator to represent the pair.

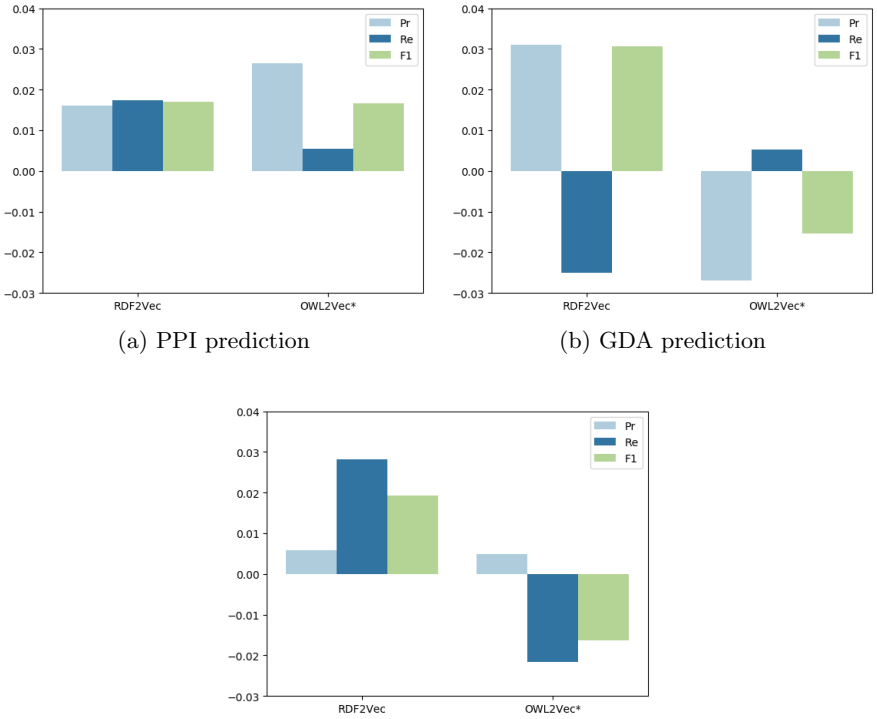
The pair representations are then fed into a Random Forest algorithm for training using Monte Carlo cross-validation (MCCV) [16]. MCCV is a variation of traditional k-fold cross-validation in which the data is divided into training and testing sets (with β being the proportion of the dataset to include in the test split) M times. Our experiments use MCCV with $M = 30$ and $\beta = 0.3$ for PPI and GDA prediction. Given the large number of pairs for disease prediction, we use MCCV with $M = 5$ and $\beta = 0.3$.

Each embedding method is run with two different KGs, one with only positive statements and the other with both positive and negative statements. Table 4 reports each task's median of recall, precision and weighted average F-measure.

Figure 3 compares the impact of using only positive statements versus both positive and negative statements on our datasets. The bars represent the difference in performance for precision, recall and weighted average F-measure, with

Table 4: Median precision, recall and weighted average F-measure (Pr, Re, and F1) for PPI, GDA, and disease prediction using only positive statements (Pos) or positive and negative statements (Pos+Neg).

Method	Statements	PPI			GDA			Disease		
		Pr	Re	F1	Pr	Re	F1	Pr	Re	F1
RDF2Vec	Pos	0.831	0.826	0.828	0.623	0.625	0.615	0.994	0.742	0.850
	Pos+Neg	0.847	0.844	0.845	0.654	0.600	0.645	1.000	0.771	0.870
OWL2Vec*	Pos	0.833	0.806	0.823	0.652	0.656	0.646	0.975	0.584	0.730
	Pos+Neg	0.860	0.812	0.840	0.625	0.661	0.630	0.980	0.563	0.713



(a) PPI prediction

(b) GDA prediction

(c) Disease prediction

Fig. 3: Barplots showing the differences in precision, recall, weighted average F-measure (Pr, Re, and F1) between using positive and negative statements or only positive statements.

upward bars indicating improved performance with both positive and negative statements and downward bars indicating decreased performance.

The experiments show that the added information given by negative statements generally improves the performance of RDF2Vec. However, for OWL2Vec*, the performance only improves for PPI prediction.

5 Using the Benchmark

All datasets are available on Zenodo[8] under a CC BY 4.0 license. For each dataset, we provide access to two types of files: (1) one TSV file containing pairs of entities and information about whether a relationship exists between them or not; (2) OWL files containing the KG used to describe the biomedical entities that appear in the TSV file. Together, these files can be used to perform relation prediction tasks since the TSV file provides the specific entities and relations that need to be predicted, while the OWL file provides the necessary background knowledge for generating the features.

6 Conclusions

Benchmark datasets are essential for evaluating and comparing the performance of different approaches that work over KGs. This paper presents a collection of datasets for three relation prediction tasks in the biomedical domain: PPI prediction, GDA prediction, and disease prediction. The biomedical domain is chosen since it is already demonstrated that the inadequacy of approaches to take into consideration negative statements is a limitation for several biomedical applications. However, although the datasets are domain-specific, they can be used to evaluate approaches outside the biomedical domain.

The datasets are validated using two popular KG embedding methods to generate features that are then given as input for a classifier. The results highlight the importance of incorporating negative statements into KGs to create more accurate representations of KG entities.

Acknowledgements C. P., S. S., R. T. S. are funded by the FCT through LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020), and the FCT PhD grant (ref. SFRH/BD/145377/2019). It was also partially supported by the KATY project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017453, and by HfPT: Health from Portugal under the Portuguese Plano de Recuperação e Resiliência. The authors thank Lina Aveiro for the preliminary results of this work.

References

1. Arnaout, H., Razniewski, S., Weikum, G., Pan, J.Z.: Negative statements considered useful. Journal of Web Semantics **71**, 100661 (2021), publisher: Elsevier

[8] https://doi.org/10.5281/zenodo.7709195

2. Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O.M., Antonyrajah, D., Horrocks, I.: OWL2Vec*: Embedding of OWL ontologies. Machine Learning pp. 1–33 (2021)
3. Fu, G., Wang, J., Yang, B., Yu, G.: NegGOA: negative GO annotations selection using ontology structure. Bioinformatics **32**(19), 2996–3004 (06 2016)
4. Gaudet, P., Dessimoz, C.: Gene Ontology: pitfalls, biases, and remedies. In: The Gene Ontology Handbook, pp. 189–205. Humana Press, New York, NY (2017)
5. GO Consortium: The Gene Ontology resource: enriching a GOld mine. Nucleic Acids Research **49**(D1), D325–D334 (2021)
6. GO Consortium: The Gene Ontology Resource: 20 years and still GOing strong. Nucleic Acids Research **47**(D1), D330–D338 (11 2018)
7. Kulmanov, M., Smaili, F.Z., Gao, X., Hoehndorf, R.: Semantic similarity and machine learning with ontologies. Briefings in Bioinformatics **22**(4), bbaa199 (2021)
8. Liu, L., Zhu, S.: Computational methods for prediction of human protein-phenotype associations: A review. Phenomics **1**(4), 171–185 (2021)
9. Masino, A.J., Dechene, E.T., Dulik, M.C., Wilkens, A., Spinner, N.B., Krantz, I.D., Pennington, J.W., Robinson, P.N., White, P.S.: Clinical phenotype-based gene prioritization: an initial study using semantic similarity and the human phenotype ontology. BMC bioinformatics **15**(1), 1–11 (2014)
10. Nunes, S., Sousa, R.T., Pesquita, C.: Predicting gene-disease associations with knowledge graph embeddings over multiple ontologies. In: ISMB Annual Meeting - Bio-Ontologies (2021)
11. Piñero, J., Ramírez-Anguita, J.M., Saüch-Pitarch, J., Ronzano, F., Centeno, E., Sanz, F., Furlong, L.I.: The DisGeNET knowledge platform for disease genomics: 2019 update. Nucleic Acids Research **48**(D1), D845–D855 (11 2019)
12. Ristoski, P., Paulheim, H.: RDF2Vec: RDF graph embeddings for data mining. In: International Semantic Web Conference. pp. 498–514. Springer (2016)
13. Szklarczyk, D., Gable, A.L., Nastou, K.C., Lyon, D., Kirsch, R., Pyysalo, S., Doncheva, N.T., Legeay, M., Fang, T., Bork, P., Jensen, L.J., von Mering, C.: The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. Nucleic Acids Research **49**(D1), D605–D612 (11 2020)
14. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE Transactions on Knowledge and Data Engineering **29**(12), 2724–2743 (2017)
15. Warwick Vesztrocy, A., Dessimoz, C.: Benchmarking Gene Ontology function predictions using negative annotations. Bioinformatics **36**(Supplement_1), i210–i218 (07 2020)
16. Xu, Q.S., Liang, Y.Z.: Monte Carlo cross validation. Chemometrics and Intelligent Laboratory Systems **56**(1), 1–11 (2001)
17. Youngs, N., Penfold-Brown, D., Bonneau, R., Shasha, D.: Negative example selection for protein function prediction: The NoGO database. PLOS Computational Biology **10**(6), 1–12 (06 2014)

Appendix E

Explaining Protein-Protein Interaction Predictions with Genetic Programming

Explaining Protein-Protein Interaction Predictions with Genetic Programming

Rita T. Sousa, Sara Silva, and Catia Pesquita

LASIGE, Faculdade de Ciências da Universidade de Lisboa
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.pt

Abstract. Explainability is crucial to support the adoption of machine learning as a tool for scientific discovery. In the biomedical domain, ontologies and knowledge graphs are a unique opportunity to explore domain knowledge, but most knowledge graph-based approaches employ graph embeddings, which are not explainable. However, when the prediction target is finding a relation between two entities represented in the graph, such as in the case of protein-protein interaction prediction, semantic similarity presents itself as a natural explanatory mechanism. This work uses genetic programming over a set of semantic similarity values, each describing a semantic aspect represented in the knowledge graph, to generate global and interpretable explanations for protein-protein interaction prediction. Our experiments reveal that genetic programming algorithms coupled with semantic similarity produce global models relevant to understanding the biological phenomena.

Keywords: Explainable Artificial Intelligence· Knowledge Graph · Genetic Programming · Protein-Protein Interaction Prediction.

1 Introduction

In artificial intelligence (AI) applications in science, explanations are crucial not only for the user's trust but also for discovering of new knowledge. Several explainable artificial intelligence (XAI) approaches have been proposed, but only a few approaches integrate domain knowledge modelled through semantic technologies such as ontologies and knowledge graphs (KGs) [1]. However, most KG-based machine learning (ML) approaches apply KG embedding methods which are sub-symbolic representations of KG entities that are not interpretable by default [2].

Since similarity assessment is a natural explanatory mechanism [8], an alternative explanatory strategy is to use the ontologies and KGs to measure the semantic similarity (SS) between entities in the graph. This is particularly relevant in the biomedical domain where ontologies allow the description of complex biological phenomena, providing the scaffolding for comparing biological entities through their ontology representations. Furthermore, since SS can be computed using different portions of the KG to reflect different semantic aspects (SA) [5], we propose that SS can provide more granular explanations with higher information content.

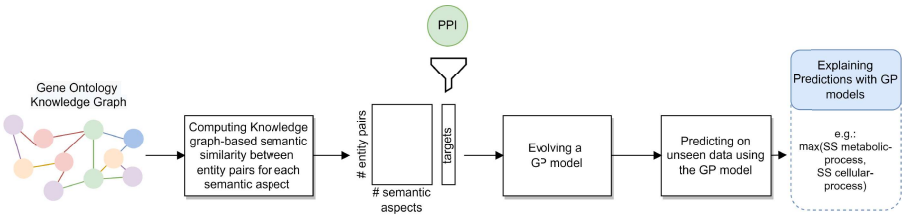


Fig. 1. Overview of our methodology.

We address the explainability problem for protein-protein interaction (PPI) prediction by using genetic programming (GP) algorithms [3] over ontology-based SS values that capture different SAs. GP algorithms were chosen given their ability to produce potentially interpretable models that provide a global explanation of how the model works, unlike many classical ML algorithms and deep learning methods.

2 Methods

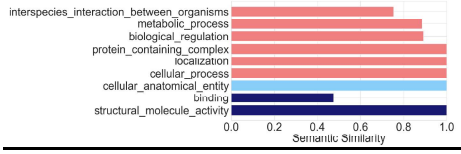
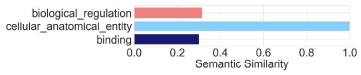
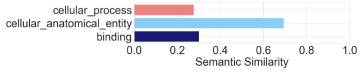
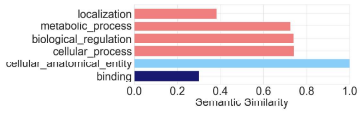
PPI prediction is cast as a classification task that takes as input a KG and a PPI dataset containing a set of protein pairs that interact (Figure 1). The PPI dataset was obtained from the STRING Database[1]. We used the Gene Ontology (GO) [7] KG composed by the ontology and annotations that link proteins to GO functions. The functions in GO are described concerning three domains: biological processes, molecular functions and cellular components. The three GO domains are represented as root classes. The child classes of these roots, such as *'cellular_proccess'* and *'catalytic_activity'*, represent an ontology SA.

We followed the methodology of [6]. First we computed SS scores for the protein pairs according to different ontology SAs corresponding to the 50 subgraphs rooted in the direct subclasses of GO roots (removing aspects with potential bias for PPI, namely *'binding'* and *'protein-containing_complex'*). We used the $ResnikMax_{Seco}$ similarity measure implemented in [6]. Then, we evolved a GP model to predict PPI for protein pairs represented by their 50 SS scores.

Although GP searches the space using genetic operators that manipulate their syntactic representation fulfilling every constraint for transparency, sometimes the solutions grow exponentially with each generation, and the interpretability is lost. To tackle this, we modified the fitness function of standard GP to penalize solutions with a depth greater than six (value given as a parameter), thus lowering the probability of deep trees. In addition, this variation of GP (GP6x), only uses interpretable operators, namely maximum, minimum, addition and subtraction. Operators such as multiplication and division were excluded as it is difficult to interpret the biological meaning of multiplication/division between SS values. We performed 10-fold cross-validation and compared GP (no depth penalization and 6 operators) with GP6x.

[1] https://string-db.org

Table 1. Example of an explainable GP6x model and description of protein pairs SS with supporting evidence for interaction status (dark blue: molecular function, light blue: cellular component, pink: biological process).

Explainable Model: $\max(SS_{\text{multicellular\_organismal\_process}}, SS_{\text{cellular\_process}}, SS_{\text{molecular\_adaptor\_activity}}, SS_{\text{signaling}}, SS_{\text{molecular\_function\_regulator}}, SS_{\text{catalytic\_activity}}, SS_{\text{behavior}} + SS_{\text{immune\_system\_process}})$	
40S ribosomal protein S12 – 40S ribosomal protein S10 (+/+)	
	40S ribosomal protein S12 and 40S ribosomal protein S10 are components of the 40S ribosomal subunit that plays a central role in protein translation and is characterized by multiple binding sites.
S100-A10 protein – neuroblast differentiation-associated protein (+/−)	
	Protein S100-A10 works together with neuroblast differentiation-associated protein AHNAK in the development of the intracellular membrane.
Kinetochore-associated protein 1 – Tubulin beta-6 chain (−/−)	
	Kinetochore-associated protein 1 and tubulin beta-6 chain are both located in the nucleus but have different functions: while the first one is envolved in mytosis, the late is envolved in GTP binding.
Protransforming growth factor α – Disks large homolog 2 (−/+)	
	TGF-α is a mitogenic polypeptide, and disks large homolog 2 is a member of the membrane-associated guanylate kinase. Both participate in MAPK cascade.

3 Results

The median weighted average of F-measures (WAF) for GP is 0.875 while for GP6x it is 0.866 (p-value of 0.0065 with Kruskal-Wallis test). As to the number of nodes in the unsimplified models, the medians are 49 for GP and 17 for GP6x (p-value of 0.0004). With small differences, all GP6x models consider maximum similarities of multiple SAs with a majority describing biological processes. This corroborates prior knowledge that for two proteins to interact they usually participate in the same biological processes.

To investigate the trade-off between performance and explainability we chose four protein pairs and analysed the input SS values and one of the GP6x models and its predictions, in Table 1. The protein pairs were randomly chosen among well-predicted positive pairs (+/+), well-predicted negative pairs (−/−), wrong-predicted positive pairs (+/−), and wrong-predicted negative pairs (−/+). One of the most interesting results is the analysis of the two pairs for which GP6x fails. *S100-A10* protein and the *neuroblast differentiation-associated* protein are

known to interact, and the biological processes where both proteins participate are described in the literature. However, according to GO annotations, proteins have the same location in the cell but do not share biological processes, resulting in low SS values for relevant SAs. The misclassification can then be justified by the incomplete annotation of these proteins. Concerning the pair *TGF-α - Disks large homolog 2*, GP6x predicts an interaction given the high similarity values for SAs relevant, but it appears as not interacting in the dataset. It is important to note that the negative PPI dataset examples were generated by negative random sampling. Interestingly, the literature describes interactions between proteins of the same family of the pair, which probably means that this pair is actually positive but not yet present in STRING.

4 Conclusion

A significant direction of XAI research is the discussion of trade-offs involving performance prediction and interpretability [4]. Our results show that the performance of the more interpretable methods is lower, but what they sacrifice in performance is gained in explainability. The analysis of the selected examples highlights how explainability can be key to uncover issues with the underlying data and even pose new hypothesis. One of the main advantages of transparent methods, such as GP, is that the explanation is the model itself, avoiding the need for local explanations or post-hoc techniques.

Acknowledgements This work was funded by FCT through LASIGE Research Unit (UIDB/00408/2020, UIDP/00408/2020); projects GADgET (DSAIPA/DS/0022/2018) and BINDER (PTDC/CCI-INF/29168/2017); PhD grant SFRH/BD/145377/2019.

References

1. Confalonieri, R., Weyde, T., Besold, T.R., Moscoso del Prado Martín, F.: Using ontologies to enhance human understandability of global post-hoc explanations of black-box models. Artificial Intelligence **296**, 103471 (2021)
2. Palmonari, M., Minervini, P.: Knowledge graph embeddings and explainable ai. Knowledge Graphs for Explainable Artificial Intelligence: Foundations, Applications and Challenges **47**, 49 (2020)
3. Poli, R., Langdon, W.B., McPhee, N.F., Koza, J.R.: A field guide to genetic programming. Freely available at `http://www.gp-field-guide.org.uk` (2008)
4. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature Machine Intelligence **1**(5), 206–215 (2019)
5. Sousa, R.T., Silva, S., Pesquita, C.: Evolving knowledge graph similarity for supervised learning in complex biomedical domains. BMC Bioinformatics **21**(1), 6 (2020)
6. Sousa, R.T., Silva, S., Pesquita, C.: evoKGsim+: a framework for tailoring knowledge graph-based similarity for supervised learning. In: ESWC 2021 Poster and Demo Track (2021)
7. The Gene Ontology Consortium: The gene ontology resource: 20 years and still GOing strong. Nucleic Acids Research **47**(D1), D330–D338 (2018)
8. Wang, D., Yang, Q., Abdul, A., Lim, B.Y.: Designing theory-driven user-centric explainable ai. In: Proc. of the 2019 CHI conference on human factors in computing systems. pp. 1–15 (2019)

Appendix F

Is there Data Leakage in Protein-Protein Interaction Prediction using Knowledge Graphs?

Is there Data Leakage in Protein-Protein Interaction Prediction using Knowledge Graphs?

Rita T. Sousa ✉, Sara Silva, Catia Pesquita

LASIGE, Faculdade de Ciências da Universidade de Lisboa, Portugal
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.pt

Abstract. There is a high potential for data leakage in biomedical machine learning applications since biomedical data resources share, reuse and import data from each other routinely. We have investigated potential data leakage in the prediction of protein-protein interactions using the Gene Ontology knowledge graph, by comparing the performance of models trained and tested on the same versions of data versus training on archived data and predicting only for newly discovered protein interactions. Our results were not able to detect an influence of data leakage, indicating that if this problem exists, its magnitude is not affecting the performance of knowledge graph-based protein interaction predictions.

1 Introduction

Machine learning methods have become a significant trend in several research fields in recent years, and the semantic web is no exception. As machine learning is increasingly being used, concerns about data leakage have been raised [1]. Leakage occurs when information about the target of a data mining problem that should not be legitimately available to mine from is introduced [3], and it can lead to overestimation of the model's performance.

In biomedical applications, such as protein-protein interaction (PPI) prediction, data leakage can also be an issue. It is not uncommon that multiple databases and resources reuse the same sources of information. The majority of PPI prediction methods that are based on knowledge graphs (KGs) [7,11] explore the Gene Ontology (GO) KG that defines the universe of classes associated with proteins functions. The GO KG, composed of the GO [9] and GO annotations [2] that link proteins to GO classes, is continuously evolving as more data become available [10]. The majority of GO annotations are inferred by electronic annotation (IEA), which means they are based on the automated processing of other data sources. This could result in the same information that is used to support a PPI in a database (e.g. STRING [8]) to also be used to establish a GO annotation for the proteins.

We hypothesize that if this type of data leakage is common, then the performance of GO-based PPI prediction methods would be artificially increased. To test this hypothesis, we compare PPI prediction models trained on older GO data and PPI interactions and tested on previously unknown interactions captured in more recent versions of STRING with same version training and testing. Furthermore, by training the models on labeled examples from the past, we more closely simulate real-world applications.

2 Methods

PPI prediction is cast as a classification task that takes as input the GO KG and a set of protein pairs. The first step of our approach is using historical data to build the PPI datasets. Then we use the GO KG and the protein pairs to predict interactions using several machine learning algorithms.

2.1 Data

The PPI datasets were obtained from the STRING Database[1] which is one of the largest available PPI databases that integrates both physical interactions as well as functional associations between proteins collected from several sources. We considered the following criteria to select protein pairs from STRING: (i) each protein must be annotated with the GO; (ii) protein interactions must be experimentally determined or from curated databases (as opposed to computationally determined); (iii) interactions must have a confidence score above 950 to retain only high confidence interactions. We employed random sampling to create negative pairs composed of the human proteins present in the positive pairs but without any STRING interactions between them, building a balanced dataset.

We built several PPI datasets using three archived versions of the STRING database (v9.1, v10, and v10.5) and the current version (v11). For the current version, we created three datasets each excluding protein pairs present in each of the older versions (see Table 1). Regarding the GO KG, we obtained archived versions of the GO and GO annotations in 2015, 2017 and 2019 from the Gene Ontology Data Archive[2].

2.2 Protein-Protein Interaction Prediction

We follow the setup in [7] that predicts relations between KG entity pairs that are not encoded in the graph using similarity-based semantic representations. We employed three KG-based semantic similarity measures to compute semantic similarity: two taxonomic measures (ResnikMax [5], SimGIC [4]) and one based on graph embedding methods (RDF2Vec [6]). We applied six well-known classes

[1] https://string-db.org
[2] http://release.geneontology.org/

STRING Version	Date	Number of positive pairs
v9.1	04/2015	12 681
v10	05/2017	26 863
v10.5	01/2019	31 384
v11 (excluding pairs in v9.1)	10/2020	41 227
v11 (excluding pairs in v10)	10/2020	31 642
v11 (excluding pairs in v10.5)	10/2020	23 571

Table 1. Number of positive pairs in each version of the STRING database.

of machine learning models to train classifiers using the scikit-learn library: K-nearest neighbor (KNN), genetic programming (GP), decision tree (DT), XGBoost (XGB), random forest (RF), and multi-layer perceptron (MLP). The classification performance was evaluated using the weighted average of F-measures (WAF).

3 Results and Discussion

We conducted two types of experiments: (i) ***Same version***, where we train the model with randomly chosen 10 000 protein interacting pairs from the archived STRING version and test it with the remaining pairs; (ii) ***Future version***, where we train the model with randomly chosen 10 000 protein pairs from the archived STRING version and test it on data from the current STRING version (excluding interactions present in the archived version). The same randomly chosen 10 000 protein pairs are used in both settings.

Since we used three archived versions, the *Future version* experiments also allow us to measure the impact of using increasingly older versions of STRING and GO in training. Table 2 shows no substantial differences between *Same version* and *Future version* experiments.

The results do not support a clear indication for data bias. While for the 2019 version, it is always slightly easier to predict future PPIs, this is reversed in the 2017 version, and varies between methods for the 2015 version, so no clear trend is discernible. The median weighted F-measure for the *Same version* experiments is 0.844, while it is 0.845 for the *Future version* (see Figure 1).

In addition to not detecting data leakage, the results also indicate that the relation between the functions of a protein and its interactions do not fundamentally change over time. Even for more recently discovered interactions that can be biologically different, protein functions are still a good predictor of PPIs.

4 Conclusion

Biomedical data resources share, reuse and import data from each other routinely. This can be a potential source of data leakage for machine learning applications. We investigated potential data leakage between the GO KG and the STRING database in the task of PPI prediction, by comparing performance on unseen interactions using archived data. Our results were not able to detect an

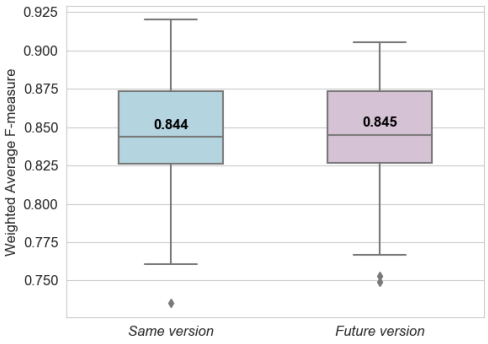


Fig. 1. Weighted Average F-measure Boxplot using the *Same version* and the *Future version* to test.

influence of data leakage, indicating that if this problem exists, its magnitude is not affecting the performance of KG-based PPI predictions.

Acknowledgements

CP, SS, RTS are funded by the FCT through LASIGE Research Unit, ref. UIDB/00408/2020 and ref. UIDP/00408/2020. CP and RTS are funded by project SMILAX (ref. PTDC/EEI-ESS/4633/2014), SS by projects BINDER (ref. PTDC/CCI-INF/29168/2017) and PREDICT (ref. PTDC/CCI-CIF/29877/2017), and RTS

ML	SSM	KG Version 01/19		05/17		04/15	
		Same	*Future*	*Same*	*Future*	*Same*	*Future*
KNN	ResnikMax	0.905	0.892	0.877	0.891	0.853	0.861
	SimGIC	0.858	0.843	0.821	0.842	0.825	0.826
	RDF2Vec	0.832	0.813	0.788	0.806	0.815	0.800
GP	ResnikMax	0.896	0.893	0.877	0.888	0.856	0.843
	SimGIC	0.873	0.855	0.835	0.859	0.835	0.842
	RDF2Vec	0.848	0.829	0.813	0.830	0.836	0.823
DT	ResnikMax	0.900	0.880	0.863	0.875	0.855	0.852
	SimGIC	0.815	0.800	0.768	0.788	0.767	0.772
	RDF2Vec	0.784	0.767	0.735	0.753	0.761	0.749
XGB	ResnikMax	0.920	0.903	0.887	0.906	0.874	0.880
	simGIC	0.873	0.858	0.839	0.860	0.834	0.846
	RDF2Vec	0.851	0.830	0.812	0.831	0.837	0.821
RF	ResnikMax	0.912	0.902	0.885	0.902	0.867	0.880
	SimGIC	0.874	0.858	0.837	0.859	0.832	0.845
	RDF2Vec	0.851	0.830	0.813	0.831	0.838	0.822
MLP	ResnikMax	0.902	0.894	0.880	0.896	0.860	0.869
	SimGIC	0.871	0.857	0.838	0.861	0.835	0.845
	RDF2Vec	0.851	0.829	0.813	0.831	0.839	0.823

Table 2. Weighted Average of F-Measures for each combination of semantic similarity measure (SSM) and machine learning (ML) algorithm for different GO KG version.

by FCT PhD grant (ref. SFRH/BD/145377/2019). It was also partially supported by the KATY project which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017453.

References

1. Akrami, F., Saeef, M.S., Zhang, Q., Hu, W., Li, C.: Realistic re-evaluation of knowledge graph completion methods: An experimental study. In: Proc. of the 2020 ACM SIGMOD Int. Conference on Management of Data. pp. 1995–2010 (2020)
2. Huntley, R.P., Sawford, T., Mutowo-Meullenet, P., Shypitsyna, A., Bonilla, C., Martin, M.J., O'Donovan, C.: The GOA database: gene ontology annotation updates for 2015. Nucleic acids research **43**(D1), D1057–D1063 (2015)
3. Kaufman, S., Rosset, S., Perlich, C., Stitelman, O.: Leakage in data mining: Formulation, detection, and avoidance. ACM Transactions on Knowledge Discovery from Data (TKDD) **6**(4), 1–21 (2012)
4. Pesquita, C., Faria, D., Bastos, H., Ferreira, A.E., Falcão, A.O., Couto, F.M.: Metrics for GO based protein semantic similarity: a systematic evaluation. In: BMC Bioinformatics. vol. 9, pp. 1–16. Springer (2008)
5. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proc. of the 14th Int. Joint Conference on Artificial Intelligence - Volume 1. p. 448–453. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1995)
6. Ristoski, P., Paulheim, H.: RDF2Vec: RDF graph embeddings for data mining. In: The Semantic Web. pp. 498–514 (2016)
7. Sousa, R.T., Silva, S., Pesquita, C.: evoKGsim+: a framework for tailoring knowledge graph-based similarity for supervised learning. In: ESWC 2021 Poster and Demo Track (2021)
8. Szklarczyk, D., Gable, A.L., Nastou, K.C., Lyon, D., Kirsch, R., Pyysalo, S., Doncheva, N.T., Legeay, M., et al.: The STRING database in 2021: customizable protein–protein networks, and functional characterization of user-uploaded gene/measurement sets. Nucleic acids research **49**(D1), D605–D612 (2021)
9. The Gene Ontology Consortium: The Gene Ontology Resource: 20 years and still GOing strong. Nucleic Acids Research **47**(D1), D330–D338 (11 2018)
10. Tomczak, A., Mortensen, J.M., Winnenburg, R., Liu, C., Alessi, D.T., Swamy, V., Vallania, F., Lofgren, S., Haynes, W., Shah, N.H., et al.: Interpretation of biological experiments changes with evolution of the gene ontology and its annotations. Scientific reports **8**(1), 1–10 (2018)
11. Zhong, X., Rajapakse, J.C.: Graph embeddings on gene ontology annotations for protein–protein interaction prediction. BMC bioinformatics **21**(16), 1–17 (2020)

Appendix G

evoKGsim+: a framework for tailoring Knowledge Graph-based similarity for supervised learning

evoKGsim+: a framework for tailoring Knowledge Graph-based similarity for supervised learning

Rita T. Sousa ✉, Sara Silva, Catia Pesquita

LASIGE, Faculdade de Ciências da Universidade de Lisboa, Portugal
{risousa,sgsilva,clpesquita}@ciencias.ulisboa.pt

Abstract. Knowledge graphs represent an unparalleled opportunity for machine learning, given their ability to provide meaningful context to the data through semantic representations. However, general-purpose knowledge graphs may describe entities from multiple perspectives, with some being irrelevant to the learning task. Despite the recent advances in semantic representations such as knowledge graph embeddings, existing methods are unsuited to tailoring semantic representations to a specific learning target that is not encoded in the knowledge graph.
We present evoKGsim+, a framework that can evolve similarity-based semantic representations for learning relations between knowledge graph entity pairs, which are not encoded in the graph. It employs genetic programming, where the evolutionary process is guided by a fitness function that measures the quality of relation prediction. The framework combines several taxonomic and embedding similarity measures and provides several baseline evaluation approaches that emulate domain expert feature selection and optimal parameter setting.

1 Introduction

Knowledge graphs (KGs) have been explored as providers of features and background knowledge in a wide variety of machine learning (ML) application scenarios [8]. One of these is predicting relations between KG entities that are not encoded in the KG, a problem cast as a classification task that takes as input a KG and a set of KG entity pairs. In the biomedical domain, ontologies are commonly employed to describe biological entities through semantic annotation. Tasks such as predicting protein-protein interactions using the Gene Ontology (GO) [12] or the mining of gene-disease associations on the Human Phenotype Ontology (HPO) [1] can be framed in this scenario.

In these cases, when we have a general-purpose KG (e.g., a KG that includes proteins and described their functions) that we aim to explore in the context of an independent and specific learning task (e.g., predicting if two proteins interact), it may very well be the case that large portions of the KG are irrelevant for the task. While in node/link/type prediction, instance representations such as embeddings [10] may be trained within the context of a particular learning task,

in our scenario, no such tuning is possible since the classification targets are not a part of the KG. This problem is exacerbated in complex domains, such as the biomedical, where KGs represent multiple views (or semantic aspects) over the underlying data, some of which may be less relevant to train the model towards a specific target. For instance, the prediction of protein-protein interactions using the GO is more accurate if only a portion of the ontology is used [9] (in this case, the one concerning biological processes).

This brings us to the challenge of tailoring the semantic representation (SR) of the KG entities to an independent and specific classification task when the classification target is not encoded in the KG. A KG-based SR is a set of features describing a KG entity obtained by processing the KG and bridge the gap between KGs and the typical vector-based representations of entities used by most ML techniques. Most state-of-the-art KG-based numeric representations are based on graph embeddings [10], which produce feature vector (propositional) representations of the KG entities. Taxonomic semantic similarity [4] can also be used as an SR by comparing entities based on the properties they share and their taxonomic relationships. Both types of approaches are, in fact, methods for feature generation, but they also result in feature selection by the heuristics and approaches they employ in creating the representations.

To address the specific goal of predicting relations between KG entities when those relations are not encoded in the KG, we postulate that similarity between the entities is a suitable frame for SR to be used by downstream supervised learning approaches. Then, the problem is how to tailor a given semantic similarity representation to the classification task, i.e., classifying a pair of entities as related or not. Previously, we presented evoKGsim [9], a methodology that learns suitable semantic similarity-based SRs of data objects extracted from KGs optimized for supervised learning. This tailoring is achieved by evolving a suitable combination of semantic aspects using Genetic Programming (GP) using taxonomy-based semantic similarity measures.

In this work, we present an extension of evoKGsim into a full framework, evoKGsim+, that encompasses 10 KG-based similarity measures based on a selection of representative state-of-the-art KG embeddings and taxonomic similarity approaches. We evaluate the framework in its full extension in benchmark datasets devoted to protein-protein interaction (PPI) prediction.

2 Methodology

evoKGsim+ targets classification tasks that take as input a KG and a set of KG individual pairs for which we wish to learn a relation that is outside the scope of the KG. The models are trained using external information about the classification targets for each pair. The evoKGsim+ framework is able to: (1) compute semantic similarity-based representations of KG individuals according to different semantic aspects and using different similarity approaches; (2) employ GP to learn a suitable representation targeted to a supervised learning task by combining the different semantic aspects; and (3) evaluate the outcome of

(2) against a set of static representations emulating experts. An overview of the framework is shown in Figure 1.

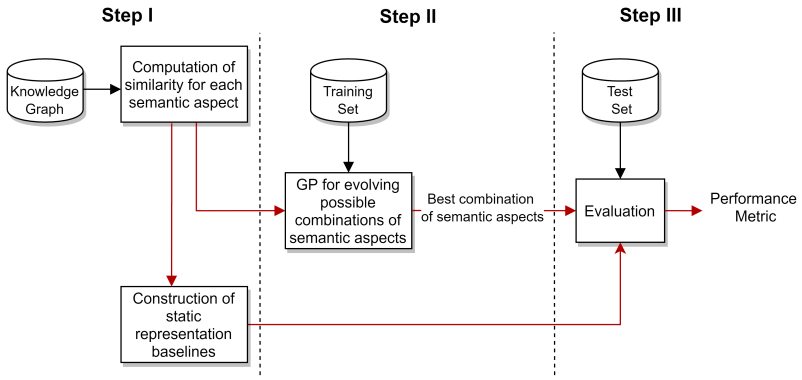


Fig. 1. Overview of the evoKGsim+ framework.

The first step of the framework is to represent each instance (i.e., a pair of KG entities) according to KG-based similarities computed for each semantic aspect. Currently, evoKGsim+ takes as semantic aspects the subgraphs rooted in the classes at a distance of one from the root class of the T-box in the KG, but this parameter can be easily adjusted. The second step is to employ GP to learn a suitable combination of the different aspect-based similarities, using a set of predefined operators, to address a given ML task. The last step is to evaluate the predictions made on the test set, and comparing them against optimized static representations that represent expert feature selection and parameter tuning.

This framework is independent of the specific implementation of KG-based similarity and the GP parameters employed to evolve the representations. Currently, evoKGsim+ supports 10 different KG-based similarity measures: 6 taxonomic similarity measures, derived by combining one of two information content approaches (IC_{Seco} and IC_{Resnik}) with one of three set similarity measures (ResnikMax, ResnikBMA, and SimGIC [6]); 4 measures based on cosine similarity over embeddings generated from TransE[2], distMult[11], RDF2Vec [7] and Owl2Vec [5].

3 Evaluation

PPI prediction was chosen as our evaluation domain for the following reasons: (1) it is backed by a large ontology with multiple semantic aspects, the GO; (2) there are gold-standard datasets [3]; (3) it is well known that the GO aspects biological process (BP) and cellular component (CC) describe properties that are stronger indicators for PPI than the molecular function aspect (MF) [9], which provides an ideal test bed for the need of adapting the SR to the learning task.

Table 1 presents the results obtained using different similarity-based SRs. As baselines, we have used five static SRs (the BP, CC and MF single aspects, and the average and maximum of the single aspect similarities). The static SRs are based on a simple threshold-based classifier, where a similarity score for a protein pair above the threshold predicts a positive interaction. For evaluating the quality of a predicted classification, the weighted average F-measure (WAF) was used for stratified 10-fold cross-validation.

Table 1. Median of WAF for 10-fold cross-validation.

Similarity Measure	**Static SRs**					**evoKGsim**
	BP	CC	MF	Avg	Max	
ResnikMax + IC_{Seco}	0.760	0.713	0.646	0.749	0.743	**0.765**
ResnikMax + IC_{Resnik}	0.750	0.717	0.653	0.766	0.774	**0.776**
ResnikBMA + IC_{Seco}	0.753	0.715	0.643	0.771	0.744	**0.777**
ResnikBMA + IC_{Resnik}	0.753	0.714	0.648	0.777	0.772	**0.782**
SimGIC + IC_{Seco}	0.736	0.682	0.642	0.729	0.701	**0.746**
SimGIC + IC_{Resnik}	0.739	0.704	0.651	0.750	0.734	**0.758**
TransE	0.501	**0.534**	0.502	0.519	0.521	0.521
distMult	0.704	0.599	0.498	0.670	0.668	**0.712**
RDF2Vec	0.675	0.654	0.631	0.684	0.668	**0.685**
Owl2vec	0.678	0.662	0.621	**0.693**	0.686	**0.693**

evoKGsim with taxonomic similarity always achieves the best performance compared to the static SRs. Regarding the graph embedding approaches, TransE has performed worse than the other embedding methods. These differences are not unexpected since we are interested in learning which aspects of a KG are more relevant to the learning task, and most of the information to be processed is represented in the ontology portion of the KG, where taxonomic relations play an important role. Therefore, translational distance approaches that emphasize local neighbourhoods are less suitable than semantic matching methods, like disMult, or methods that capture longer-distance relations, such as path-based approaches (RDF2Vec and Owl2Vec).

When comparing the two SRs, evoKGsim with taxonomic similarity achieves a better performance than evoKGsim with embedding similarity. Although embeddings consider all types of relations, we hypothesize that taxonomic similarity can take into account class specificity that may give it the advantage over embedding similarity in more accurately estimating similarity.

4 Conclusion

We have developed a framework, evoKGsim+, that tailors KG-based similarity representations for supervised learning of relations between KG instances when the classification target is not encoded in the KG. We have shown that evoKGsim+ can generate tailored SRs that improve classification performance

over static SRs both using embedding similarity and taxonomic semantic similarity. This framework can be readily generalized to other applications and domains, where KG-based similarity is a suitable instance representation, such as prediction of drug-target interactions and gene-disease association, KG link prediction or recommendations.

Acknowledgements CP, SS, RTS are funded by the FCT through LASIGE Research Unit, ref. UIDB/00408/2020 and ref. UIDP/00408/2020. CP and RTS are funded by project SMILAX (ref. PTDC/EEI-ESS/4633/2014), SS by projects BINDER (ref. PTDC/CCI-INF/29168/2017) and PREDICT (ref. PTDC/CCI-CIF/29877/2017), and RTS by FCT PhD grant (ref. SFRH/BD/145377/2019). It was also partially supported by the KATY project which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101017453.

References

1. Asif, M., Martiniano, H.F., Vicente, A.M., Couto, F.M.: Identifying disease genes using machine learning and gene functional similarities, assessed through gene ontology. PLoS ONE **13**(12), e0208626 (2018)
2. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in neural information processing systems. pp. 2787–2795 (2013)
3. Cardoso, C., Sousa, R.T., Köhler, S., Pesquita, C.: A collection of benchmark data sets for knowledge graph-based similarity in the biomedical domain. Database (2020)
4. Harispe, S., Ranwez, S., Janaqi, S., Montmain, J.: Semantic Similarity from Natural Language and Ontology Analysis. Morgan & Claypool Publishers (2015)
5. Holter, O.M., Myklebust, E.B., Chen, J., Jimenez-Ruiz, E.: Embedding OWL ontologies with OWL2vec. In: CEUR Workshop Proceedings. vol. 2456, pp. 33–36. Technical University of Aachen (2019)
6. Pesquita, C., Faria, D., Bastos, H., Ferreira, A.E., Falcão, A.O., Couto, F.M.: Metrics for GO based protein semantic similarity: a systematic evaluation. In: BMC Bioinformatics. vol. 9, pp. 1–16. Springer (2008)
7. Ristoski, P., Paulheim, H.: RDF2Vec: RDF graph embeddings for data mining. In: The Semantic Web. pp. 498–514 (2016)
8. Ristoski, P., Paulheim, H.: Semantic Web in data mining and knowledge discovery: A comprehensive survey. Journal of Web Semantics **36**, 1–22 (2016)
9. Sousa, R.T., Silva, S., Pesquita, C.: Evolving knowledge graph similarity for supervised learning in complex biomedical domains. BMC Bioinformatics (2019)
10. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge graph embedding: A survey of approaches and applications. IEEE TKDE **29**(12), 2724–2743 (2017)
11. Yang, B., Yih, S.W.t., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: Proceedings of the ICLR (2015)
12. Zhong, X., Kaalia, R., Rajapakse, J.C.: GO2Vec: transforming GO terms and proteins to vector representations via graph embeddings. BMC Genomics **20**(9), 1–10 (2019)