UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Predicting Gene-Disease Associations with Knowledge Graph Embeddings over Multiple Ontologies

Susana Catarina Plácido Nunes

Mestrado em Bioinformática e Biologia Computacional

Dissertação orientada por:
Professora Doutora Cátia Luísa Santana Calisto Pesquita

2021

Acknowledgments

First, I would like to thank my supervisor, Professor Cátia Pesquita, for her relentless support throughout this dissertation. Her mentoring was crucial, still is, and will be for my future. I am truly grateful for her dedication and for seeing potential in me and in my hard work.

My most special thanks is to my dear colleague Rita with the purest heart I know. Thank you for welcoming me into the LASIGE family and being a role model along the process always motivating me to be better even without knowing you were doing that. It has been a true pleasure to work alongside you and hopefully still will for many years.

To my big brother who financially and emotionally supported me throughout my academic life and has been a strong father figure all my life. Without you, I would never achieve all the success I am having and hoping to have. Just wish someday I can repay everything you did for me. To my niece, for just being the cutest little baby in this world. Hope to be someday a role model for you. And thank you for my brother to make such a incredible tiny human and for being my annoying brother that I care just a little bit.

To Vera, my best friend, my ride or die. My thanks to you have no limits and would require a full page just to point out a small part of gratitude I have for you. We have come so far I am just gonna say that I see a bright future for us and really hope we end up very rich old ladies together. Alone but together, always. To my annoying but awesome best friend André for accompanying me on this incredible journey and always helping me and motivating me to move forward. Prepare yourself for all the future years putting up with me. To my dear friend Ricardo for all the great academic experiences, we shared together over the years. Without you, academic life wouldn't be the same. To Cavaleira, my oldest and dearest friend, you are amazing, thank you for always being there, whether face to face or in thousand-hour calls.

To every other person that has passed my life and marked it in some kind of way. Without those experiences, I wouldn't be where I am today.

Finally, I would like to thank to Fundação para a Ciência e a Tecnologia, which provides the funding under LASIGE Research Unit (ref. UIDB/00408/2020 and ref. UIDP/00408/2020. It was also partially supported by the KATY project which has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 101017453.

Abstract

There are still more than 1,400 Mendelian conditions whose molecular cause is unknown. In addition, almost all medical conditions are somehow influenced by human genetic variation. This challenge also presents itself as an opportunity to understand the mechanisms of diseases, thus allowing the development of better mitigation strategies, finding diagnostic markers and therapeutic targets. Deciphering the link between genes and diseases is one of the most demanding tasks in biomedical research. Computational approaches for gene-disease associations prediction can greatly accelerate this process, and recent developments that explore the scientific knowledge described in ontologies have achieved good results. State-of-the-art approaches that take advantage of ontologies or knowledge graphs for these predictions are typically based on semantic similarity measures that only take into consideration hierarchical relations. New developments in the area of knowledge graphs embeddings support more powerful representations but are usually limited to a single ontology, which may be insufficient in multi-domain applications such as the prediction of gene-disease associations.

This dissertation proposes a novel approach of gene-disease associations prediction by exploring both the Human Phenotype Ontology and the Gene Ontology, using knowledge graph embeddings to represent gene and disease features in a shared semantic space that covers both gene function and phenotypes. Our approach integrates different methods for building the shared semantic space, as well as multiple knowledge graph embeddings algorithms and machine learning methods. The prediction performance was evaluated on curated gene-disease associations from DisGeNET and compared to classical semantic similarity measures. Our experiments demonstrate the value of employing knowledge graph embeddings based on random walks and highlight the need for closer integration of different ontologies.

Keywords: ontologies, semantic similarity, knowledge graph, knowledge graph embedding, machine learning

Resumo Alargado

Existem ainda, mais de 1.400 condições mendelianas cuja causa molecular é desconhecida e quase todas as condições médicas são, de alguma forma, influenciadas pela variação genética humana. Descobrir a base genética das doenças continua a ser um desafio em aberto, apesar dos tremendos avanços em genómica nas últimas duas décadas. Além disso, a maioria das doenças apresenta um genótipo altamente heterogéneo, o que dificulta identificação de marcadores biológicos. Doenças como o transtorno do espectro do autismo ou doenças cardiovasculares, que costumam ter múltiplas etiologias com o envolvimento de possivelmente centenas de diferentes genes representam um desafio adicional.

Este desafio também se apresenta como uma oportunidade para compreender mecanismos de doenças, permitindo assim o desenvolvimento de melhores estratégias de atenuação, encontrando marcadores de diagnóstico e alvos terapêuticos. Decifrar a ligação entre genes e doenças é uma das tarefas mais exigentes na investigação biomédica. Desenvolvimentos recentes que exploram o conhecimento científico descrito nas ontologias tem alcançado bons resultados.

Experiências de alto rendimento, tais como estudos de *linkage*, geram uma grande quantidade de dados que podem apontar para associações entre genes e doenças. Contudo, uma validação precisa destas associações é dispendioso e demorado. Isto fomentou o desenvolvimento de abordagens computacionais para prever as associações de doenças genéticas e identificar associações mais promissoras a validar futuramente.

Centenas de ontologias foram desenvolvidas, cobrindo quase todos os domínios de investigação biológica e biomédica. As ontologias biomédicas têm-se tornado cada vez mais importante para estruturar e descrever conhecimento biológico existente e impulsionaram um novo panorama de dados biomédicos semânticos, onde milhões de entidades biomédicas semanticamente descritas são anotadas com conceitos de ontologias e estruturadas em *knowledge graphs*. Ter dados num *knowledge graph* possibilita uma representação semanticamente rica e partilhada de dados, e que permite codificar as provas por detrás de uma asserção.

Recentemente, abordagens mais sofisticadas baseadas em incorporações de *knowledge graph embeddings* permitem a representação de cada entidade com um vetor que aproxima as propriedades de semelhança do grafo e pode ser utilizado para calcular semelhança

ou aplicar num algoritmo de aprendizagem automática. *Knowledge graph embeddings* suportam em principio representações mais poderosas que consideram múltiplos tipos de relações enquanto que a semelhança semântica é limitada a relações hierárquicas.

O principal objetivo deste trabalho é investigar abordagens para prever as associações de doenças genéticas que exploram a riqueza semântica dos *knowledge graphs*. O trabalho aborda a riqueza semântica de diferentes perspetivas: semelhança semântica vs. *knowledge graph embeddings*; uma ontologia vs. duas ontologias; ontologias desconectadas vs. ontologias ligadas. A nossa hipótese é que representações mais ricas semanticamente, alimentadas pelo conhecimento dos *knowledge graph embeddings* computadas em múltiplas ontologias interligadas conseguem uma melhor performance de previsão do que abordagens mais simples baseadas na semelhança semântica ou *knowledge graph embeddings* usando uma única ontologia.

Explorámos a *Human Phenotype Ontology* e a *Gene Ontology*, duas das mais populares ontologias biomédicas, para representar as características do gene e da doença num espaço semântico partilhado que abrange tanto a função genética como os fenótipos. Temos como hipótese de que a utilização de mais do que uma ontologia pode melhorar a previsão da associação gene-doença e que uma integração mais rica pode ter um impacto positivo. Em particular, consideramos as definições lógicas que associam entidades de diferentes ontologias com relações semânticas complexas podem ser exploradas para ligar domínios e contextualizar relações entre diferentes entidades, tais como um gene e uma doença.

A metodologia proposta neste trabalho pode ser dividida em 4 passos principais. O primeiro passo na abordagem é integrar as diferentes ontologias e dados de anotações para construir cinco tipos de *knowledge graphs*. Numa segunda etapa, são criadas as incorporações que representam o gene e a doença de acordo com as suas anotações em diferentes *knowledge graphs*. Numa terceira etapa, os *embeddings* são combinados utilizando 5 tipos de operadores vetoriais (Concatenação, Média, Hadamard, Weighted-L1, Weighted-L2), produzindo uma representação de genes e doenças naquilo que é efetivamente um espaço semântico partilhado. Finalmente, numa quarta etapa, quatro algoritmos de aprendizagem automática são treinados sobre os vários *knowledge graph embeddings* de genes para prever as associações de genes e doenças. Gerámos *knowledge graph embeddings* com uma dimensão de 200 características e utilizámos cinco métodos diferentes que cobrem diferentes abordagens: distância translacional (TransE), correspondência semântica (DistMult) e caminhos aleatórios (RDF2Vec, OPA2Vec, OWL2Vec). O desempenho da previsão foi avaliado em associações de doenças genéticas curadas da DisGeNET e contra abordagens baseadas em seis medidas clássicas de semelhança semântica (linha de base que permite estabelecer a performance de métodos que usam uma única ontologia e medidas de semelhança semântica clássica) e *knowledge graphs* que incorporam similaridade.

O desempenho das classificações foi avaliado a média ponderada das *F-measures*. Também avaliámos o desempenho com a classificação AUC (área sob a curva da característica de operação do receptor (ROC)). Em cada experiência, realizámos um método de divisão estratificado de 70% treino e de 30% dos testes, sendo que a mesma divisão foi utilizada em todas as experiências, incluindo na linha de base. A previsão da associação na linha de base é expressa como um problema de classificação em que um valor de semelhança semântica para um par gene-doença que exceda um determinado limiar indica uma associação positiva. Para cada medida, foi escolhido um limiar após avaliação da média ponderada das medidas F (para previsões positivas e negativas) em diferentes limiares e seleção do máximo (valores no intervalo de 0 a 1).

Há vários fatores da metodologia proposta para as representações semânticas ricas que podem ter impacto no desempenho da previsão da associação genes-doença, tais como a riqueza semântica e a cobertura do domínio do *knowledge graph*, os métodos de *knowledge graph embedding*, e os operadores utilizados para combinar os vetores do gene e da doença. Dados estes fatores, há três aspetos importantes que precisam de ser considerados ao elucidar o impacto do desempenho: (1) como combinar os vetores do gene e da doença; (2) que métodos de *knowledge graph embeddings* são mais adequados para esta tarefa; (3) qual é o impacto de considerar mais de uma ontologia.

De um modo geral, os resultados demonstraram que as incorporações de *knowledge graph embeddings* quando acopladas a algoritmos de aprendizagem automática alcançam um melhor desempenho do que as medidas de semelhança semântica. OPA2Vec alcança os melhores resultados, em conjunto com Hadamard, e é também capaz de tirar partido da integração de ontologias com uma pequena melhoria no desempenho. OPA2Vec gera um corpus com um conjunto de axiomas afirmados e inferidos de uma ontologia, o conjunto de axiomas de anotação envolvendo designações, descrições, sinónimos e criadores, e as anotações do fenótipo genético e da doença. Ao incluir todas estas características, esta é uma causa provável para se apresentar como um algoritmo mais estável e bem sucedido em relação aos outros.

No entanto, as nossas experiências revelam que as diferenças entre a utilização de uma única ontologia ou a combinação de duas ontologias (*Human Phenotype Ontology* e *Gene Ontology*) são comparativamente pequenas, independentemente de uma integração mais rica utilizando definições lógicas. Uma possível hipótese é o facto de que a informação fornecida pelas definições lógicas não fornece informação adicional substancial comparando com o que já está presente na *Human Phenotype Ontology*. Isto pode ser parcialmente explicado pela existência de apenas 350 definições lógicas que ligam as duas ontologias. Os próximos passos irão explorar técnicas de correspondência de ontologias para criar definições lógicas adicionais e ligações entre ontologias, o que também nos permitirá expandir as ontologias utilizadas para aquelas que não contêm definições lógicas entre elas.

Palavras-chave: ontologias, semelhança semântica, grafos de conhecimento, representações semânticas, aprendizagem automática

Contents

List of Figures

List of Tables

Acronyms

ABox Assertion Box.

ASD Autism Spectrum Disorder.

AUC-ROC Area Under the Receiver Operating Characteristic Curve.

BMA Best-Match Average.

BP Biological Process.

CC Cellular Component.

CS Cosine Similarity.

DAG Directed Acyclic Graph.

DO Disease Ontology.

GAF Gene Association File.

GO Gene Ontology.

GOA Gene Ontology Annotation.

HP Human Phenotype Ontology.

IC Information Content.

KG Knowledge Graph.

KGE Knowledge Graph Embeddings.

LD Logical Definitions.

Max Maximum.

MF Molecular Function.

MGI Mouse Genome Informatics.

ML Machine Learning.

MLP Multi-Layer Perceptron.

MPO Mammalian Phenotype Ontology.

NB Naïve Bayes

OBO Open Biomedical Ontology.

OMIM Online Mendelian Inheritance in Man.

OWL Web Ontology Language.

RDF Resource Description Framework.

RF Random Forest.

SS Similarity Score.

SSM Semantic Similarity Measure.

TBox Terminology Box.

TSV Tab-separated Values.

URI Uniform Resource Identifier.

WAF Weighted Average of F-measures.

XGB eXtreme Gradient Boosting.

XML Extensible Markup Language.

Chapter 1

Introduction

There are more than 1,400 Mendelian conditions (single genetic locus) whose molecular cause is unknown (Amberger et al., 2014). In addition, almost all medical conditions are somehow influenced by human genetic variation. Uncovering the genetic basis of diseases remains an open challenge despite the tremendous advances in genomics of the past two decades. Genomics studies and high-throughput experiments often produce large lists of candidate genes, of which only a small portion is truly relevant to the disease, phenotype, or biological process of interest.

Furthermore, most diseases present a highly heterogeneous genotype, which hinders biological marker identification. Diseases like Autism Spectrum Disorder or Cardiovascular Disease that often have multiple etiologies with the involvement of possibly hundreds of different genes represent an additional challenge (Asif et al., 2018). However, this challenge also presents itself as an opportunity to understand the mechanisms of diseases and human biology by exploring the interplay between genes, phenotypes, and diseases, uncovering new diagnostic markers and therapeutic targets.

High-throughput experiments such as linkage studies, genome-wide association studies, and RNA interference screens generate a large amount of data that can point towards associations between genes and diseases. However, a precise validation of these associations in the wet lab is expensive and time-consuming. This fostered the development of computational approaches for predicting gene-disease associations and identifying the most promising associations to be further validated. These approaches typically explore diverse databases (e.g., OMIM, DisGeNet, dbSNP) and employ a diversity of computational approaches ranging from machine learning to network-based algorithms.

Opap and Mulder (Opap and Mulder, 2017) have identified three main challenges in gene–disease associations:

1. how to represent the data in a readily accessible manner for researchers;
2. how to attribute evidence to assertions made by algorithms;
3. how to scale the algorithms with the rate of increase in data size and complexity.

Methods that explore the scientific knowledge described in ontologies can provide an answer to the first two challenges. Ontologies are formal and explicit specifications of a conceptualization of a given domain. They provide a structured way to define concepts and relations between them and have been used in the biomedical domain for the past two decades to support a shared and computationally amenable description of biological entities. Hundreds of ontologies have been developed, covering almost all domains of biological and biomedical research. Biomedical ontologies have become increasingly important to structure and describe existing biological knowledge and have propelled a new panorama of semantic biomedical data, where millions of semantically described biomedical entities are annotated with ontology concepts and structured in knowledge graphs. Having data in a knowledge graph allows for a shared and semantically rich representation of the data, and also allows encoding the evidence behind an assertion.

The third challenge is not directly addressed by ontologies and knowledge graphs, however, ontologies and knowledge graphs can be explored by different algorithmic approaches that can tackle the challenges of data size and perhaps more importantly, complexity.

There are several well-established works that explore semantic similarity algorithms in the context of ontologies. Semantic similarity expresses the similarity between two entities based on their shared meaning. For example, if both genes and diseases are annotated under the same ontology (e.g. Human Phenotype Ontology (HP)), we can compare them by comparing the classes (which in the case of HP describe phenotypes) with which they are annotated. Semantic similarity provides a single score view of an association between a gene and a disease.

Recently, more sophisticated approaches based on knowledge graph embeddings allow the representation of each entity with a vector that approximates the similarity properties of the graph (Wang et al., 2017) and can then be used either to compute similarity or to feed a machine learning algorithm. Knowledge graph embeddings support in principal more powerful representations that semantic similarity since that consider multiple types of relations and are multi-dimensional. However, in a complex task such as predicting gene-disease associations, employing a single graph with a single ontology may be insufficient, since multiple perspectives may be necessary for prediction, such as gene function and phenotype.

1.1 Objectives

The main goal of this work is to investigate approaches to predict gene-disease associations that explore the semantic richness of knowledge graphs. The work tackles semantic richness from different perspectives: semantic similarity vs. knowledge graph embeddings; one ontology vs. two ontologies; disconnected ontologies vs. linked on-

tologies. Our guiding hypothesis is that richer representations, powered by knowledge graph embeddings computed over multiple linked ontologies achieve a better predictive performance than simpler approaches based on semantic similarity or knowledge graph embeddings using a single ontology.

The work aims to answer three research questions:

1. RQ1: What are the advantages of knowledge graph embedding over semantic similarity measures as a representation strategy?

2. RQ2: How can different knowledge graph embedding approaches be computed over multiple biomedical ontologies to represent both genes and diseases as vectors?

3. RQ3: What is the impact of employing multiple ontologies and having logical links between them?

We explore the Human Phenotype Ontology (Köhler et al., 2021) and the Gene Ontology (Consortium, 2020; Ashburner et al., 2000), two of the most popular biomedical ontologies, to represent gene and disease features in a shared semantic space that covers both gene function and phenotypes.

1.2 Contributions

The main contributions of this dissertation are:

1. Development of a novel approach with a rich semantic representation through the use of multiple ontologies and knowledge graph embedding methods prediction for gene-disease associations.

2. Creation of an unbiased benchmark dataset for gene-disease association prediction.

3. Poster with the preliminary results presented in 6th LASIGE Workshop.

4. Poster with the preliminary results presented at the Bioinformatics Opens Days 2021.

5. Short paper and Poster with the main results presented at the 29th Conference on Intelligent Systems for Molecular Biology and the 20th European Conference on Computational Biology. The short paper was also published and being considered for an extension on a special issue on the Journal of Biomedical Semantics.

1.3 Document Structure

The present introductory chapter gives a contextualization for the problem at hand and introduces the main objectives and contributions of this dissertation. The remaining five chapters are organized as follows:

- Chapter 2 defines and explains the basic concepts vital for the understanding of the problem itself, namely, ontologies, knowledge graphs, and machine learning.
- Chapter 3 presents methods in two relevant areas, semantic similarity and knowledge graph embeddings, but also overviews the field of gene-disease association.
- Chapter 4 presents an overview of the methodology developed with a description of the main tasks.
- Chapter 5 summarizes the results from the methodology implementation and discussion.
- Chapter 6 discusses the main conclusions and limitations of this work and indicates some directions for future work.

Chapter 2

Background

2.1 Ontologies

An ontology is a technique or technology used to represent the knowledge about a domain, by modeling concepts and the relationships between them, being that these relationships describe the properties of those concepts (Bodenreider and Stevens, 2006). Ontologies are thus semantic models for reality domains. Ontologies have two major components: **(i)** a set of classes (concepts) that define the entities in a domain; and **(ii)** a set of semantic links (relationships) between the classes that describe interactions between classes or properties of classes. Ontologies often structure their classes and the relationships between them as a Directed Acyclic Graph (DAG), where the classes are nodes and relationships are edges. An example of an excerpt of a DAG of the Gene Ontology (GO), a very successful biomedical ontology that describes the function of genes and gene products, is depicted in Figure 2.1. The relations between classes can be structured in triples, as seen in Figure 2.2. Each class or property in an ontology is identified by a unique Uniform Resource Identifier (URI) that is used to identify each component of a triple: subject, predicate, and object. The predicate (e.g. 'is_a') denotes the relationship that exists between the subject and the object.

Typically, ontologies are stored in files conforming to a specific file format, although there are exceptions that are stored in custom-built infrastructures. Ontologies can be represented in different underlying ontology languages:

- **Resource Description Framework (RDF):** is a simple language; its underlying data structure is a labeled directed graph, and its only syntactic construct is the triple that specifies the relation between the subject and the object via the predicate. A set of triples is called an RDF graph and in order to facilitate the sharing and exchanging of graphs on the Web, the RDF specification includes an Extensible Markup Language (XML) serialization (Horrocks, 2008).

- **Web Ontology Language (OWL):** more powerful language consisting of a set of axioms of 2 types: Terminology Box (TBox) and Assertion Box (ABox). TBox

axioms define a hierarchy of classes and properties, but also restrictions as the disjointness of two classes or characteristics of some properties. ABox assert facts about concrete entities and are usually not included in an ontology definition (Horrocks, 2008).

- **Open Biomedical Ontology (OBO):** compact language, readable by humans, and easy to parse composed by a header and a stanza. The header describes generic information about the ontology and each stanza enclosures the description and relations of each ontology element (Golbreich et al., 2007).

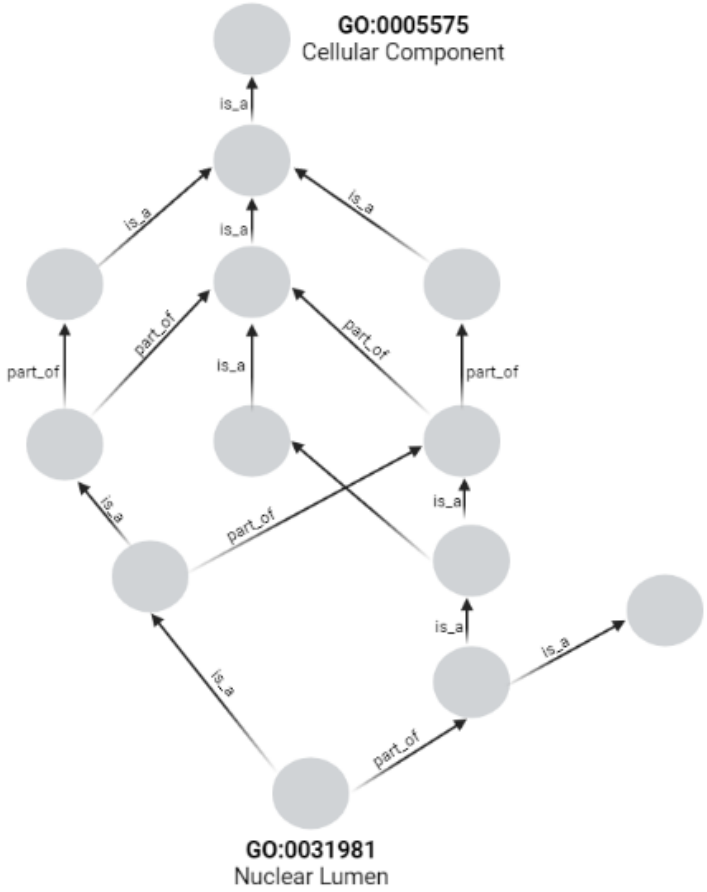

Figure 2.1: Excerpt of a DAG representing the class GO:0031981 ”Nuclear Lumen” and its anscestors. Created in BioRender.com

Open repositories such as the BioPortal (Whetzel et al., 2011) provide access to over 900 biomedical ontologies (dating august 2021) expressed in these various formats, with scopes as diverse as the characterization of gene products (Gene Ontology) to phenotypic abnormalities in human diseases (Human Phenotype Ontology).

The purposes that are supported by ontologies are diverse. The most straightforward application of ontologies is to support the structured annotation of data. Semantic annotation is about assigning real-world entities in a domain to their semantic description

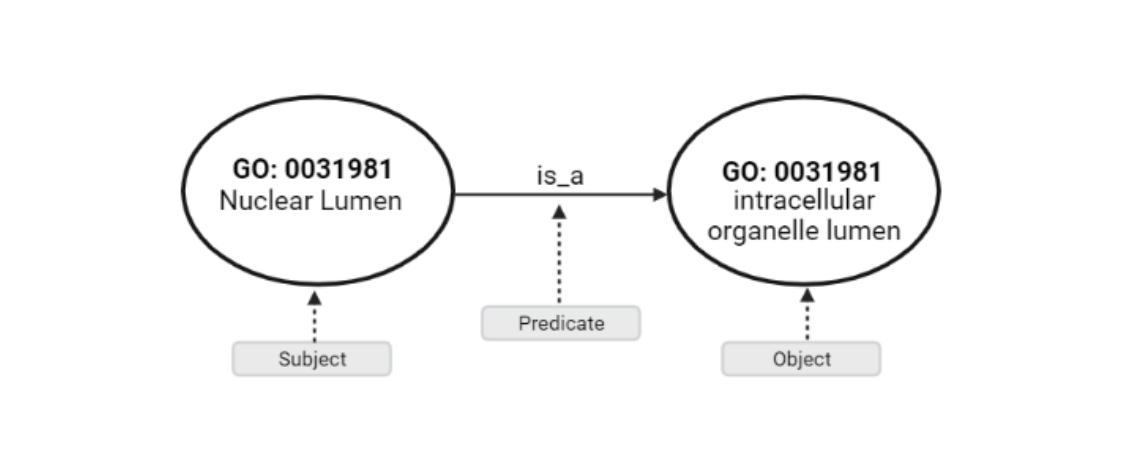

Figure 2.2: Structure and example of a triple from the GO. Created in BioRender.com

(Kiryakov et al., 2004). Relying on ontology classes to annotate biomedical entities allows automatic reasoning to be applied directly to them. Thus, the richer the ontology is in relations between classes and the thorough the annotation is, the better captured the semantic description of the entity will be (Stevens et al., 2004). In Figure 2.3 we can see an example of using the Gene Ontology to annotate a protein.

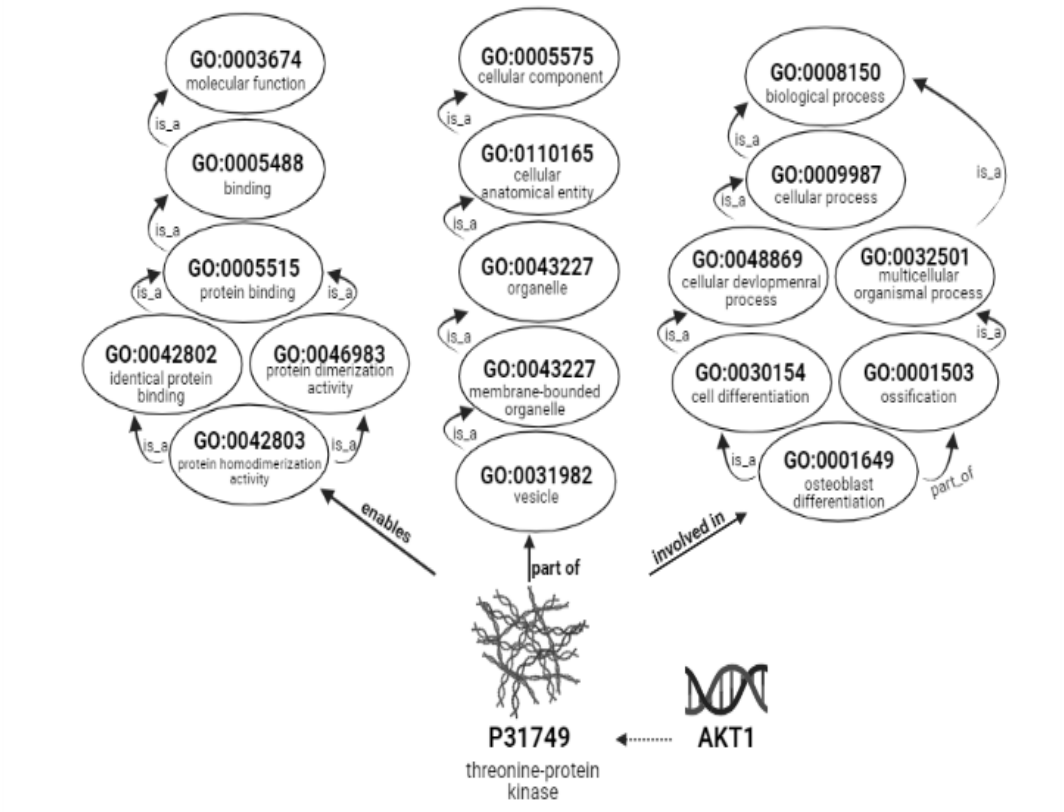

Figure 2.3: Graph representation of an excerpt of GO and GO annotations regarding the gene AKT1. Created in BioRender.com

Ontologies can be used as a rich source of vocabulary for a domain of interest, providing a dictionary of names, synonyms, and interrelationships, thereby facilitating text mining, intelligent searching, and unambiguous identification. When used in multiple independent contexts, such a common vocabulary can become additionally powerful. For instance, a shared ontology allows the comparison and translation of entities from one discipline to another such as between biology and chemistry, enabling interdisciplinary tools that would be impossible computationally without a unified reference vocabulary (Hastings, 2017).

To employ multiple ontologies for describing entities, we need to link them and many biomedical ontologies have logical definitions that relate to classes from different ontologies with complex semantic relations. A recent approach of defining classes using logical definitions is now increasingly being adopted as a method for facilitating interoperability and data integration (Köhler et al., 2011). These can be explored to bridge domains and contextualize relations between different entities, such as a gene and a disease. An example of a logical definition is the Human Phenotype ontology class for "Hearing impairment" (HP:0000365) that is equivalent to a restriction that involves four other ontologies, as depicted in Figure 2.4 allowing to create a bridge between them (this example is explored again in section 4.3).

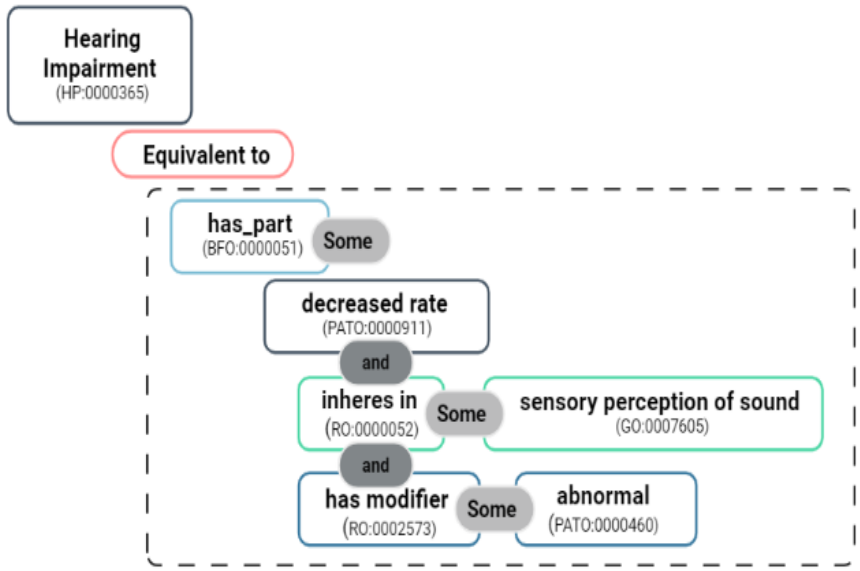

Figure 2.4: Example of a logical definition of the class Human Phenotype ontology class for "Hearing impairment" (HP:0000365). The explanation for this logical definition is 'Hearing impairment' EquivalentTo 'has part' some ('decreased rate' and ('inheres in' some 'sensory perception of sound') and ('has modifier' some 'abnormal')). Created in BioRender.com

2.2 Knowledge Graphs

Knowledge graphs structure and link data that is described using an ontology into a graph (Paulheim, 2016). These graphs provide a conceptualization of a domain based on a formal definition of its entities, that are described by associated ontological concepts, and their relations.

Within the biomedical domain, Nicholson and Greene (2020) defined a knowledge graph as a resource that integrates one or more expert-derived sources of information into a graph where nodes represent biomedical entities and edges represent relationships between two entities. In other words, the nodes of the knowledge graph are employed in representing ontology classes and RDF statements subjects and objects, and edges are employed in representing ontology classes relations and RDF statements predicates. For example, the Human Phenotype Ontology and its associated annotations that link diseases and genes to HP classes (phenotypes) and to other diseases or genes make up a knowledge graph. An example of a portion of a knowledge graph is represented in Figure 2.5, contextualized by the Human Phenotype Ontology and its annotations, where diseases and genes are linked to HP classes and to other diseases and genes.

Knowledge graphs can help tackle many problems in the biomedical domain, based on the ontological descriptions of the entities (Kulmanov et al., 2020b). For instance, finding new treatments for existing drugs, aiding efforts to diagnose patients, and identifying associations between diseases and genes.

2.3 Machine Learning

The exponential growth of biomedical data in recent years has urged the application of numerous machine learning techniques to address emerging problems in biology and clinical research. It is defined as a field in computer science that studies the use of computers to simulate human learning by exploring patterns in the data and applying self-improvement to continually enhance the performance of learning tasks (Auslander et al., 2021).

Traditional machine learning algorithms take as input a feature vector, which represents an object in terms of numeric or categorical attributes. The main learning task is to learn a mapping from this feature vector to an output prediction of some form (Nickel et al., 2016a). The algorithms can be divided into supervised and unsupervised learning algorithms. Supervised learning algorithms learn to map input examples into their respective output (subsection 2.3.1 will deepen this category). Unsupervised learning algorithms identify hidden patterns in unlabeled data.The advances made in machine learning over the past decade transformed the landscape of data analysis.

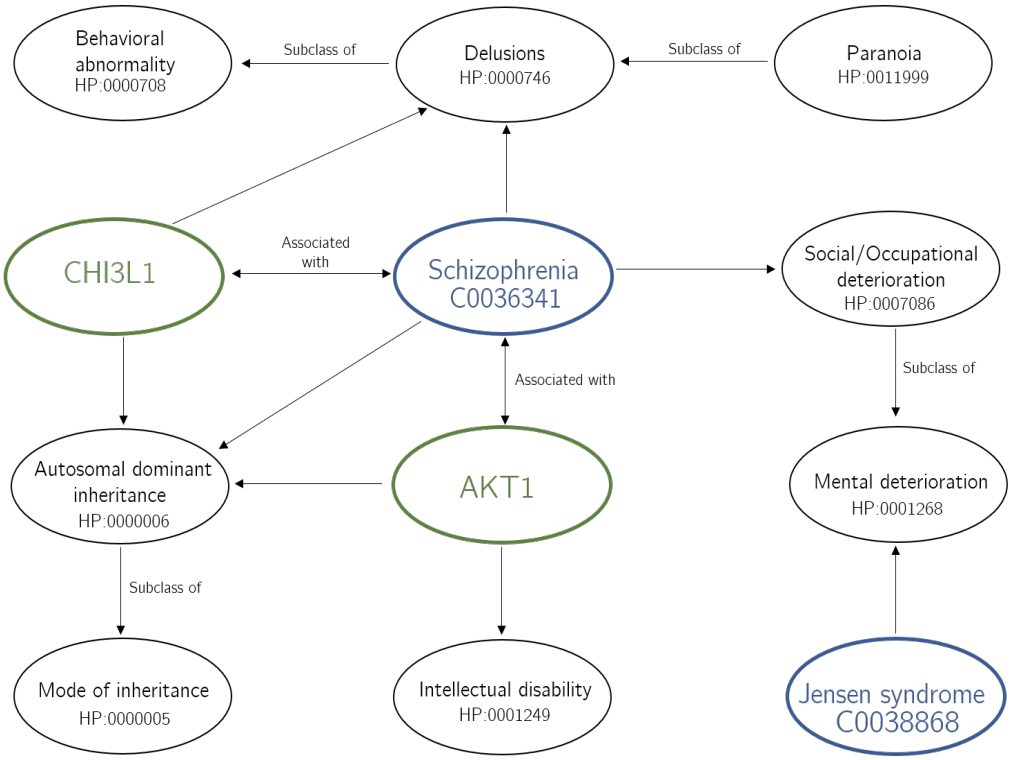

Figure 2.5: Subgraph of the HP KG illustrating the relationships between genes and diseases. The green and blue nodes are the genes and diseases (biological entities), respectively, and the black nodes are the HP classes (ontology concepts).

2.3.1 Supervised Learning

The most common form of machine learning, deep or not, is supervised learning. In supervised learning, models are trained using a labeled dataset, where the model learns about each type of data. Once the training process is completed, the model is tested on the basis of test data (a subset of the training set), and then it predicts the output.

Supervised learning can be divided further into two categories of problem:

- **Classification** uses an algorithm to accurately assign test data into specific categories. It recognizes specific entities within the dataset and attempts to draw some conclusions on how those entities should be labeled or defined. Common classification algorithms are linear classifiers, support vector machines, decision trees, k-nearest neighbor, and random forest, etc.

- **Regression** is used if there is a relationship between the input variable and the output variable. It is used for the prediction of continuous variables. Linear regression, logistical regression, and polynomial regression are popular regression algorithms.

Machine learning methods have become a rapidly growing research area, redefining the state-of-the-art performance for a wide range of fields. Given the rapid growth in the availability of biomedical and clinical datasets in the past decades, these techniques can be

expected to similarly transform multiple avenues of biomedical research, and indications of their high efficacy are already accumulating (Auslander et al., 2021).

In this work, we focused on four different supervised learning algorithms: Random Forest (RF) (Breiman, 2001), Gradient Boosting (Chen and Guestrin, 2016), Naïve Bayes (NB) (Friedman et al., 1997), and Multi-Layer Perceptron (MLP) (Rumelhart et al., 1986).

Random forest (Breiman, 2001) is a machine learning algorithm that utilizes ensemble learning, which is a technique that combines many classifiers to provide solutions to complex problems. It operates by constructing a multitude of decision trees at training time. The decision tree is a hierarchical structure that is built using the features (or the independent variables) of a dataset. This algorithm establishes the outcome based on the predictions of the decision trees and predicts by taking the average or mean of the output from various trees.

Gradient boosting (Chen and Guestrin, 2016) is a machine learning algorithm, used for both classification and regression problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. When a decision tree is a weak learner (i.e shallow tree), the resulting algorithm is called gradient boosted trees, which usually outperforms random forest. XGB is an open-source library that provides an efficient and effective implementation of the gradient boosting algorithm.

Naïve Bayes classifier (Friedman et al., 1997) is a probabilistic machine learning model based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, it assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

Multi-Layer Perceptron (Rumelhart et al., 1986) is a type of feed-forward artificial neural network. It consists of three types of layers—the input layer, output layer and hidden layer. The input layer receives the input signal to be processed. The required task such as prediction and classification is performed by the output layer. An arbitrary number of hidden layers that are placed in between the input and output layer are the true computational engine of the MLP. This algorithm is designed to approximate any continuous function and can solve problems which are not linearly separable. The major use cases of MLP are pattern classification, recognition, prediction and approximation.

Chapter 3

Related Work

This chapter presents methods in two relevant areas, semantic similarity and knowledge graph embeddings, and overviews the field of gene-disease association prediction with a focus on ontology and knowledge-graph based approaches.

3.1 Semantic Similarity

A semantic similarity measure is a function that, given two ontology classes or two sets of classes annotating two entities, returns a numerical value reflecting the closeness in meaning between them (Pesquita et al., 2009). The meaning of the classes being compared is automatically extracted from the ontologies. In the case of Human Phenotype Ontology (Köhler et al., 2021) and HP annotations, semantic similarity can be calculated between two entities each annotated with a set of HP classes, for instance calculating the similarity between two diseases through their phenotypes, two genes or even a gene and a disease. In other words, when biological entities are described using a common schema provided by an ontology, they can be compared by means of their semantic annotations.

Semantic similarity measures can be used as unsupervised methods for association prediction based on a threshold (Cáceres and Paccanaro, 2019; Wu et al., 2008), as features in supervised learning models (Sousa et al., 2020) or in clustering algorithms (Sun et al., 2011). Ontology-based similarity measures have been applied to a variety of prediction processes such as protein-protein interactions (Maetschke et al., 2011; Liu et al., 2018), gene-disease associations (Liu et al., 2018; Li et al., 2014) and is also useful to diagnose patients, determining sequence similarity, or evaluating computational methods, which predict ontology class annotations (Kulmanov et al., 2020b).

Most state-of-the-art methods are categorized as taxonomic (hierarchical) semantic similarity. Taxonomic semantic similarity also commonly known as ontology-based semantic similarity, compares ontology entities based on the taxonomic relations within the ontology graph (d'Amato et al., 2008).

The approaches used to quantify semantic similarity can be distinguished based on

which entities they intend to compare: approaches for comparing two classes and approaches for comparing two entities annotated with its own set of classes (Pesquita et al., 2009).

3.1.1 Semantic Similarity for Classes

For comparing classes there are two types of approaches:

- **Edge-based:** rely on counting the number of edges in the graph path between two classes. The most common technique is by calculating the *distance* where selects either the shortest path or average of all paths when more than one path exists. Other possible technique is the *common path*, through the length of the lowest common ancestor of the two classes to the root node. These approaches assume that the distance between all the relationships in an ontology is constant or depth-dependent. This assumption is not valid in existing biomedical ontologies so edge-based measures are rarely used in the biomedical domain.

- **Node-based:** depend on comparing the properties of the classes involved, which can be related to the classes themselves, their ancestors, or their descendants. They typically rely on the Information Content (IC) of a class which gives a measure of how informative or, rather, specific a class is. The IC can be calculated through the graph structure (intrinsic approach) or the number of annotations a class is used on (extrinsic approach).

Node-based measures are more commonly employed, since they do not suffer the limitations of edge-based methods. A crucial aspect of node-based measures is the calculation of information content. Two popular IC measures are Resnik (Resnik, 1995) and Seco (Seco et al., 2004).

ICResnik is a corpus-based approach to compute information content proposed by Resnik (1995) and based on the number of entities annotated with class h in the knowledge graph, which is given by:

$$IC_{Resnik}(h) = -log\ p(h) \tag{3.1}$$

where $p(h)$ is the probability of annotation in the corpus.

ICSeco, proposed by Seco et al. (2004), is a structure-based approach based on the number of direct and indirect children from a class h and is given by:

$$IC_{Seco}(h) = 1 - \frac{log[hypo(h)] + 1}{log[maxnodes]} \tag{3.2}$$

where $hypo(h)$ is the number of direct and indirect children from class h, including class h, and maxnodes is the total number of classes in the ontology.

A popular class similarity measure is Resnik's similarity (Resnik, 1995) in which the similarity between two classes corresponds to the Information Content of their most informative common ancestor:

$$sim(e1, e2) = max\{IC(e)) : \ e \ \epsilon\{A(e1) \cap A(e2)\}\} \tag{3.3}$$

where e is a class in $A(e_i)$, the set of ancestors of e_i.

3.1.2 Semantic similarity for Entities

Calculating semantic similarity for two entities, each annotated with a set of classes, usually uses one of two approaches:

- **Pairwise:** measure functional similarity between two entities by combining the semantic similarities between their classes.
- **Groupwise:** set, vector or graph-based measures are employed. In set measures only direct annotations. In graph approaches entities are represented as the subgraphs of the ontology corresponding to all their annotations (direct and inherited). In vector approaches an entity is represented in vector space, with each class corresponding to a dimension.

Both pairwise and groupwise measures have been shown to work well in different scenarios (Pesquita et al., 2009).

Popular pairwise measures are the Best-Match Average (BMA) and the Maximum (Max). BMA and Max are pairwise approaches that can work with class-based measures. The Max simply takes the maximum similarity between all annotating classes of two entities.

$$Max(e1, e2) = max\{sim(e1, e2) : \ h_1 \ \epsilon \ HP(e_1), \ h_2 \ \epsilon \ HP(e_2)\} \tag{3.4}$$

BMA considers the best scoring pairs of classes from each entity.

$$BMA(e1, e2) = \frac{\Sigma_{h_1 \epsilon HP(e1)} sim(h1, h2)}{2|HP(e1)|} + \frac{\Sigma_{h_2 \epsilon HP(e2)} sim(h1, h2)}{2|HP(e2)|} \tag{3.5}$$

where $HP(ei)$ is the number of annotations for entity ei and $sim(e1, e2)$ is the semantic similarity between the HP class $h1$ and HP class $h2$

SimGIC is a popular groupwise measure proposed by Pesquita et al. (2007) which resorts to a Jaccard similarity, in which each HP term is weighted by its IC and given by

$$simGIG(e1, e2) = \frac{\Sigma_{h \epsilon \{HP(e1) \cap HP(e2)\}} IC(h)}{\Sigma_{h \epsilon \{HP(e1) \cup HP(e2)\}} IC(h)} \tag{3.6}$$

where $HP(h_i)$ is the set of annotations (direct and inherited) for entity e_i.

3.2 Knowledge Graph Embeddings

An embedding is a vector representation resulting from the use of semantic information mapping techniques (Ristoski and Paulheim, 2016b). Knowledge graph embeddings aim to represent represent concepts and relationships in a graph as low dimensional vectors while the graph structures are preserved. Knowledge graph embeddings can be used as features for machine learning, but they can also support similarity computation through vector similarity operations such as the cosine similarity.

There are a variety of methods for building knowledge graph embeddings (Cai et al., 2018). While some focus on exploring the knowledge graph facts solely, others also include additional information, such as entity types, relation paths, axioms, and rules or textual information, and more recently, path-based approaches have been proposed by transforming the knowledge graph into node sequences (Ristoski and Paulheim, 2016a).

The categorization of knowledge graph embedding methods is still not widely agreed upon, with different works recognizing different categories. While Wang et al. (2017) considers matrix-factorization, translational distance, semantic matching, deep learning-based, Makarov et al. (2021) considers matrix factorization, deep learning and random walks. Table 3.1 overviews all categories and provides examples of works from each of them. However, the random walk category was expanded to include path-based methods that do not employ random walks, and renamed accordingly.

The following sections briefly explain each of the categories characteristics and provide a summary of representative approaches for each category.

3.2.1 Matrix Factorization

In the early 2000s, graph embedding methods were mostly designed to reduce the high dimensionality of the non-relational data by assuming the data lie in a low dimensional manifold, in other words, were mainly matrix factorization based. They represent a graph property (e.g., pairwise node similarity, node transition probability matrix, etc.) in the form of a matrix and factorize this matrix to obtain an embedding (Cai et al., 2018). Locally linear embedding (Roweis and Saul, 2000) and IsoMap (Tenenbaum et al., 2000) were the pioneering studies.

Locally linear embedding learns node similarity by reconstructing weights matrix W with which neighboring nodes affect each other and the possibly attributed features X,:

$$\left\|X - W^T U\right\|_2^2 \tag{3.7}$$

and repeats that procedure to learn manifold U with achieved matrix W (Makarov et al., 2021).

The IsoMap algorithm is based on graph Laplacian eigenmaps. IsoMap finds the

Table 3.1: Summary of the representative methods. In bold the methods used in this work.

Category	Method	Reference
Matrix Factorization	Locally linear embedding	Roweis and Saul (2000)
	IsoMap	Tenenbaum et al. (2000)
Translational Distance	**TransE**	Bordes et al. (2013)
	TransH	Wang et al. (2014)
	TransR	Lin et al. (2015)
	TransD	Ji et al. (2015)
	TranSparse	Ji et al. (2016)
	KG2E	He et al. (2015)
Semantic Matching	RESCAL	Nickel et al. (2011)
	DistMult	Yang et al. (2015)
	HolE	Nickel et al. (2016b)
	CompleEx	Trouillon et al. (2016)
Deep learning	SDNE	Wang et al. (2016)
	DNGR	Cao et al. (2016)
Path-based	DeepWalk	Perozzi et al. (2014)
	Node2Vec	Grover and Leskovec (2016)
	RDF2Vec	Ristoski and Paulheim (2016a)
	Onto2Vec	Smaili et al. (2018)
	OPA2Vec	Smaili et al. (2019)
	OWL2Vec	Chen et al. (2021)

shortest path between two nodes and applies Metric Multidimensional Scaling by incorporating the geodesic distances imposed by a weighted graph (Makarov et al., 2021).

IsoMap and locally linear embedding were proposed to model global structure while preserving local distances or sampling from the local neighborhood of nodes. The lower bound for methods complexity was quadratic in the number of vertices being inappropriate for large networks.

3.2.2 Translational Distance

Translational Distance embeddings methods exploit distance-based scoring functions being that each fact represents the distance between the two entities, usually after a translation carried out by the relation (Wang et al., 2017). The considered methods were TransE (Bordes et al., 2013), TransH (Wang et al., 2014), TransR (Lin et al., 2015), TransD (Ji et al., 2015), TranSparse (Ji et al., 2016) and KG2E (He et al., 2015).

TransE is the most representative translational distance model by representing both entities and relations as vectors in the same space.

Given a 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 $h + r \simeq t$ when (h, r, t) holds. The

scoring function is then defined as the (negative) distance between $h + r$ and t:

$$f_r(h, t) = - \|h + r - t\|_{\frac{1}{2}} \tag{3.8}$$

Despite its simplicity and efficiency, a drawback of TransE is that it cannot deal well with one-to-many, many-to-one and many-to-many relations. To address this challenge, there are other extensions like TransH and TransR. Figure 3.1 (Wang et al., 2017) gives a comparison between TransE, TransH, and TransR.

TransH introduces a hyperplane for each relation r (relation-specific hyperplane) and projects h and t into the hyperplane. TransH models entities again as vectors, but each relation r as a vector r on a hyperplane with w_r as the normal vector.

TransR shares a very similar idea with TransH but introduces a space for each relation r (relation-specific space), rather than hyperplanes. In TransR, entities are represented as vectors in an entity space, and each relation is associated with a specific space and modeled as a translation vector in that space.

TransD and TranSparse simplify TransR: TransD simplifies by further decomposing the projection matrix into a product of two vectors; TranSparse simplifies by enforcing sparseness on the projection matrix.

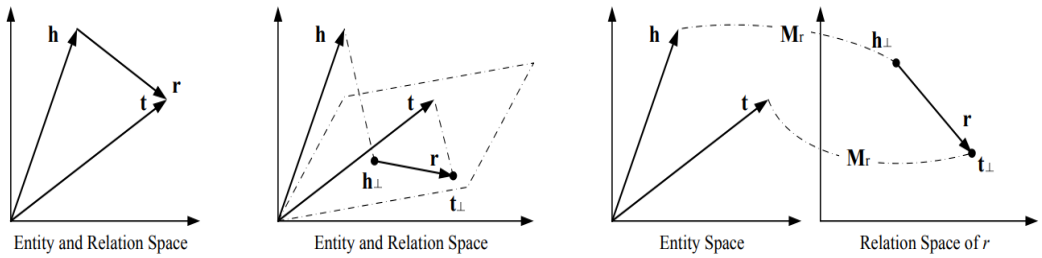

Figure 3.1: Simple illustrations of TransE, TransH, and TransR extracted from Wang et al. (2017).

The five methods introduced model entities and relations as deterministic points in vector spaces. Other works take into account their uncertainties, and model them as random variables. For instance, KG2E uses multivariate Gaussian distributions to drawn vectors to represents entities and relations.

3.2.3 Semantic Matching

Semantic Matching embedding methods exploit similarity-based scoring functions by matching latent semantics of entities and relations embodied in their vector space representations (Wang et al., 2017; Su et al., 2020). Table 3.2 shows the scoring functions of the four semantic matching approaches.

Table 3.2: Scoring functions for each semantic matching approach.

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

RESCAL was proposed based on the idea that entities are similar if connected to similar entities via similar relations. By associating each relation r with a matrix M_r, it defines the energy function by a bilinear model

$$(h, r, t) = h^t M_r t \tag{3.9}$$

where h, $t \; \epsilon \; R^d$ are d-dimensional $(d \gg n)$ embedding vectors for entities h and t, respectively. RESCAL jointly learns embedding results for entities by h and t and for relation by M_r.

DistMult simplifies RESCAL by restricting matrix M_r for relation r as a diagonal matrix. Though DistMult is more efficient than RESCAL, it can only deal with the undirected networks. To address this problem, HolE composes h and t by their circular correlation. Consequently, the power of RESCAL and simplicity of DistMult are inherited by HolE. Figure 3.2 (Wang et al., 2017) provides a comparison between these three approaches.

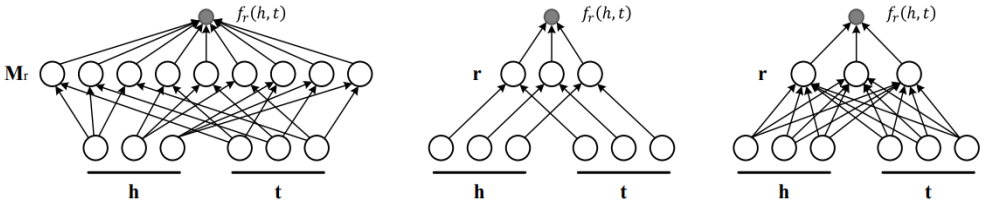

Figure 3.2: Simple illustrations of RESCAL, DistMult and HolE extracted from Wang et al. (2017).

ComplEx extends DistMult by introducing complex-valued embeddings to better handle various asymmetric relations. In ComplEx, entity and relation embeddings h, r, t no longer lie in a real space but in a complex space instead.

3.2.4 Path-based

Motivated by drawbacks of the matrix factorization approach (section 3.2.1), another approach emerged that attempts to preserve local neighborhoods of nodes and their properties based on paths (random and non-random walks) (Makarov et al., 2021).

In graph theory, random walks can be explored to capture structural relationships between nodes (Su et al., 2020). A graph is transformed into node sequences by performing

truncated random walks, which preserve the structural proximity of the network. After representing a graph as a set of random walk paths sampled from it, natural language methods, such as Word2vec (Mikolov et al., 2013), can be applied to the sampled paths for graph embedding, which preserves graph properties carried by the paths.

DeepWalk utilizes short random walks to extract information from a graph by generating a sequence of vertices corresponding to a sentence in natural language. Specially, they adopt the SkipGram, a famous deep model for neuro-linguistic programming, that embeds words into a low dimensional space by incorporating the context of words in sentences. Finally, DeepWalk utilizes hierarchical softmax to reduce computational complexity by transforming the nodes into a huffman tree (Hou et al., 2020).

Node2Vec explores the original graph through 'biased' random walks and therefore can force walks to remain within a certain distance of the origin node or explore further away (Kulmanov et al., 2020a). Similar to the DeepWalk, Node2Vec turns the embedding problem into maximizing the probability of finding the co-occurrence neighbor vertices by utilizing the SkipGram. Also, the negative sampling method is leveraged to solve the high computational complexity by regarding the neighborhood nodes 'negative sampling'.

Compared to DeepWalk, Node2Vec (Grover and Leskovec, 2016) introduces a more flexible random walk strategy. In Figure 3.3 is depicted an overview of the framework of these two approaches (Hou et al., 2020).

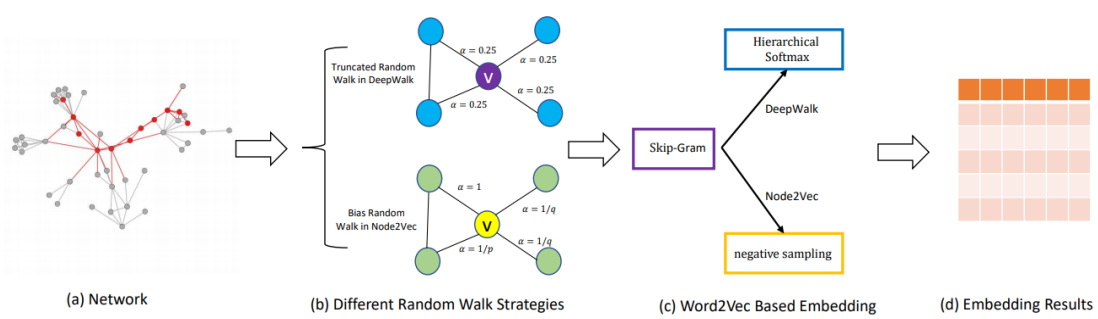

Figure 3.3: Framework of DeepWalk and Node2Vec extracted from Hou et al. (2020).

RDF2Vec (Ristoski and Paulheim, 2016a) adapts Node2Vec strategy to RDF graphs. Unlike DeepWalk and Node2Vec, that are defined for graphs with just one type of edges, RDF2vec has been tailored to RDF graphs by respecting the type of edges, enriching the learning approach's semantics. This method also relies on Word2Vec, used for producing the embeddings.

OWL2Vec (Chen et al., 2021) computes embeddings for ontologies by projecting the ontology axioms into a graph and performing random walks over the ontology graph to create a corpus of sentences. This corpus is then given to Word2Vec language model, to create concept embeddings. It encodes the semantics of an OWL ontology by taking into

account its graph structure, lexical information and logical constructors.

Moving on to non-random walk methods, Onto2Vec (Smaili et al., 2018) also uses language modeling approaches, generating embeddings for ontology classes and instances taking into account the logical axioms that define the semantics of ontology classes. It takes an ontology as input, uses a reasoner to infer additional logical axioms, mainly subclass axioms between named classes, then treats each asserted or inferred axiom as a sentence and embeds the set of axioms using the Word2Vec language model. This allows Onto2Vec to embed ontologies directly based on their axioms while considering all axiom types, no matter how complex they are (Kulmanov et al., 2020a).

OPA2Vec (Ontologies Plus Annotations to Vectors) (Smaili et al., 2019) extends the Onto2Vec method to include logical axioms and annotation properties. Annotation properties in biomedical ontologies provide labels, synonyms, definitions, and other types of information about classes and instances in ontologies. It combines the corpus generated from the asserted and inferred logical axioms in Onto2Vec with a corpus generated from all or selected annotation properties. Then applies a Word2Vec skipgram model on the combined corpus to generate vector representations of all entities in the ontology. In particular, a pre-train of the Word2Vec model occurs on all PubMed abstracts so that natural language words are assigned a semantics (and vector representation) based on their use in biomedical literature.

3.2.5 Deep Learning-based

Over the past years, deep learning methods have shown impressive improvement across diverse domains (Cai et al., 2018; Makarov et al., 2021). Due to its robustness and effectiveness, deep learning has been widely used in graph embedding.

A deep autoencoder is a deep learning algorithm that constitutes of two symmetrical deep-belief networks, autoencoder and decoder, with four or five shallow layers, as depicted in Figure 3.4 (Abirami and Chitra, 2020). The autoencoders belong to the neural network family. The aim of an autoencoder is to learn a lower-dimensional representation for a higher dimensional data, typically for dimensionality reduction. Both encoder and decoder contain multiple non-linear functions. The encoder compresses the input data into a representation space and the decoder reconstructs the data back from its encoded form (reconstruction space).

The idea of adopting autoencoder for graph embedding is similar to matrix factorization (section 3.2.1) in terms of neighbourhood preservation. Specifically, the adjacency matrix captures a node's neighbourhood. If the autoencoder's input is the adjacency matrix, the reconstruction process will make the nodes with similar neighbourhood have similar embedding.

SDNE (Wang et al., 2016) and DNGR (Cao et al., 2016) use deep autoencoder to capture non-linearity in graphs and simultaneously apply dimension reduction for construct-

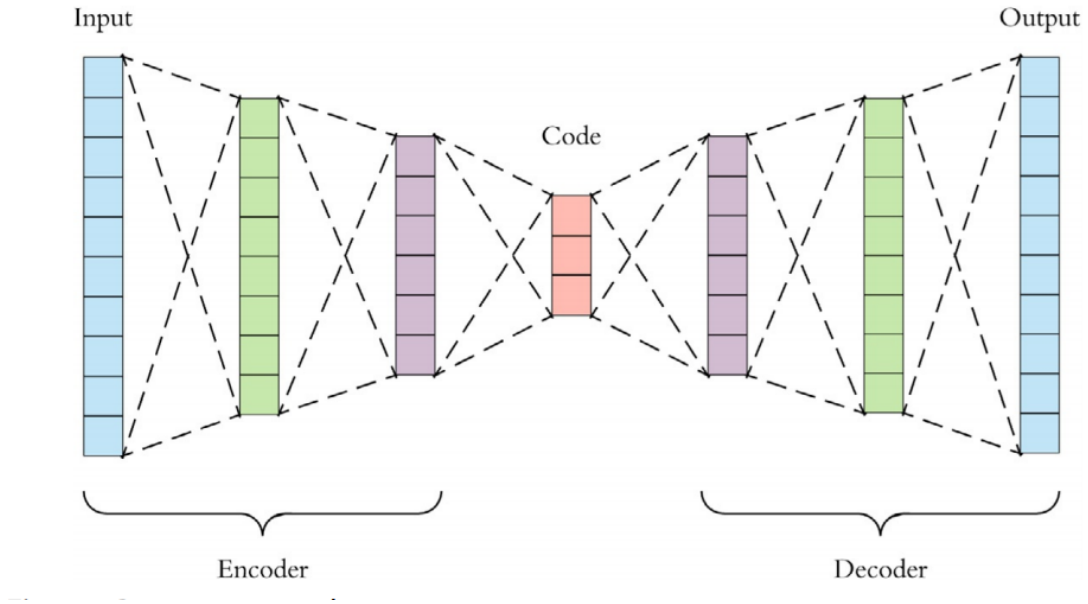

Figure 3.4: Autoencoder architecture as a whole extracted from Abirami and Chitra (2020).

ing graph embedding. Both methods use global information and thus are not appropriate for large networks. SDNE use autoencoder preserving first order proximity and Laplacian Eigenmaps for penalizing long distances for embedding vectors of similar vertices. DNGR uses stacked denoising autoencoders over positive pointwise mutual information matrix obtained from similarity information based on random surfing. The random surfing model is inspired by the PageRank model (Chebolu and Melsted, 2008).

3.3 Gene-Disease Prediction

The past decades have witnessed the development of several algorithms to predict gene-disease associations. In this work, we categorize them into two types of approaches: those that do not employ ontologies, and those that do.

3.3.1 Non Ontology-Based approaches

The majority of gene-disease prediction methods train a machine learning classifier with various types of features extracted from different kinds of data (Mordelet and Vert, 2011; Yang et al., 2012; Singh-Blom et al., 2013; Luo et al., 2019b,a). Since the features are collected for genes, these algorithms are usually single task, which means they can only predict disease genes for a specific disease. Thus, these approaches struggle with diseases that have few or no known associated genes, because the number of the genes would be too small to train the model. Moreover, the relationships between diseases are generally not used in the prediction since only one disease is considered at a time.

Matrix completion methods address these issues by jointly predicting gene-disease associations and leveraging disease similarities during calculation (Zeng et al., 2017a). Natarajan and Dhillon (2014) collected the gene–phenotype association's studies in eight species with phenotypes from Online Mendelian Inheritance in Man (OMIM) and if two genes have similar phenotype features, it indicates two genes are associated with a similar set of phenotypes across different species. Zeng et al. (2017b) also used phenotype–phenotype similarity to prioritize novel gene–phenotype associations. However, these methods generally have difficulties in finding a global optimal solution and can take a very long time to converge.

There are also studies that have tackled this challenge via a network-based approach, motivated by the observation that genes causing the same or similar diseases tend to lie close to one another in a network of protein-protein or functional interactions. Centrality indices, random walk, and network energy are used in many methods to predict disease-gene associations (Köhler et al., 2008; Chen et al., 2014a,b). Although most network-based methods are unaffected by the lack of known genes for a particular disease, their performance is strongly affected by the quality of networks, and they generally perform worse than machine learning-based methods on diseases with many known associated genes (Chen et al., 2015, 2016).

3.3.2 Ontology-Based approaches

Table 3.3 provides an overview of several methods that explore ontologies or knowledge graphs to predict gene-disease associations.

There are works in this area that use only one ontology and take advantage of classical semantic similarity measures such as Asif et al. (2018). This author showed that machine learning classifiers trained on gene functional similarities, using Gene Ontology, can improve the identification of genes involved in complex diseases as was applied to autism spectrum disorder.

We also encounter methods that for the learning process use several types of networks as data. Luo et al. (2019c) proposed dgManifold to predict disease-gene associations with manifold learning regularized by two similarity networks: gene similarity calculated based on the Gene Ontology and disease similarity based on Human Phenotype Ontology. This method was evaluated in Lung Cancer and Bladder Cancer. Vanunu et al. (2010) presented a novel network-based approach for predicting causal genes and protein complexes that are involved in a disease of interest (Prostate Cancer, Alzheimer or type 2 Diabetes Mellitus) and explored the Gene Ontology for manually annotated protein complexes. Both these authors applied their method to analyze disease-gene association data from the OMIM knowledgebase.

Network fusion algorithms combine different sources of information on both genes and diseases and provide a universal ranking of associations for any disease gene pairs.

Table 3.3: Summary of the existing work on ontology-based approaches.The abbreviations used in this table are defined at the beginning of the document (acronyms).

Reference	Ontology	Data	Task	Method
Vanunu et al. (2010)	GO	OMIM	Single disease (prostate cancer, alzheimer and type 2 diabetes mellitus)	Similarity networks
Robinson et al. (2014)	HP, MPO	OMIM	Mutiple diseases	Taxonomic SS
Alshahrani et al. (2017)	GO, HP, DO	DisGeNET	Multiple diseases	KGE
Asif et al. (2018)	GO	SFAR Gene Database	Single disease (ASD)	Taxonomic SS
Zakeri et al. (2018)	GO, HP	OMIM	Single disease (Diseases of the nervous system)	Matrix Factorization
Luo et al. (2019c)	GO, HP	OMIM	Single disease (Lung Cancer and Bladder Cancer)	Similarity networks
Smaili et al. (2019)	PhenomeNET	MGI Database	Multiple diseases	KGE
Shu et al. (2021)	Human DO	OMIM	Multiple diseases	Taxonomic SS

Robinson et al. (2014) developed a cross-species analysis approach that allows computational reasoning where a phenotypic relevance score is calculated based on the semantic similarity of human disease (annotations from the Human Phenotype Ontology) and the phenotypic manifestations observed in a mouse model (annotations from the Mammalian Phenotype Ontology (MPO)).

Another type of approach implementing the same intuition tries to model this problem as a recommender system, in which diseases and genes represent customers and products, respectively. Zakeri et al. (2018) presents a gene prioritization method that can innovatively not only integrate data sources describing genes like Gene Ontology, but also data sources describing Human Phenotype Ontology classes. The proposed method offers promising results on several types of diseases including diseases of the nervous system.

More recently, the limitations of semantic similarity-based approaches have begun to be tackled by more sophisticated approaches based on knowledge-graph embeddings.

Recent advances in knowledge graph embeddings such as OPA2Vec (Smaili et al., 2019), an extension of Onto2Vec (Smaili et al., 2018), generate vector representations of biological entities in ontologies by combining formal ontology axioms and annotation axioms from the ontology metadata and applied the approach to a single ontology, PhenomeNET (Hoehndorf et al., 2011; Rodríguez-García et al., 2016). This ontology is a system for prioritizing candidate disease genes based on the phenotype similarity between a disease and a database of genotype-phenotype associations.

Alshahrani et al. (2017) employed knowledge graph embeddings over a knowledge graph based on three ontologies, the Gene Ontology, the Human Phenotype Ontology, and the Disease Ontology (DO). This approach also utilizes structured data sources such as human protein interactions, protein-chemical interactions, drug side effects and gene-disease associations. It applies automated reasoning to enrich the graph with inferred relations and employs a random-walk embedding approach. The application of this work to gene-disease association prediction presents some challenges regarding data leakage. DisGeNet (Piñero et al., 2019) includes gene-disease associations extracted from multiple sources including OMIM and OrphaNet, which are the same sources used to create some of the HP annotations.

Overall, these works are limited in their use of the ontologies, because they mostly employ a single ontology, and when they employ more than one, they are included in the graph without considering semantic links between them.

Chapter 4

Methodology

This chapter gives a detailed explanation of the proposed methodology and is organized as follows. Section 4.1 gives a general overview of the several steps into our approach. Section 4.2 explains how the gene-disease associations were chosen to create a final dataset to analyze our methodology (section 4.2.1) but also what ontologies and annotations were used to enrich the knowledge graphs (section 4.2.2). Sections 4.3 to 4.6 provide a closer look into the methodology showing how the prediction was realized and evaluated but also how the baseline of comparison was created.

4.1 Overview

The methodology proposed in this work can be divided into four main steps as depicted in Figure 4.1. The first step in the approach is to integrate the different ontologies and annotation data to build the knowledge graph. In a second step, the embeddings that represent the gene and the disease according to their annotations in different knowledge graphs are created. In a third step, these embeddings are combined using different vector operators producing a representation of genes and diseases in what is effectively a shared semantic space. Finally, in a fourth step, supervised learning algorithms are trained over the combined embeddings to predict gene-disease associations. This approach is evaluated against non-machine learning approaches based on classical semantic similarity measures (baseline) and knowledge graph embedding similarity.

4.2 Data

4.2.1 Gene-Disease associations

Obtained 84038 curated gene-disease associations (dated july 2020) from DisGeNET - a discovery platform that contains a comprehensive catalog of genes and variants associated with human diseases (Piñero et al., 2019). These pairs were formed from 9703 genes and

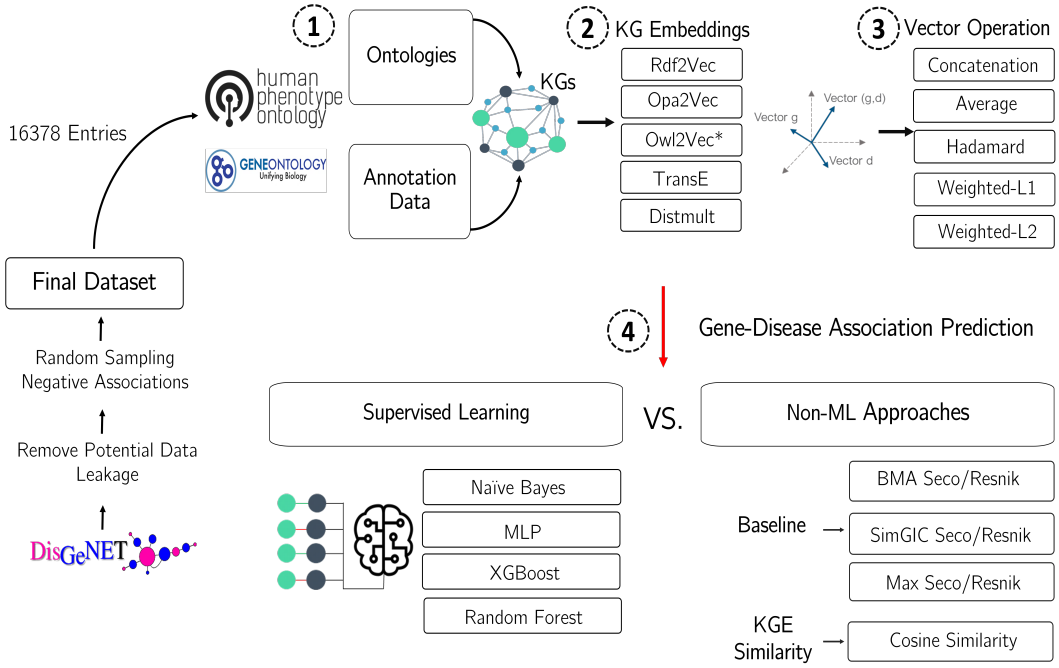

Figure 4.1: Overview of the methodology with four basic steps: 1) build the KG with ontologies and annotations; 2) create embeddings to represent each gene and disease; 3) produce a final vector of the pairs in the dataset; 4) gene-disease association prediction. Created in BioRender.com

11181 diseases, being that the medium number of genes by each disease were two as well as the medium number of diseases by each gene.

The goal of predictive modeling is to develop a model that makes accurate predictions on new data, unseen during training. However, there are problems, such as data leakage, that can create overly optimistic if not completely invalid predictive models. Data leakage is when information from outside the training dataset is used to create the model (Kaufman et al., 2011). This additional information can allow the model to learn or know something that it otherwise would not know and in turn invalidate the estimated performance of the model being constructed. To avoid this problem the original pairs from DisGeNET suffered a process of filtration and only associations whose original source reporting the gene-disease association did not rely on the databases Uniprot (Consortium, 2019), OMIM (Amberger et al., 2014), or Orphanet (Wakap et al., 2019) was chosen. The reason for this decision is because in the process of constructing the knowledge graph the annotation data for the pairs (presented in section 4.2.2) also rely on these sources and would have enhanced the performance. After this process remained a total of 73469 pairs, composed of 8545 genes and 6490 diseases.

To have accurate annotations for all the pairs, they were filtered and excluded if they did not correspond to one of these three criteria:

(i) the genes must have correspondence with a protein from Uniprot that is annotated

with GO classes;

(ii) the genes must be annotated with HP classes;

(iii) the diseases must be annotated with HP classes.

A total of 2716 genes, 1807 diseases, and 8189 gene-disease associations remained in the dataset. Taking into consideration that that negative samples are not included in DisGeNET, we employed a random sampling method to create negative examples composed of the genes and diseases present in the positive examples, but without known associations between them, building a final balanced dataset with 16378 entries.

4.2.2 Ontologies and Knowledge Graphs

The knowledge graphs used in this work (described in section 4.3) are composed of one or two ontologies and their associated annotations. The process of selecting the ontologies prioritizes the capacity to provide formal knowledge for the pairs in the final dataset to enrich the final knowledge graph. Two of the most popular biomedical ontologies were chosen: Human Phenotype Ontology and Gene Ontology. In Table 4.1, we summarized the information regarding the ontologies and annotations used.

Table 4.1: Number of classes, branches and annotation data for the two ontologies.

	Human Phenotype Ontology	**Gene Ontology**
Classes	15340	44117
Branches	Phenotypic Abnormality: 15149 Mode of Inheritance: 31 Frequency: 6 Clinical Course: 48 Clinical Modifier: 106	Cellular Component: 4185 Biological Process: 28769 Molecular Function: 11163
Logical Definitions	350	—
Gene Annotations	136068	76161
Disease Annotations	40583	—

The Human Phenotype Ontology provides comprehensive bioinformatic resources for the analysis of human diseases and phenotypes, offering a computational bridge between genome biology and clinical medicine (Köhler et al., 2021). It is organized as independent subontologies that cover different categories, being “Phenotypic Abnormality” the largest one. The ”Mode of inheritance” describes the relationship between patients or diseases and their symptoms. The ”Mortality/Aging” similarly allows the age of death typically associated with a disease or observed in a specific individual to be annotated. Finally,

"Clinical Modifier" is designed to characterize and specify the phenotypic abnormalities defined in the "Phenotypic Abnormality" subontology (Köhler et al., 2019). The main domain application of the Human Phenotype Ontology has, to date, been on rare disorders, and has in the past provided a large corpus of disease-HP annotation profiles using OMIM, Orphanet and DECIPHER for disease entities (Köhler et al., 2014). The Human Phenotype Ontology can be used to annotate both patients, diseases, or human genes. In the latter case, all phenotype classes associated with any disease that is associated with variants in a gene are assigned to that gene.

In terms of annotations data, the Human Phenotype Ontology graph (dated October 2020) was collected from the Human Phenotype Ontology website[1] in OWL format and contains 27391 ontology classes. The HP annotations were downloaded from the Human Phenotype Ontology website in a Tab-separated Values (TSV) file (dated October 2020), providing links between genes or diseases to HP classes (Köhler et al., 2021).

Gene Ontology is the most successful case of the use of an ontology in biomedical research and it is used for the annotation of gene products. All functional knowledge is structured and represented in a form amenable to computational analysis, which is essential to support modern biological research. This ontology is structured using a formal ontology, by defining classes of gene functions (GO classes) that have specified relations to each other (Consortium, 2020). It covers three distinct aspects of gene function:

- Cellular Component (CC) - refers to the cellular location where a gene product is active.
- Biological Process (BP) - refers to a biological objective to which the gene or gene product contributes.
- Molecular Function (MF) - is defined as the biochemical activity (including specific binding to ligands or structures) of a gene product.

The Gene Ontology graph was collected from the Gene Ontology website[2] (dated December 2020) in OWL format and contains 44117 ontology classes subdivided into 4185 CC classes, 28769 BP classes, and 11163 MF classes. The annotations were downloaded from the Gene Ontology Annotation (GOA) database (dated 11 August 2020) for the human species (Huntley et al., 2014) in Gene Association File (GAF) 2.1 format. These annotations link Uniprot (Consortium, 2019) identifiers for proteins with GO classes describing them. The genes, identified by their Entrez Gene Code, are associated with the proteins and, the final annotations provide links between the genes and the GO classes.

[1] https://hpo.jax.org/app/
[2] http://geneontology.org/

4.3 Knowledge Graph Integration

The essence of a knowledge graph is the entities, the ontologies, and associated links between them. To build the different knowledge graphs employed in this work, several strategies needed to be employed due to the different inputs accepted by the embeddings implementations. A first step was to merge the Human Phenotype Ontology and Gene Ontology through a common virtual root and save into an OWL file because both OWL2Vec and OPA2Vec methods only accepted a single ontology file with a single root. This was achieved by using the RDFlib [3] library. RDFLib contains parsers for most of the known RDF serializations, including RDF/XML (OWL). For RDF2Vec, DistMult, and TransE, both ontologies graphs also needed to be integrated into the same file. In terms of the entities annotations, only OPA2Vec is able to process annotation files separately from the ontology graph. For RDF2Vec, DistMult, and TransE, the annotations were parsed and integrated using RDFlib, whereas for OWL2Vec, the OWLready2 package (Lamy, 2017) was employed. Owlready2 is a package for ontology-oriented programming in Python 3, that can load, modify, and save ontologies but also manage knowledge graphs.

The final knowledge graphs used in this work are divided into five possible types:

(i) **HP-simple:** composed by HP without logical definitions and HP annotations both for genes and diseases;

(ii) **HP-full:** composed by HP with all logical definitions and HP annotations both for genes and diseases;

(iii) **HP-simple + GO:** composed by HP without logical definitions, HP annotations (for genes and diseases), GO and GO annotations (for genes). HP and GO are integrated through a common virtual root;

(iv) **HP-full + GO:** composed by HP with all logical definitions, HP annotations (for genes and diseases), GO and GO annotations (for genes). HP and GO are integrated through a common virtual root;

(v) **HP-simple + LD + GO:** composed by HP with specifically added logical definitions, HP annotations (for genes and diseases), GO and GO annotations (for genes). HP and GO are also integrated through a common virtual root;

Regarding the **HP-simple + LD + GO**, to simplify the graph embeddings approach, as seen in the example of Figure 4.2, the existing logical definitions are simplified to a more direct relation between the HP class and GO class through an equivalent class statement reaching a total of 350 links. This allows for the path to be shorter and direct for random-walk based methods and for the other methods to extract a single triple with

[3] https://github.com/RDFLib/rdflib; https://rdflib.readthedocs.io/en/stable/

the necessary information to enrich the knowledge graph. The owlready2 was the package used to facilitate this process.

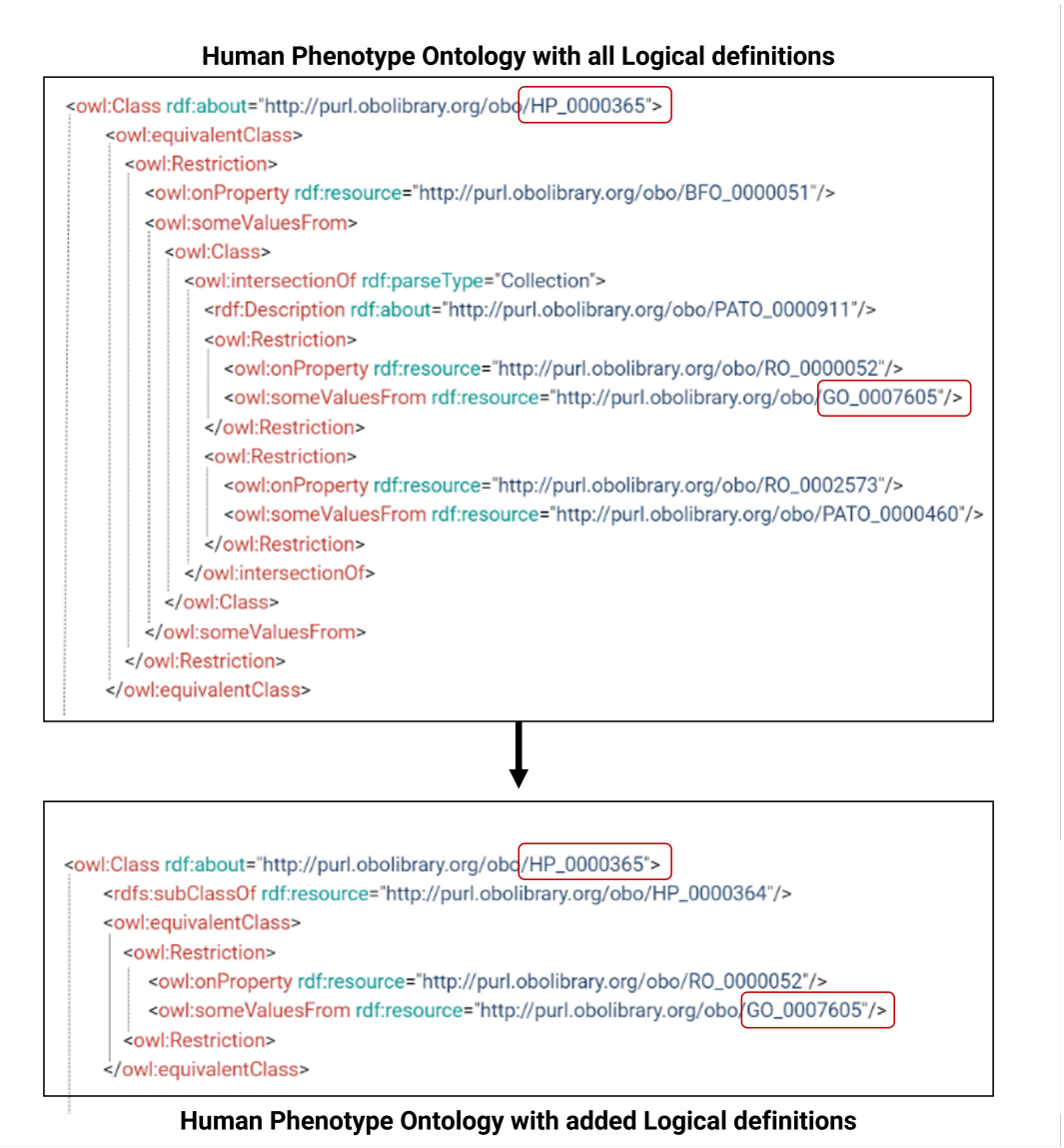

Figure 4.2: LD Simplification Process Example. The HP term for “Hearing impairment” (HP:0000365) is equivalent to a restriction that involves the GO term “Sensory perception of sound” (GO:0007605). A possible simplification is to create a direct relation with an equivalent class statement between the two classes.

4.4 Knowledge Graph Embeddings and Representation

In this work, five methods of knowledge graph embeddings were used to learn feature vectors for the different knowledge graphs, described in section 4.3, and create a representation of two distinct vectors for each gene-disease pair of the dataset. The embed-

dings present 200 features and cover three types of popular knowledge graph embedding approaches (AppendixA for default parameters):

- **Translational Distance**: TransE[4] (Bordes et al., 2013) with default parameters;
- **Semantic Matching**: DistMult[4](Yang et al., 2015) with default parameters;
- **Path-based:**

  Random Walk:

  -RDF2Vec[5] (Ristoski and Paulheim, 2016a) with sequences generated using the Weisfeiler-Lehman algorithm with walks depth 8 and a limited number of 500 by entity. The corpora of sequences were used to build a Skip-Gram model with the default parameters for Word2Vec;

  - OWL2Vec*[6] (Chen et al., 2021) with the same parameters used with RDF2Vec.

  Non-Random Walk:

  - OPA2Vec[7] (Smaili et al., 2019) with default parameters;

We present one example of each type of approach of knowledge graph embedding, except the case of path-based approaches that are explored in more depth with different methods. Given that we are using data where the core is an ontology, not the instances, and that it has no relations between instances but only relations between the instances and the concepts that describe them in the ontologies, it will be necessary to capture relations at a greater distance. This means relationships between entities and the various concepts that describe them in the ontology's hierarchy. It is thus expected that path-based methods have the potential to work better because they can capture longer distance relations. Moreover, OPA2Vec also explores embeddings of the textual component of the ontologies, which are a defining feature of biomedical ontologies.

After the knowledge graph embeddings methods, each gene-disease pair corresponds to two vectors, $f_i(g)$ and $f_i(d)$, associated with a gene and a disease, respectively. We define a binary operator over the corresponding feature vectors g and d in order to generate a representation $r(g, d)$ such that r : $V \times V \longrightarrow \mathbb{R}^{d'}$ where d' is the representation size for the pair (g, d). Several choices for the binary operator were considered from a set of commonly employed operators with knowledge graph embeddings (Grover and Leskovec, 2016). The chosen operators are summarized in Table 4.2.

[4]https://github.com/thunlp/OpenKE
[5]https://github.com/IBCNServices/pyRDF2Vec
[6]https://github.com/KRR-Oxford/OWL2Vec-Star
[7]https://github.com/bio-ontology-research-group/opa2vec

Table 4.2: Choice of binary operators.

Operator	Definition
Concatenation	$f_i(g) \mid\mid g_i(d)$
Average	$\frac{f_i(g) + g_i(d)}{2}$
Hadamard	$f_i(g) \times g_i(d)$
Weighted-L1	$\lvert f_i(g) - g_i(d) \rvert$
Weighted-L2	$\lvert f_i(g) - g_i(d) \rvert^2$

4.5 Gene-Disease Prediction

To evaluate the approach, we tested the performance of supervised classifiers to predict gene-disease associations using the proposed feature vectors.

We used four different machine learning algorithms: RF (Breiman, 2001), XGB (Chen and Guestrin, 2016), NB (Friedman et al., 1997) and MLP (Rumelhart et al., 1986). A Grid search was employed to obtain optimal parameters for RF, XGB, and MLP which is summarized in Table 4.3.

Table 4.3: Grid-Search parameters for the machine learning algorithms.

Algorithm	Parameters	Values
RF	maximum depth	2, 4, 6, None
	nr of estimatores	50, 100, 200
XGB	maximum depth	2, 4, 6
	nr of estimatores	50, 100, 200
	learning_rate	0.1, 0.01, 0.001
MLP	hidden layer sizes	(50, 50, 50), (50, 100, 50), (100,)
	activation	tanh, relu
	solver	sgd, adam
	alpha	0.0001, 0.05
	learning_rate	constant, adaptive

Initially, in each experiment, we performed a stratified ten-fold cross-validation being that the same folds and, for each fold, the Weighted Average of F-measures (WAF) of classifications were assessed and reported in the form of a median. A stratified ten-fold cross-validation consists of the following steps:

1. Split the dataset into ten equally sized folds with class probabilities similar to the original dataset;
2. Train classifier on nine randomly selected folds (training set);
3. Test the trained classifier using the remaining fold (test set);
4. Repeat the process ten times and each time a different fold is used as a test set.

We also performed a stratified 70% training and 30% testing split method, and verified that the overall conclusions were comparable to the 10-fold cross-validation experiments (Table B.1 in Appendix B) although the running time of cross-validation is much higher. Consequently the results presented in the chapter 5 will concern only the 70-30 split, with the same split being used throughout all experiments, including in the baseline presented in the following section.

4.6 Baseline and Experiments

The baseline aims to establish the performance of methods that use a single ontology and classical semantic similarity measures.

The knowledge graph used was **HP-full**. Considering that semantic similarity measures do not explore logical definitions, results using **HP-simple** or **HP-full** are equivalent. The semantic similarity was measured for all gene-disease pairs using six different semantic similarity measures that are summarized in Table 4.4. Semantic similarity computations were run using the tool SSMC[8] which was designed to measure semantic similarity between a set of objects annotated by ontology classes.

Table 4.4: Summary of SSMs used in the baseline.

SSM	IC	Type of approach	Techniques
BMA_{Resnik}	Extrinsic	best pairs	Average
BMA_{Seco}	Intrinsic	best pairs	Average
Max_{Resnik}	Extrinsic	best pairs	Maximum
Max_{Seco}	Intrinsic	best pairs	Maximum
$simGIC_{Resnik}$	Extrinsic	graph-based	Jaccard
$simGIC_{Seco}$	Intrinsic	graph-based	Jaccard

Each of the selected semantic similarity measures is a combination of two approaches: the approach used to calculate the Information Content of each HP class (ICSeco or ICResnik) and the IC-based approach used to calculate the similarity between pairs (BMA or simGIC or Max).

These approaches for IC-based entity similarity were selected because both simGIC and BMA represent high-performing classical measures of semantic similarity, whereas Max helps to elucidate whether a single source of similarity is enough to establish an interaction. By combining the approaches for entity similarity with the different IC, we arrive at the six state-of-the-art semantic similarity measures used: BMAResnik, BMASeco, MaxResnik, MaxSeco, simGICResnik, and simGICSeco. These six measures are representative of the most successful approaches for the baseline calculation using a single ontology.

[8]https://github.com/liseda-lab/SSMC

To establish the performance of the baseline, the association prediction was formulated as a classification problem where a semantic similarity score for a pair exceeding a certain threshold (semantic similarity cutoff) indicates a positive association. For each measure, a semantic similarity threshold was chosen after evaluating the weighted average of F-measures (for positive and negative predictions) at different thresholds intervals and selecting the maximum (values in the range from 0 to 1 with a step of 0.01). This emulates the best choice that a human expert could theoretically select. By comparing the performance of this optimal baseline to the performance of our proposed approach, we aim at investigating the ability of a richer semantic representation to obtain an improved classification performance.

The quality of the classifications is evaluated using the WAF. This metric accounts for class unbalance by computing the F-measure for each interacting and non-interacting class and then calculating the average of both computed F-measures, weighted by the number of instances of each class:

$$WAF = \frac{\Sigma_{c \epsilon C} \text{F-measure}_c \times Support_c}{\Sigma_{c \epsilon C} Support_c} \tag{4.1}$$

where C is the set of classes, $F\text{-}measure_c$ is the F-measure computed for class c, and $Support_c$ is the number of instances in class c. The F-measure (for a class c) is the weighted harmonic mean of the precision and recall and is given by

$$\text{F-measure} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4.2}$$

where

$$Precision = \frac{Number\ of\ instances\ correctly\ classified\ as\ class\ c}{Number\ of\ instances\ classified\ as\ class\ c} \tag{4.3}$$

and

$$Recall = \frac{Number\ of\ instances\ correctly\ classified\ as\ class\ c}{Number\ of\ instances\ labeled\ as\ class\ c} \tag{4.4}$$

In addition to the WAF, occurred an evaluation of the performance with the Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

Chapter 5

Results and Discussion

There are several factors of the proposed methodology for rich semantic representations that can impact the performance of gene-disease association prediction, such as the semantic richness and domain coverage of the knowledge graph, the knowledge graph embedding methods, and the operators used to combine gene and disease vectors. Given these factors, there are three important aspects that need to be considered when elucidating the performance impact:

1. How can we combine the gene and disease vector?
2. Which knowledge graph embedding methods are more suitable for this task?
3. What is the impact of considering more than one ontology?

In this chapter, the evaluation of the methodology described in Chapter 4 is presented and discussed. First, the results for the gene-disease prediction baseline established using semantic similarity measures are described. Then, the results obtained using a rich semantic representation with different embeddings methods, knowledge graphs, and machine learning techniques are presented and compared to the best semantic similarity measure that the baseline achieved as well as the calculated cosine similarity for each knowledge graph embedding.

5.1 Baseline Performance

The results for predictions based on the six semantic similarity measures using the knowledge graph **HP-full** are presented in Table 5.1. When using semantic similarity, it is irrelevant whether the simple or full versions of HP are used, since these measures only consider hierarchical *is_a* relations between ontology concepts. Moreover, simple semantic similarity measures are unable to explore more than one ontology.

Concerning the performance of the six semantic similarity measures in terms of information content for WAF score with a single ontology, we observe that the Max approach

Table 5.1: WAF and AUC-ROC scores for optimal SSM performance with HP ontology. The blue and red values show the best score for WAF and AUC-ROC, respectively.

SSM	BMA_{Resnik}	BMA_{Seco}	Max_{Resnik}	Max_{Seco}	$simGIC_{Resnik}$	$simGIC_{Seco}$
WAF	0.682	**0.684**	0.681	0.633	0.633	0.636
AUC-ROC	0.713	**0.725**	0.712	0.662	0.671	0.667

is more sensitive to the IC measure employed. While for BMA and simGIC, differences between IC_{Resnik} and IC_{Seco} are rather small, when using the Max approach, these differences are more pronounced, with differences up to 5% both in WAF and AUC-ROC. This may be explained by the fact that the Max approach only considers the best pair of classes when measuring similarity, with the IC measure playing a major role in the similarity score, whereas both BMA and simGIC consider all annotating classes. The differences between the approaches seen when using Max can potentially highlight that when comparing single pairs of classes, it is more relevant how often they are used to annotate entities, rather than where they are placed in the graph.

Regarding the combination approach, the pairwise approach followed by BMA achieves the best results with a top WAF of 0.684 and AUC-ROC of 0.725 when combined with IC_{Seco}. This highlights two interesting aspects. On the one hand, considering all phenotypes globally, as simGIC does, represents a loss in performance (of about 5% in WAF), since highly diverse phenotypes are compared indiscriminately. On the other, circumscribing the comparison to the most similar phenotype, as Max does, is also limiting when we consider that many diseases present multiple phenotypes. However, this is less impactful on performance, especially when using IC_{Resnik}.

These simple baselines afford a view of what a perfectly chosen similarity threshold could achieve, yet they still yield performance scores below 0.70 in WAF. Going forward, BMA_{Seco} was the measure chosen as the main semantic similarity baseline since it achieved the best results overall.

5.2 Rich Semantic Representations Performance

5.2.1 Comparison of Vector Combination Approaches

One of the challenges in achieving a rich semantic representation of genes and diseases when using knowledge graph embeddings is to define a suitable approach to combine the gene and disease vectors. Figure 5.1 summarizes the comparison of the five chosen vector operations with AUC-ROC evaluated using the three best knowledge graph embedding methods (RDF2Vec, OPA2Vec, and DistMult) coupled with Random Forest classifier (one of the best-performing machine learning algorithms) using the richest knowledge graph (**HP-simple + LD + GO**). Table 5.2 provides a more detailed view with the WAF scores achieved for each operator using all machine learning and embeddings approaches in the

same knowledge graph. Results for all knowledge graphs are available in Tables C.1, C.2, C.3, C.4, and C.5 (present in Appendix C).

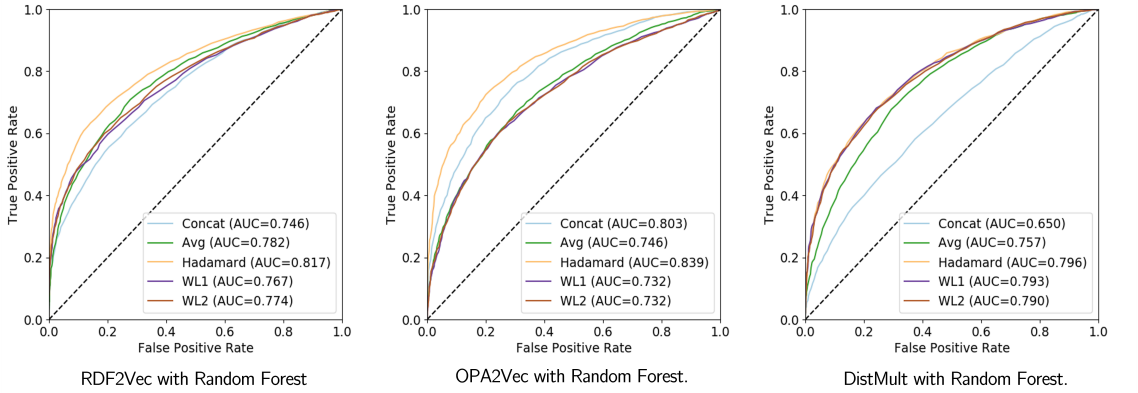

Figure 5.1: ROC curves and AUC values obtained for different vector operators with RF classifier for the **HP-simple + LD + GO**.

Table 5.2: Comparison of vector combination operators (WAF scores) using the **HP-simple+LD+GO** KG. In bold is the best result possible in every KGE.

		RDF2Vec	OPA2Vec	OWL2Vec	DistMult	TransE
Concatenation	**RF**	0.672	0.728	0.617	0.603	0.484
	XGB	0.702	0.739	0.642	0.667	0.490
	NB	0.517	0.513	0.507	0.380	0.480
	MLP	0.732	0.743	**0.707**	**0.738**	0.487
Average	**RF**	0.714	0.683	0.652	0.683	0.509
	XGB	0.700	0.690	0.645	0.704	0.511
	NB	0.608	0.573	0.582	0.409	0.492
	MLP	0.711	0.717	0.679	0.700	0.366
Hadamard	**RF**	**0.743**	0.759	0.695	0.716	0.473
	XGB	0.739	**0.760**	0.694	0.724	**0.514**
	NB	0.617	0.530	0.582	0.467	0.500
	MLP	0.732	0.749	0.694	0.714	0.333
Weighted-L1	**RF**	0.693	0.677	0.623	0.714	0.508
	XGB	0.701	0.675	0.617	0.712	0.505
	NB	0.679	0.567	0.603	0.525	0.501
	MLP	0.699	0.695	0.620	0.696	0.487
Weighted-L2	**RF**	0.702	0.676	0.611	0.707	0.508
	XGB	0.701	0.675	0.630	0.715	0.505
	NB	0.669	0.540	0.593	0.629	0.498
	MLP	0.704	0.698	0.640	0.699	0.333

The Hadamard operator outperforms other operators when using RDF2Vec, OPA2Vec, and TransE, whereas Concatenation works best with OWL2Vec and DistMult. Overall, Hadamard and Concatenation are the top two performing combination approaches, with Hadamard achieving the best prediction results when combined with OPA2Vec and Random Forests or XG-Boost.

While Hadamard, Average, Weighted-L1, and Weighted-L2 all produce vectors of the same size (200), Concatenation produces double-sized vectors (400). This impacts the training time of the machine learning algorithms. Considering the small losses in performance by using Hadamard with OWL2Vec and DistMult, going forward all experiments focus on the Hadamard operator.

5.2.2 Comparison of Knowledge Graph Embedding Methods

Table 5.3 compares knowledge graph embedding methods with the baseline, presenting the performance obtained for the XGB algorithm and with cosine similarity for all the possible competing combinations of knowledge graphs embeddings approaches, vector operators used and knowledge graphs created.

Comparing the cosine similarity of embeddings vectors with the semantic similarity baseline, it can be seen that the cosine similarity only outperforms the best semantic similarity measure, BMA_{Seco}, when using DistMult and only on some knowledge graphs like **HP-simple** and **HP-simple+LD+GO**. Since knowledge graph embeddings can explore all types of semantic relations, including logical definitions asserted between ontologies, the expectation was that cosine similarity would outperform semantic similarity measures, which was not observed in most cases.

When using a machine learning approach the differences between using knowledge graph embeddings and the baseline are more distinct, especially when it comes to the best three knowledge graph embedding methods: RDF2Vec, OPA2Vec, and DistMult. In almost every knowledge graph embedding the best result was achieved with the Hadamard operator and outperformed both the baseline as well as the cosine similarity. The only exception was TransE, the only approach that falls into the category of translational distance, which achieved poorer results overall with Weighted-L1 and Weighted-L2 but did not outperform the baseline and in many cases not even the cosine similarity. Thhis is due to the fact that translational distance methods are less well suited to capture long-distance relations than random-walk and semantic matching methods, which is more relevant in the context of ontologies (where hierarchical relations between ontology classes are the backbone) than in the strict knowledge graph context (where links between entities are the focus). These results reveal that a suitable knowledge graph embedding approach combined with an appropriate machine learning algorithm outperform a single ontology semantic similarity-based prediction, even with an optimally chosen threshold, by up to 7.6% in WAF.

Table 5.3: WAF scores for the competing combination of KGEs and vector operations for the different KGs using XGB. Best result for each KG is in bold. Best result for each KGE is underlined.

		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
RDF2Vec	**CS**	0.674	0.672	0.680	0.677	0.676
	Concatenation	0.701	0.703	0.690	0.697	0.702
	Average	0.697	0.696	0.689	0.697	0.700
	Hadamard	0.724	0.723	0.732	0.734	0.739
	Weighted-L1	0.668	0.666	0.732	0.698	0.701
	Weighted-L2	0.668	0.678	0.695	0.698	0.701
OPA2Vec	**CS**	0.674	0.658	0.653	0.666	0.671
	Concatenation	0.737	0.727	0.746	0.734	0.739
	Average	0.692	0.686	0.680	0.681	0.690
	Hadamard	**0.751**	**0.741**	**0.758**	**0.750**	**0.760**
	Weighted-L1	0.675	0.680	0.669	0.656	0.675
	Weighted-L2	0.675	0.681	0.669	0.656	0.675
Owl2Vec	**CS**	0.665	0.656	0.654	0.649	0.641
	Concatenation	0.651	0.638	0.650	0.632	0.642
	Average	0.659	0.637	0.645	0.632	0.645
	Hadamard	0.697	0.674	0.695	0.678	0.694
	Weighted-L1	0.618	0.606	0.623	0.617	0.617
	Weighted-L2	0.618	0.606	0.623	0.617	0.630
DistMult	**CS**	0.700	0.682	0.680	0.674	0.689
	Concatenation	0.683	0.676	0.676	0.651	0.667
	Average	0.692	0.699	0.698	0.686	0.704
	Hadamard	0.719	0.705	0.713	0.703	0.724
	Weighted-L1	0.708	0.711	0.712	0.701	0.712
	Weighted-L2	0.706	0.707	0.708	0.699	0.715
TransE	**CS**	0.513	0.523	0.516	0.512	0.518
	Concatenation	0.477	0.474	0.499	0.501	0.490
	Average	0.493	0.500	0.490	0.513	0.511
	Hadamard	0.506	0.503	0.502	0.510	0.514
	Weighted-L1	0.539	0.529	0.500	0.507	0.505
	Weighted-L2	0.539	0.529	0.500	0.507	0.505
BMA_{Seco}		**—-**	0.684	**—-**	**—-**	**—-**

Overall, knowledge graph embeddings coupled with machine learning algorithms achieve better results than cosine similarity. This is unsurprising since reducing the representation of a gene-disease association to a similarity score may be too limiting. A model learned on multi-dimensional representations is much better at capturing the complexity of the associations. The best results achieved by each knowledge graph embedding method with XG-Boost presented in Table 5.3 improve on the best cosine similarity results by between 2.4% for DistMult and 8.6% for OPA2Vec.

While TransE clearly struggles in the gene-disease association prediction task, regardless of operator, machine learning algorithm or knowledge graph employed, the other four methods outperform the baseline.

RDF2Vec is positioned as the second best performer, with 0.739 WAF, showing the potential of path-based methods using language modeling approaches for unsupervised feature extraction from RDF graphs. OWL2Vec, however, is the fourth best performer, with 0.697 WAF. The worse results when comparing to RD2Vec are unexpected, since it uses the same techniques as the RDF2Vec but also takes into account OWL axioms including lexical information. Potentially, a deeper exploration of OWL axioms introduces more noise into the representations.

OPA2Vec undeniably achieves the best results in every single knowledge graph when combined with the Hadamard operator, with a best score of 0.760 WAF. The better performance of OPA2Vec can be explained by multiple factors: it uses asserted and inferred logical axioms in ontologies by using a reasoner; it combines them with vector representations for the lexical component of the ontologies learned over PubMed abstracts using the word2vec model. A clear difference between OPA2Vec and RDF2Vec is the use of rich OWL axioms and word embeddings, which may explain the observed differences. Biomedical ontologies are rich in synonyms and exploring their similarities in the context of scientific literature can be immensely informative. In other words, this algorithm shows better results because it is better tailored to the specifics of bio-ontologies.

DistMult is the third best performing approach, achieving 0.724 in WAF score. DistMult is more directly comparable to RDF2Vec since they both limit themselves to exploring the RDF graph. The results show an advantage for the random-walk method, which however is highly dependent on the concatenation operator employed.

Overall, the best combination of methods achieves 0.760 WAF by using the XGB classifier with the Hadamard operator and OPA2Vec as depicted in Table 5.2. Random forest also showed very promising results using the Hadamard operator especially when it comes to the RDF2Vec method. MLP worked well with OWL2Vec and DistMult when using the Concatenation operator. These results are aligned with the fact that both XGBoost and Random Forest are among the most popular and best performing supervised learning algorithms outside of the deep learning category. In this work, the size of the data did not motivate the use of deep learning algorithms.

The poor results achieved with the NB classifier may be justified by the fact that this algorithm's main limitation is the assumption of independent predictor features. This algorithm implicitly assumes that all the attributes are mutually independent but working with linked data this assumptions clearly does not stand.

5.2.3 Comparison of different Knowledge Graphs

The last aspect to be considered is the influence of the richness of the knowledge graph employed. Overall, no dramatic differences between the different knowledge graphs were observed. Table 5.4 presents the performance obtained across all knowledge graphs, knowledge graph embeddings methods and machine learning methods.

Table 5.4: WAF scores for the combinations of KGE and machine learning algorithms for the different KGs using the Hadamard operator. In bold the best result.

		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
RDF2Vec	**CS**	0.674	0.672	0.680	0.677	0.676
	RF	0.726	0.720	0.730	0.737	0.743
	XGB	0.724	0.723	0.732	0.723	0.739
	NB	0.609	0.609	0.613	0.630	0.617
	MLP	0.717	0.722	0.726	0.737	0.732
OPA2Vec	**CS**	0.674	0.658	0.653	0.666	0.671
	RF	0.746	0.743	0.754	0.750	0.759
	XGB	0.751	0.741	0.758	0.750	**0.760**
	NB	0.511	0.501	0.532	0.529	0.530
	MLP	0.737	0.732	0.755	0.755	0.749
OWL2Vec	**CS**	0.665	0.656	0.654	0.649	0.641
	RF	0.694	0.671	0.699	0.685	0.695
	XGB	0.697	0.674	0.695	0.678	0.694
	NB	0.569	0.558	0.582	0.574	0.582
	MLP	0.689	0.672	0.690	0.676	0.694
DistMult	**CS**	0.700	0.682	0.680	0.674	0.688
	RF	0.725	0.702	0.717	0.697	0.716
	XGB	0.719	0.705	0.713	0.703	0.724
	NB	0.493	0.425	0.550	0.415	0.467
	MLP	0.708	0.703	0.708	0.708	0.714
TransE	**CS**	0.513	0.523	0.516	0.512	0.518
	RF	0.493	0.497	0.482	0.479	0.473
	XGB	0.506	0.503	0.502	0.510	0.514
	NB	0.509	0.503	0.517	0.503	0.500
	MLP	0.333	0.333	0.333	0.333	0.333
Baseline	**BMA_{Seco}**			0.684		

When using only the Human Phenotype ontology, the simple version without logical definitions (**HP-simple**) typically achieves a better performance than the complete version **HP-full**. The principal function of a logical definition is to define the classes in one ontology using classes from other ontologies, establishing a semantic bridge between

them. It is likely that this additional information is generating background noise that is irrelevant for gene-disease association prediction. However, some methods appear to be more robust to this aspect, with RDF2Vec losing little to no performance between the two knowledge graphs.

When the Gene Ontology is added, different behaviours are observed. Both RFD2Vec and OPA2Vec achieve better results with **HP-simple + GO** compared to **HP-simple**, with performance increasing by around 1-2%. For OWL2Vec and DistMult this behaviour is not observed for all machine learning algorithms, especially not the best performing ones (XGB and RF).

With **HP-full + GO**, we observe the same pattern, where RDF2Vec and OAP2Vec improve on the **HP-full** results. However, RDF2Vec and OPA2Vec behave differently when comparing with **HP-simple + GO**. While RDF2Vec's performance generally increases when using the full version, OPA2Vec's performance decreases. This decrease in performance is also observed for OWL2Vec and DistMult. It appears that these methods struggle to create more meaningful representations even in the presence of a richer graph. A possible reason behind this is that a graph with richer semantics when processed by methods that are able to explore those richer semantics results in entity vectors that capture many different aspects that may not be relevant for gene-disease association prediction. Another explanatory aspect could be related to the proximity in the graph between the HP class declaration and the related GO class. Logical definitions can be quite complex and include a number of different entities from different ontologies as well as semantic constructs. In triple oriented methods, such as OPA2Vec and DistMult, the relation between the HP class and the GO class are not directly encoded at the triple level, and it needs to be learned by jointly training on all triples. In random-walk based methods, such as RDF2Vec, paths linking both classes can be found, making the relation more explicit.

To delve deeper into this issue, the logical definitions declared in the HP ontology were analyzed and a total of 3203 definitions were identified, but only around 10% of those (350) are related to the Gene Ontology. This motivated the creation of another knowledge graph, (**HP-simple + LD + GO**), that addresses both challenges: it only includes logical definitions with GO (potentially removing noise) and it establishes direct links between HP and GO classes (making the relation more explicit in the graph).

The best results for RDF2Vec and OPA2Vec are achieved with **HP-simple + LD + GO** reaching 0.739 and 0.760 respectively. OWL2Vec and DistMult also improve on their results versus **HP-full + GO** (although not overall), which supports the hypothesis that **HP-full** is introducing noise into the prediction problem.

Some general patterns can be observed, especially when taking advantage of the Hadamard operator, for each knowledge graph embedding approach. The knowledge graph embeddings OPA2Vec and OWL2Vec exhibit the same behavior: performance

drops from using **HP-simple** to a full version, with or without the addition of the Gene Ontology. However, OPA2Vec improves and achieves the highest results when using the direct logical definitions, whereas OWL2Vec does not present this behavior. Regarding the RDF2vec method, the performance generally increases as the graph becomes richer but removing the noise from **HP-full+GO** and using **HP-simple+LD+GO** also increases the WAF score for the best machine learning algorithms.

Figure 5.2 presents precision and recall values for all knowledge graph embeddings using XGB and Hadamard. Interestingly, these values reveal that the increase in performance observed as semantic richness increases for different knowledge graph version is achieved through recall gains. This supports the hypothesis that using more ontologies and richer representations affords more information that is useful to support gene-disease association predictions.

Overall the differences between using a single ontology or combining two ontologies, in this case the Human Phenotype Ontology and Gene Ontology, are comparatively small regardless of a richer integration using logical definitions. The small contribution to performance observed when adding the GO may partially be explained by the existence of only 350 (of a total of 3203) logical definitions that link the two ontologies. Moreover, it is possible that the additional information provided by GO is not supporting novel predictions over those already uncovered by using HP.

5.2.4 A case study on gene BACH2 and KPD disease

Table 5.5 illustrates an example of a gene-disease association prediction between the gene BACH2 (Gene ID:60468) and Ketosis-prone diabetes mellitus (KPD) (C3837958).

In addition to classic type 1 (T1D) and type 2 (T2D) diabetes mellitus, atypical presentations are seen, particularly in populations of African ancestry. KPD (Balasubramanyam et al., 2008), the most common atypical form, is characterized by an acute initial presentation with severe hyperglycemia and ketosis, as seen in classic T1D. This is not a monogenic disease and in our dataset it is also associated with other 29 genes (ABCC8, SH2B3, IL2RA, STAT3, TYK2, SLC11A1, IGF1, IL10, ITPR, GLIS3, HLA-DQB1, AIFM1, CTSH, KCNJ1, DDIT3,INS, PTPN22, HLA-DQA1, SLC29A3, HSD11B2, CAT, TNF, CP, HLA-DRB1, IL6, IFNG, NOS3, HNF1A, and NOS1).

The BACH2 gene is commonly associated with T1D and in every form of diabetes there is an alteration in the functions of beta-cells, which are also associated with this gene. In our dataset this gene is also associated with other two diseases beside T1D and KPD: Crohn's disease (IBD1) and Acute myeloid leukemia (AML). In the case of KPD, a severe form of beta-cell dysfunction appears to underlie this pathophysiology (Balasubramanyam et al., 2008). So the association found between this gene and disease can be explained indirectly by the alterations of the beta-cells, because one alteration of the expression of the BACH2 gene does not necessarily means we have the KPD pathology.

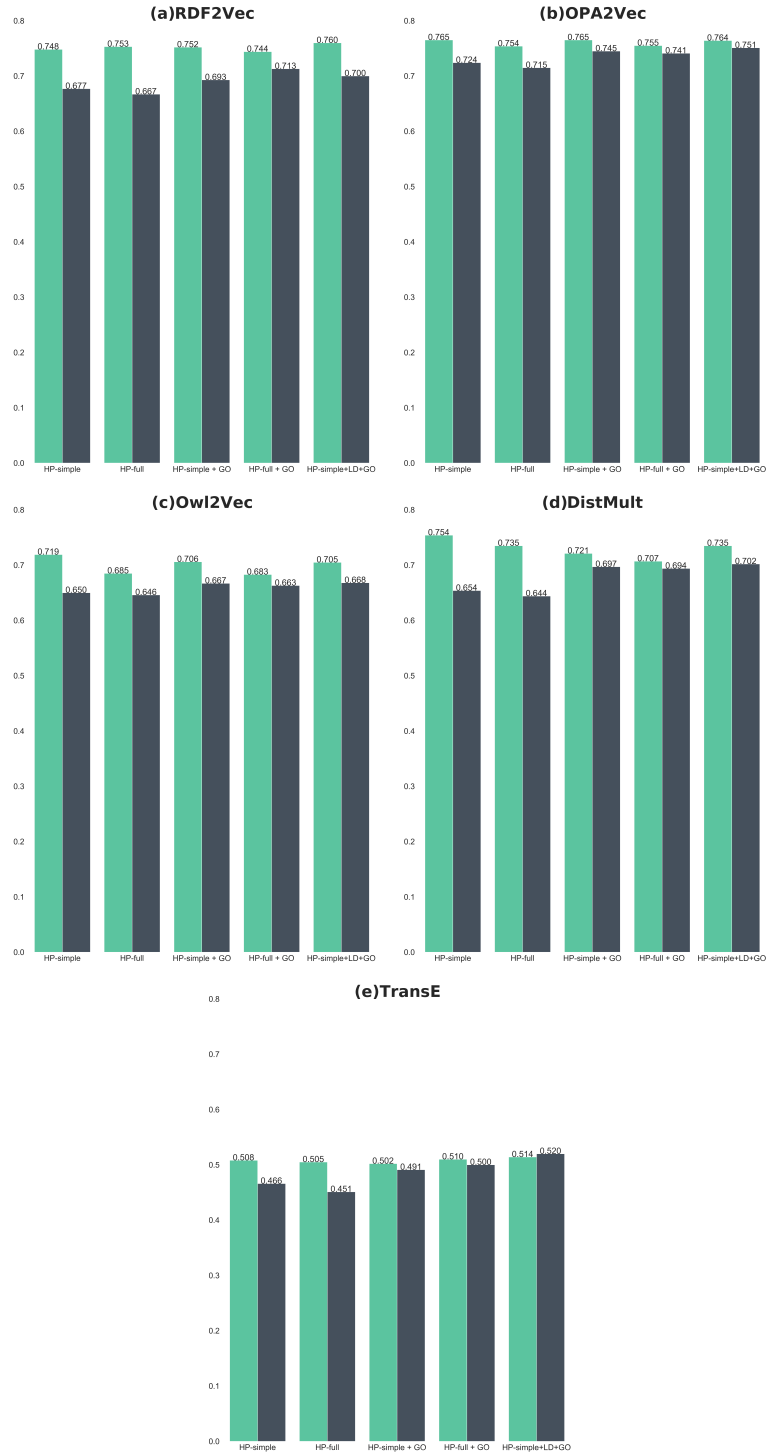

Figure 5.2: Precision and Recall for each KGE using XGB and Hadamard. The KGs appear on the x-axis. **a)** RDF2Vec **b)** OPA2Vec **c)** OWL2Vec **d)** DistMult **e)** TransE.

Analyzing the results, the knowledge graph embedding method can predict correctly the association, however, the best semantic similarity measure does not: the best cutoff is placed at 0.43 similarity score which means that under this value every pair is not considered an association. Regarding the semantic annotations of these entities, both

Table 5.5: Gene-disease association prediction of the pair BACH2-KPD made by the best SSM and KGE method OPA2Vec with random forest and the KG HP-simple + LD + GO.

	Pair	Annotations	BMASeco Prediction (max cuttof of 0.43)	KGE - OPA2Vec Prediction
Gene	BACH2	Variable expressivity (HP:0003828); Recurrent sinopulmonary infections (HP:0005425); Autosomal dominant inheritance (HP:0000006); Recurrent otitis media (HP:0000403); Bronchiectasis (HP:0002110); Colitis (HP:0002583); Splenomegaly (HP:0001744); Decreased circulating antibody level (HP:0004313); Pulmonary infiltrates (HP:0002113); Negative regulation of transcription by RNA polymerase II (GO_0000122); Nuclear chromatin (GO_0000790); DNA-binding transcription factor activity, RNA polymerase II-specific (GO_0000981); DNA-binding transcription repressor activity, RNA polymerase II-specific (GO_0001227); Protein binding (GO_0005515); Nucleoplasm (GO_0005654); Cytosol (GO_0005829); Import into nucleus (GO_0051170); Primary adaptive immune response involving T cells and B cells (GO_0090721); Sequence-specific double-stranded DNA binding (GO_1990837); Regulation of transcription by RNA polymerase II (GO_0006357); DNA-binding transcription factor activity (GO_0003700); RNA polymerase II cis-regulatory region sequence-specific DNA binding (GO_0000978); Nucleus (GO_0005634).	Incorrect SS score:0.400	Correct
Disease	Ketosis-prone diabetes mellitus	Diabetes mellitus (HP:0000819); Ketoacidosis (HP:0001993); Insulin resistance (HP:0000855); Autosomal dominant inheritance (HP:0000006); Autosomal recessive inheritance (HP:0000007); Autoimmunity (HP:0002960); Beta-cell dysfunction (HP:0006279); Multifactorial inheritance (HP:0001426).		

share 'Autosomal dominant inheritance', while 'Decreased circulating antibody level' which annotates BACH2 and 'Autoimmunity' which annotates KPD are both subsumed by 'Abnormality of the immune system'. In addition, while HP does not encode relations between inflammatory bowel disease and diabetes, there are several works identifying a possible link. Since OPA2Vec employs word embeddings trained on PubMed, it is possible that the embeddings reflect some closeness between diabetes and inflammatory bowel disease.

This example illustrates the limitations of semantic similarity-based approaches in handling complex diseases which are related to more than one gene, and genes related to more than one disease. On one hand, a single score view coupled with a simple prediction approach based on a similarity threshold can fail to identify the association between a gene and a disease with few closely related annotations when using a measure such as BMA that considers all annotations. On the other, the Max approach is less suitable when several annotations support a prediction. Moreover, machine learning models coupled with embeddings afford the formulation of a more complex solution.

Chapter 6

Conclusions

Deciphering the links between genes and diseases is an important area of research given it is a crucial challenge in human health with applications to understand disease etiology and develop new techniques for prevention, diagnosis, and therapy.

Computational approaches present themselves as an answer to the data deluge in the life sciences, and ontologies and knowledge graphs have become increasingly crucial to support data intensive applications in biology. In particular, they present several opportunities in supporting the prediction and prioritization of gene-disease associations.

State-of-the-art approaches that take advantage of ontologies for predicting gene-disease associations are typically based on semantic similarity measures and take into account only one ontology. This study proposed a novel approach to predict gene-disease associations using rich semantic representations based on knowledge graph embeddings over multiple ontologies, in this case the Human Phenotype Ontology and the Gene Ontology. The impact of different approaches to build a shared rich semantic representation for genes and diseases was investigated, as well as multiple knowledge graph embedding methods such as RDF2Vec and OPA2Vec.

An unbiased benchmark dataset was created to support evaluation, ensuring its appropriateness for gene-disease prediction. This approach contemplated the integration of the different ontologies and annotation data to build different knowledge graphs. The embeddings that represented the gene and the disease according to their annotations in the knowledge graphs were created and combined using different vector operators and finally, supervised learning algorithms were trained over the combined embeddings in order to predict gene-disease associations.

The experiments provided answers to the research questions. Namely, they showed that knowledge graph embeddings when coupled with machine learning algorithms achieve a better performance than semantic similarity measures, answering RQ1 (in section 1.1). We have shown that employing the best knowledge graph embedding method with machine learning approach outperforms optimal semantic similarity measures by eight 7.6% WAF score. They also illustrated that some vector combination approaches support gene-

disease association prediction better, but that a simple cosine similarity between vectors can support predictions as well as semantic similarity, answering RQ2 (in section 1.1). Finally, the experiments also revealed that differences between using a single ontology or combining two ontologies are comparatively small regardless of a richer integration using the logical definitions, answering RQ3 (in section 1.1). However, there is a clear advantage for most knowledge graph embeddings methods to employ graph versions where logical definitions are encoded as direct links between classes. We hypothesize that the information provided by the Gene Ontology and links to it does not provide substantial additional information comparing with what is already present in the Human Phenotype Ontology.

6.1 Limitations

The main limitation of this work is the fact that only two ontologies were employed to support gene-disease association prediction when other ontologies covering other domains, such as chemicals, drugs, diseases, side-effects, etc., can also be relevant for the domain, and be more complementary to Human Phenotype Ontology.

In terms of evaluation, there are also some limitations. We did not employ multi-ontology semantic similarity measures, which could potentially surpass the single ontology baseline. In most applications semantic similarity measures are only calculated for an ontology, but there are also applications that calculate multi-domain semantic similarity measures. Ferreira and Couto (2019) proposed two approaches that can lift single-ontology measures into multi-domain measures: aggregative approach and integrative approach. The aggregative approach compares each of the domains of relevance independently using existing single-ontology measures and then aggregates the several calculated values; the integrative approach integrates all the ontologies under the same common root and then applies single-ontology measures on it.

6.2 Future Work

From the evaluation perspective, two possible avenues present themselves. First, the inclusion of multi-ontology semantic similarity measures. Secondly, the design of ablation studies, where a part of the Knowledge Graph is removed in order to study the impact of extra knowledge and understand the impact of each component in our approach.

Regarding the predictive approach, a clearly interesting next step to take is the inclusion of other ontologies, for example the Chemical Entities of Biological Interest is an freely available dictionary of molecular entities focused on ‘small’ chemical compounds (Degtyarenko et al., 2008) and could be an interesting addition to the enrich the existing semantic representation.

Additionally, ontology matching techniques can be employed to create additional logical definitions and links between ontologies (Oliveira and Pesquita, 2018). Ontology matching finds correspondences between semantically related entities of ontologies and these correspondences can be used for various tasks, such as ontology merging, query answering, or data translation (Shvaiko and Euzenat, 2013). This opens the possibility to expand the ontologies used to those that do not contain logical definitions between them.

References

Abirami, S. and Chitra, P. (2020). Chapter fourteen - energy-efficient edge based real-time healthcare support system. In Raj, P. and Evangeline, P., editors, *The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases*, volume 117 of *Advances in Computers*, pages 339–368. Elsevier. xi, 21, 22

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. 24, 25

Amberger, J., Bocchini, C., Schiettecatte, F., Scott, A., and Hamosh, A. (2014). Omim.org: Online mendelian inheritance in man (omim®), an online catalog of human genes and genetic disorders. *Nucleic acids research*, 43. 1, 28

Ashburner, M., Ball, C., Blake, J., Botstein, D., Butler, H., Cherry, J., Davis, A. P., Dolinski, K., Dwight, S., Eppig, J., Harris, M., Hill, D., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J., Richardson, J., Ringwald, M., Rubin, G., and Sherlock, G. (2000). Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nat Genet*, 25:25–29. 3

Asif, M., Martiniano, H., and Couto, F. (2018). Identifying disease genes using machine learning and gene functional similarities, assessed through gene ontology. *PLOS ONE*, 13:e0208626. 1, 23, 24

Auslander, N., Gussow, A., and Koonin, E. (2021). Incorporating machine learning into established bioinformatics frameworks. *International Journal of Molecular Sciences*, 22:2903. 9, 11

Balasubramanyam, A., Nalini, R., Hampe, C., and Maldonado, M. (2008). Syndromes of ketosis-prone diabetes mellitus. *Endocrine reviews*, 29:292–302. 45

Bodenreider, O. and Stevens, R. (2006). Bio-ontologies: current trends and future directions. *Briefings in Bioinformatics*, 7(3):256–274. 5

Bordes, A., Usunier, N., García-Durán, A., Weston, J., and Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. In *NIPS*. 17, 33

Breiman, L. (2001). Machine learning, volume 45, number 1 - springerlink. *Machine Learning*, 45:5–32. 11, 34

Cai, H., Zheng, V., and Chang, K. (2018). A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 30:1616–1637. 16, 21

Cao, S., Lu, W., and Xu, Q. (2016). Deep neural networks for learning graph representations. In *AAAI*. 17, 21

Chebolu, P. and Melsted, P. (2008). Pagerank and the random surfer model. In *SODA '08*, pages 1010–1018. 22

Chen, B., Li, M., Wang, J., and Wu, F.-X. (2014a). Disease gene identification by using graph kernels and markov random fields. *Science China. Life sciences*, 57. 23

Chen, B., Shang, X., Li, M., Wang, J., and Wu, F.-X. (2015). A two-step logistic regression algorithm for identifying individual-cancer-related genes. In *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 195–200. 23

Chen, B., Shang, X., Li, M., Wang, J., and Wu, F.-X. (2016). Identifying individual-cancer-related genes by rebalancing the training samples. *IEEE Transactions on NanoBioscience*, 15:309–315. 23

Chen, B., Wang, J., Li, M., and Wu, F.-x. (2014b). Identifying disease genes by integrating multiple data sources. *BMC Medical Genomics*, 7(Suppl 2):1–12. 23

Chen, J., Hu, P., Jimenez-Ruiz, E., Holter, O. M., Antonyrajah, D., and Horrocks, I. (2021). Owl2vec*: Embedding of owl ontologies. 17, 20, 33

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 11, 34

Consortium, T. G. O. (2020). The Gene Ontology resource: enriching a GOld mine. *Nucleic Acids Research*, 49(D1):D325–D334. 3, 30

Consortium, U. (2019). Uniprot: a worldwide hub of protein knowledge. *Nucleic acids research*, 47:D506–D515. 28, 30

Cáceres, J. J. and Paccanaro, A. (2019). Disease gene prediction for molecularly uncharacterized diseases. *PLOS Computational Biology*, 15(7):1–14. 13

d'Amato, C., Staab, S., and Fanizzi, N. (2008). On the influence of description logics ontologies on conceptual similarity. In Gangemi, A. and Euzenat, J., editors, *EKAW*, volume 5268 of *Lecture Notes in Computer Science*, pages 48–63. Springer. 13

Degtyarenko, K., Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcántara, R., Darsow, M., Guedj, M., and Ashburner, M. (2008). Chebi: A database and ontology for chemical entities of biological interest. *Nucleic acids research*, 36:D344–50. 50

Ferreira, J. and Couto, F. (2019). Multi-domain semantic similarity in biomedical research. *BMC Bioinformatics*, 20:246. 50

Friedman, N., Geiger, D., and Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, 29:131–163. 11, 34

Golbreich, C., Horridge, M., Horrocks, I., Motik, B., and Shearer, R. (2007). Obo and owl: Leveraging semantic web technologies for the life sciences. In Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., and Cudré-Mauroux, P., editors, *The Semantic Web*, pages 169–182, Berlin, Heidelberg. Springer Berlin Heidelberg. 6

Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 17, 20, 33

Hastings, J. (2017). *Primer on Ontologies*, pages 3–13. Springer New York, New York, NY. 8

He, S., Liu, K., Ji, G., and Zhao, J. (2015). Learning to represent knowledge graphs with gaussian embedding. *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. 17

Hoehndorf, R., Schofield, P., and Gkoutos, G. (2011). Phenomenet: A whole-phenome approach to disease gene discovery. *Nucleic acids research*. 25

Horrocks, I. (2008). Ontologies and the semantic web. *Communications of the ACM*, 51:58–67. 5, 6

Hou, M., Ren, J., Zhang, D., Kong, X., Zhang, D., and Xia, F. (2020). Network embedding: Taxonomies, frameworks and applications. *Computer Science Review*, 38:100296. xi, 20

Huntley, R., Sawford, T., Mutowo-Meullenet, P., Shypitsyna, A., Bonilla, C., Martin, M., and O'Donovan, C. (2014). The goa database: Gene ontology annotation updates for 2015. *Nucleic acids research*, 43. 30

Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. (2015). Knowledge graph embedding via dynamic mapping matrix. In *ACL*. 17

Ji, G., Liu, K., He, S., and Zhao, J. (2016). Knowledge graph completion with adaptive sparse transfer matrix. In *AAAI*. 17

Kaufman, S., Rosset, S., and Perlich, C. (2011). Leakage in data mining: formulation, detection, and avoidance. In *KDD*. 28

Kiryakov, A., Popov, B., Terziev, I., Manov, D., and Ognyanoff, D. (2004). Semantic annotation, indexing, and retrieval. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2:49–79. 7

Köhler, S., Bauer, S., Mungall, C. J., Carletti, G. O. N., Smith, C. L., Schofield, P. N., Gkoutos, G. V., and Robinson, P. N. (2011). Improving ontologies by automatic reasoning and evaluation of logical definitions. *BMC Bioinformatics*, 12:418 – 418. 8

Köhler, S., Doelken, S., Mungall, C., Bauer, S., Firth, H., Bailleul-Forestier, I., Black, G., Brown, D. L., Brudno, M., Campbell, J., FitzPatrick, D., Eppig, J., Jackson, A., Freson, K., Gîrdea, M., Helbig, I., Hurst, J., Jähn, J., Jackson, L., Kelly, A., Ledbetter, D., Mansour, S., Martin, C., Moss, C., Mumford, A., Ouwehand, W., Park, S.-M., Riggs, E., Scott, R., Sisodiya, S., Vooren, S. V., Wapner, R., Wilkie, A., Wright, C., Silfhout, A. V., Leeuw, N., Vries, B., Washington, N., Smith, C. L., Westerfield, M., Schofield, P., Ruef, B., Gkoutos, G., Haendel, M., Smedley, D., Lewis, S., and Robinson, P. (2014). The human phenotype ontology project: linking molecular biology and disease through phenotype data. *Nucleic Acids Research*, 42:D966 – D974. 30

Köhler, S., Gargano, M. A., Matentzoglu, N., Carmody, L., Lewis-Smith, D., Vasilevsky, N. A., Danis, D., Balagura, G., Baynam, G., Brower, A., Callahan, T., Chute, C., Est, J. L., Galer, P. D., Ganesan, S., Griese, M., Haimel, M., Pazmandi, J., Hanauer, M., Harris, N., Hartnett, M. J., Hastreiter, M., Hauck, F., He, Y., Jeske, T., Kearney, H., Kindle, G., Klein, C., Knoflach, K., Krause, R., Lagorce, D., McMurry, J., Miller, J. A., Munoz-Torres, M., Peters, R. L., Rapp, C. K., Rath, A., Rind, S. A., Rosenberg, A., Segal, M. M., Seidel, M., Smedley, D., Talmy, T., Thomas, Y., Wiafe, S., Xian, J., Yüksel, Z., Helbig, I., Mungall, C., Haendel, M., and Robinson, P. (2021). The human phenotype ontology in 2021. *Nucleic Acids Research*, 49:D1207 – D1217. 3, 13, 29, 30

Kulmanov, M., Smaili, F. Z., Gao, X., and Hoehndorf, R. (2020a). Machine learning with biomedical ontologies. *bioRxiv*. 20, 21

Kulmanov, M., Smaili, F. Z., Gao, X., and Hoehndorf, R. (2020b). Semantic similarity and machine learning with ontologies. *Briefings in Bioinformatics*, 22. 9, 13

Köhler, S., Bauer, S., Horn, D., and Robinson, P. (2008). Walking the interactome for prioritization of candidate disease genes. *American journal of human genetics*, 82:949–58. 23

Köhler, S., Carmody, L., Vasilevsky, N., Jacobsen, J., Danis, D., Gourdine, J.-P., Gargano, M., Harris, N., Matentzoglu, N., McMurry, J., Osumi-Sutherland, D., Cipriani, V., Balhoff, J., Conlin, T., Blau, H., Baynam, G., Palmer, R., Gratian, D., Dawkins, H., and Robinson, P. (2019). Expansion of the human phenotype ontology (hpo) knowledge base and resources. *Nucleic Acids Research*, 47. 30

Lamy, J.-B. (2017). Owlready: Ontology-oriented programming in python with automatic classification and high level constructs for biomedical ontologies. *Artificial Intelligence in Medicine*, 80. 31

Li, M., Li, Q., Ganegoda, U., Wang, J., Wu, F., and Pan, Y. (2014). Prioritization of orphan disease-causing genes using topological feature and go similarity between proteins in interaction networks. *Science China. Life sciences*, 57. 13

Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *AAAI*. 17

Liu, W., Liu, J., and Rajapakse, J. (2018). Gene ontology enrichment improves performances of functional similarity of genes. *Scientific Reports*, 8. 13

Luo, P., Ding, Y., Lei, X., and Wu, F.-X. (2019a). deepdriver: Predicting cancer driver genes based on somatic mutations using deep convolutional neural networks. *Frontiers in Genetics*, 10:13. 22

Luo, P., Tian, L.-P., Ruan, J., and Wu, F.-X. (2019b). Disease gene prediction by integrating ppi networks, clinical rna-seq data and omim data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 16(1):222–232. 22

Luo, P., Xiao, Q., Wei, P.-J., Liao, B., and Wu, F.-X. (2019c). Identifying disease-gene associations with graph-regularized manifold learning. *Frontiers in Genetics*, 10. 23, 24

Maetschke, S. R., Simonsen, M., Davis, M. J., and Ragan, M. A. (2011). Gene Ontology-driven inference of protein–protein interactions using inducers. *Bioinformatics*, 28(1):69–75. 13

Makarov, I., Kiselev, D., Nikitinsky, N., and Subelj, L. (2021). Survey on graph embeddings and their applications to machine learning problems on graphs. *PeerJ Computer Science*, 7:e357. 16, 17, 19, 21

Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *ICLR*. 20

Mordelet, F. and Vert, J.-P. (2011). Prodige: Prioritization of disease genes with multitask machine learning from positive and unlabeled examples. *BMC Bioinformatics*, 12:389 – 389. 22

Natarajan, N. and Dhillon, I. (2014). Inductive matrix completion for predicting gene-disease associations. *Bioinformatics (Oxford, England)*, 30:i60–i68. 23

Nicholson, D. N. and Greene, C. (2020). Constructing knowledge graphs and their biomedical applications. *Computational and Structural Biotechnology Journal*, 18:1414 – 1428. 9

Nickel, M., Murphy, K. P., Tresp, V., and Gabrilovich, E. (2016a). A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104:11–33. 9

Nickel, M., Rosasco, L., and Poggio, T. (2016b). Holographic embeddings of knowledge graphs. In *AAAI*. 17

Nickel, M., Tresp, V., and Kriegel, H. (2011). A three-way model for collective learning on multi-relational data. In *ICML*. 17

Oliveira, D. and Pesquita, C. (2018). Improving the interoperability of biomedical ontologies with compound alignments. *Journal of Biomedical Semantics*, 9. 51

Opap, K. and Mulder, N. (2017). Recent advances in predicting gene–disease associations. *F1000Research*, 6:578. 1

Paulheim, H. (2016). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 8:489–508. 9

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). Deepwalk: online learning of social representations. *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 17

Pesquita, C., Faria, D., Bastos, H., Falco, A., and Couto, F. (2007). Evaluating go-based semantic similarity measures. *Proc 10th Annual Bio-Ontologies Meeting*. 15

Pesquita, C., Faria, D., Falcão, A., Lord, P., and Couto, F. (2009). Semantic similarity in biomedical ontologies. *PLoS Computational Biology*, 5. 13, 14, 15

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. 25, 27

Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. *ArXiv*, abs/cmp-lg/9511007. 14, 15

Ristoski, P. and Paulheim, H. (2016a). Rdf2vec: Rdf graph embeddings for data mining. In Groth, P., editor, *The Semantic Web - ISWC 2016 : 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I*, volume 9981, pages 498–514, Cham. Springer International Publishing. 16, 17, 20, 33

Ristoski, P. and Paulheim, H. (2016b). Semantic web in data mining and knowledge discovery: A comprehensive survey. *J. Web Semant.*, 36:1–22. 16

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. 24

Rodríguez-García, M. Á., Gkoutos, G. V., Schofield, P. N., and Hoehndorf, R. (2016). Integrating phenotype ontologies with phenomenet. *Journal of Biomedical Semantics*, 8. 25

Roweis, S. and Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290 5500:2323–6. 16, 17

Rumelhart, D., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323:533–536. 11, 34

Seco, N., Veale, T., and Hayes, J. (2004). An intrinsic information content metric for semantic similarity in wordnet. In *ECAI*, volume 16, pages 1089–1090. 14

Shu, J., Li, Y., Wang, S., Xi, B., and Ma, J. (2021). Disease gene prediction with privileged information and heteroscedastic dropout. *Bioinformatics*, 37:i410–i417. 24

Shvaiko, P. and Euzenat, J. (2013). Ontology matching: State of the art and future challenges. *Knowledge and Data Engineering, IEEE Transactions on*, 25:158–176. 51

Singh-Blom, U. M., Natarajan, N., Tewari, A., Woods, J., Dhillon, I., and Marcotte, E. (2013). Prediction and validation of gene-disease associations using methods inspired by social network analyses. *PloS one*, 8:e58977. 22

Smaili, F. Z., Gao, X., and Hoehndorf, R. (2018). Onto2vec: joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics*, 34:i52 – i60. 17, 21, 25

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. 17, 21, 24, 25, 33

Sousa, R., Silva, S., and Pesquita, C. (2020). Evolving knowledge graph similarity for supervised learning in complex biomedical domains. *BMC Bioinformatics*, 21. 13

Stevens, R., Wroe, C., Lord, P., and Goble, C. (2004). 1. ontologies in bioinformatics. *Springer, Berlin, Heidelberg*. 7

Su, C., Tong, J., Zhu, Y., Cui, P., and Wang, F. (2020). Network embedding in biomedical data science. *Briefings in bioinformatics*. 18, 19

Sun, P. G., Gao, L., and Han, S. S. (2011). Prediction of human disease-related gene clusters by clustering analysis. *International Journal of Biological Sciences*, 7:61 – 73. 13

Tenenbaum, J., Silva, V. D., and Langford, J. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290 5500:2319–23. 16, 17

Trouillon, T., Welbl, J., Riedel, S., Gaussier, É., and Bouchard, G. (2016). Complex embeddings for simple link prediction. In *ICML*. 17

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. 23, 24

Wakap, S., Lambert, D., Olry, A., Rodwell, C., Gueydan, C., Valérie, L., Murphy, D., Cam, Y., and Rath, A. (2019). Estimating cumulative point prevalence of rare diseases: analysis of the orphanet database. *European Journal of Human Genetics*, 28. 28

Wang, D., Cui, P., and Zhu, W. (2016). Structural deep network embedding. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 17, 21

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*, 29:2724–2743. xi, 2, 16, 17, 18, 19

Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014). Knowledge graph embedding by translating on hyperplanes. In *AAAI*. 17

Whetzel, P., Noy, N., Shah, N., Alexander, P., Nyulas, C., Tudorache, T., and Musen, M. (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:W541–5. 6

Wu, X., Jiang, R., Zhang, M., and Lu, A. (2008). Network-based global inference of human disease genes. mol syst biol 4:189. *Molecular systems biology*, 4:189. 13

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 (ICLR) 2015*. 17, 33

Yang, P., Li, X.-L., Mei, J.-P., Kwoh, C.-K., and Ng, S.-K. (2012). Positive-unlabeled learning for disease gene identification. *Bioinformatics*, 28(20):2640–2647. 22

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. 24

Zeng, X., Ding, N., Rodríguez-Patón, A., and Zou, Q. (2017a). Probability-based collaborative filtering model for predicting gene–disease associations. *BMC Medical Genomics*, 10. 23

Zeng, X., Liao, Y., Liu, Y., and Zou, Q. (2017b). Prediction and validation of disease genes using hetesim scores. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 14(3):687–695. 23

Appendices

Appendix A

KGE Default Parameters

Table A.1: Default Parameters for the KGE.

	Parameters
RDF2Vec/OWL2Vec	Word2vec default parameters: sentences=None, corpus file=None, alpha=0.025, window=5, min count=5, max vocab size=None, sample=0.001, seed=1, workers=3, min alpha=0.0001, sg=0, hs=0, negative=5, ns exponent=0.75, hashfxn=, epochs=5, null word=0, trim rule=None, sorted vocab=1, batch words=10000, compute loss=False, callbacks=(), comment=None, max final vocab=None, shrink windows=True
OPA2Vec	annotations [metadata annotations]: All annotation properties. pretrained [pre-trained model]: Default pre-trained model from http://bio2vec.net/data/pubmed model/ reasoner [reasoner]: Elk debug [debug]: set to no, in which case no intermediate files are kept once the program exits.
DistMult/TransE	work threads(4), train times(500), nbatches(100), alpha(0.001), margin(1.0), bern(0), dimension(50), ent neg rate(1), rel neg rate(0), opt method(”SGD”)

Appendix B

Ten-fold Cross Validation

Table B.1: Median of WAF scores obtained for RDF2Vec and OPA2VEC combined with RF classifier and hadamard operator. The KG used was **HP-simple + LD + GO**.

10-Fold Cross Validation		**Partitions**									
		1st	**2nd**	**3rd**	**4th**	**5th**	**6th**	**7th**	**8th**	**9th**	**10th**
RDF2Vec	**RF**	0.745	0.744	0.728	0.755	0.741	0.757	0.747	0.760	0.728	0.745
	Median	0.745									
OPA2Vec	**RF**	0.777	0.780	0.743	0.777	0.782	0.767	0.767	0.753	0.758	0.759
	Median	0.767									

Appendix C

Results for KGE Methods

Table C.1: WAF scores for RDF2Vec with the competing combinations of ML algorithms and operators for the different KGs in a 70/30 split. In bold is the best result possible in every KG.

RDF2Vec		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
Cosine Similarity		0.674	0.672	0.680	0.677	0.676
Concatenation	**RF**	0.676	0.687	0.676	0.678	0.672
	XGB	0.701	0.703	0.690	0.697	0.702
	NB	0.513	0.509	0.517	0.510	0.517
	MLP	**0.730**	**0.734**	0.722	0.736	0.732
Average	**RF**	0.709	0.698	0.709	0.711	0.714
	XGB	0.697	0.696	0.689	0.697	0.700
	NB	0.594	0.598	0.606	0.616	0.608
	MLP	0.696	0.700	0.723	0.717	0.711
Hadamard	**RF**	0.726	0.720	0.730	0.737	**0.743**
	XGB	0.697	0.723	**0.732**	0.734	0.739
	NB	0.609	0.609	0.613	0.630	0.617
	MLP	0.717	0.722	0.726	**0.737**	0.732
Weighted-L1	**RF**	0.681	0.677	0.704	0.703	0.693
	XGB	0.668	0.666	0.732	0.698	0.701
	NB	0.656	0.656	0.673	0.679	0.679
	MLP	0.678	0.675	0.695	0.693	0.699
Weighted-L2	**RF**	0.682	0.678	0.701	0.696	0.702
	XGB	0.668	0.678	0.695	0.698	0.701
	NB	0.645	0.655	0.668	0.667	0.669
	MLP	0.674	0.670	0.706	0.715	0.704

Table C.2: WAF scores for OPA2Vec with the competing combinations of ML algorithms and operators for the different KGs in a 70/30 split. In bold is the best result possible in every KG.

OPA2Vec		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
Cosine Similarity		0.674	0.658	0.653	0.666	0.671
Concatenation	**RF**	0.732	0.718	0.728	0.716	0.728
	XGB	0.737	0.727	0.746	0.734	0.739
	NB	0.496	0.499	0.498	0.496	0.513
	MLP	0.731	0.733	0.738	0.746	0.743
Average	**RF**	0.698	0.688	0.668	0.676	0.683
	XGB	0.692	0.686	0.680	0.681	0.690
	NB	0.540	0.536	0.564	0.559	0.573
	MLP	0.714	0.712	0.718	0.723	0.717
Hadamard	**RF**	0.746	**0.743**	0.754	0.750	0.759
	XGB	**0.751**	0.741	**0.758**	0.750	**0.760**
	NB	0.511	0.501	0.532	0.529	0.530
	MLP	0.737	0.732	0.755	**0.755**	0.749
Weighted-L1	**RF**	0.679	0.666	0.678	0.658	0.677
	XGB	0.675	0.680	0.669	0.656	0.675
	NB	0.586	0.580	0.557	0.560	0.567
	MLP	0.683	0.691	0.691	0.697	0.695
Weighted-L2	**RF**	0.674	0.669	0.673	0.657	0.676
	XGB	0.675	0.681	0.669	0.656	0.675
	NB	0.507	0.528	0.524	0.531	0.540
	MLP	0.687	0.695	0.693	0.697	0.698

Table C.3: WAF scores for Owl2Vec with the competing combinations of ML algorithms and operators for the different KGs in a 70/30 split. In bold is the best result possible in every KG.

Owl2Vec		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
Cosine Similarity		0.665	0.656	0.654	0.649	0.641
Concatenation	**RF**	0.620	0.607	0.623	0.614	0.617
	XGB	0.651	0.638	0.650	0.632	0.642
	NB	0.499	0.510	0.511	0.507	0.507
	MLP	**0.711**	**0.692**	0.691	**0.692**	**0.707**
Average	**RF**	0.666	0.644	0.664	0.640	0.652
	XGB	0.659	0.637	0.645	0.632	0.645
	NB	0.568	0.560	0.585	0.577	0.582
	MLP	0.672	0.667	0.668	0.668	0.679
Hadamard	**RF**	0.694	0.671	**0.700**	0.685	0.695
	XGB	0.697	0.674	0.695	0.678	0.694
	NB	0.569	0.558	0.582	0.574	0.582
	MLP	0.689	0.672	0.690	0.676	0.694
Weighted-L1	**RF**	0.626	0.600	0.621	0.622	0.623
	XGB	0.618	0.606	0.623	0.617	0.617
	NB	0.604	0.600	0.601	0.599	0.603
	MLP	0.631	0.603	0.630	0.628	0.620
Weighted-L2	**RF**	0.625	0.609	0.624	0.615	0.611
	XGB	0.618	0.606	0.623	0.617	0.630
	NB	0.597	0.599	0.599	0.597	0.593
	MLP	0.628	0.615	0.641	0.618	0.640

Table C.4: WAF scores for DistMult with the competing combinations of ML algorithms and operators for the different KGs in a 70/30 split. In bold is the best result possible in every KG.

DistMult		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
Cosine Similarity		0.700	0.682	0.680	0.674	0.689
Concatenation	**RF**	0.641	0.592	0.592	0.589	0.603
	XGB	0.683	0.676	0.676	0.651	0.667
	NB	0.385	0.426	0.401	0.501	0.380
	MLP	**0.731**	**0.728**	**0.718**	**0.721**	**0.738**
Average	**RF**	0.704	0.677	0.685	0.672	0.683
	XGB	0.692	0.699	0.698	0.686	0.704
	NB	0.417	0.581	0.409	0.416	0.409
	MLP	0.693	0.709	0.711	0.704	0.700
Hadamard	**RF**	0.726	0.702	0.717	0.697	0.716
	XGB	0.719	0.705	0.713	0.703	0.724
	NB	0.493	0.425	0.550	0.415	0.467
	MLP	0.708	0.703	0.708	0.708	0.714
Weighted-L1	**RF**	0.712	0.703	0.708	0.693	0.714
	XGB	0.708	0.711	0.712	0.701	0.712
	NB	0.577	0.634	0.454	0.435	0.525
	MLP	0.690	0.687	0.702	0.697	0.696
Weighted-L2	**RF**	0.699	0.700	0.701	0.699	0.707
	XGB	0.706	0.707	0.708	0.699	0.715
	NB	0.655	0.632	0.496	0.412	0.629
	MLP	0.692	0.693	0.691	0.698	0.699

Table C.5: WAF scores for TransE with the competing combinations of ML algorithms and operators for the different KGs in a 70/30 split. In bold is the best result possible in every KG.

TransE		HP-simple	HP-full	HP-simple + GO	HP-full + GO	HP-simple + LD + GO
Cosine Similarity		0.513	0.523	0.516	0.512	0.518
Concatenation	**RF**	0.479	0.504	0.486	0.503	0.484
	XGB	0.477	0.474	0.499	0.501	0.490
	NB	0.487	0.490	0.493	0.493	0.480
	MLP	0.510	0.496	0.488	0.502	0.487
Average	**RF**	0.502	0.496	0.496	0.502	0.509
	XGB	0.493	0.500	0.490	0.513	0.511
	NB	0.491	0.497	0.496	0.500	0.492
	MLP	0.447	0.506	0.333	0.336	0.366
Hadamard	**RF**	0.493	0.497	0.482	0.479	0.473
	XGB	0.506	0.503	0.502	**0.510**	**0.514**
	NB	0.509	0.503	**0.517**	0.503	0.500
	MLP	0.333	0.333	0.333	0.333	0.333
Weighted-L1	**RF**	0.527	0.521	0.531	0.514	0.508
	XGB	**0.539**	**0.529**	0.500	0.507	0.505
	NB	0.516	0.514	0.510	0.509	0.501
	MLP	0.508	0.510	0.491	0.500	0.487
Weighted-L2	**RF**	0.520	0.499	0.493	0.487	0.500
	XGB	0.539	**0.529**	0.500	0.507	0.505
	NB	0.504	0.516	0.501	0.497	0.498
	MLP	0.333	0.333	0.333	0.333	0.333