University of Oxford

Department of Computer Science

Description Logic Embeddings for Neuro-Symbolic Reasoning

Candidate number: 1049343

Word count: 15,774 (obtained with the `texcount` tool)

A dissertation submitted in partial completion of the
MSc in Computer Science

Trinity 2022

Contents

List of Abbreviations **iii**

1 Introduction **1**

2 Background **4**

2.1 Knowledge graphs 4

2.1.1 Definition 4

2.1.2 Link prediction 5

2.1.3 Knowledge graph embeddings 6

2.2 Description logics 11

2.2.1 The description logic $\mathcal{EL}^{++}$ 12

2.2.2 Reasoning in $\mathcal{EL}^{++}$ 15

2.2.3 $\mathcal{EL}^{++}$ and first-order logic 17

2.2.4 Relationship to knowledge graphs 17

3 A Spatio-Translational Embedding Model for the Description Logic $\mathcal{EL}^{++}$ **19**

3.1 Description logic embeddings and geometric models 19

3.2 Concept representation 20

3.3 Role representation 23

3.3.1 Limitations of translational embedding models 23

3.3.2 Traditional extensions of TransE 24

3.3.3 A spatio-translational model for roles 26

3.4 Training procedure 29

3.4.1 Normal forms 29

3.4.2 Loss functions 30

3.4.3 Negative sampling 36

3.4.4 Regularisation 38

3.4.5 Training algorithm 38

3.5 Soundness 39

4 Empirical Evaluation **41**
4.1 Implementation . . . 41
4.2 Proof of concept: family ontology . . . 42
4.3 Subsumption prediction . . . 44
4.3.1 Datasets . . . 44
4.3.2 Subsumptions between named and complex concepts . . . 45
4.3.3 Baselines . . . 46
4.3.4 Evaluation protocol . . . 46
4.3.5 Experimental protocol . . . 49
4.3.6 Results . . . 49
4.4 Link prediction . . . 53
4.4.1 Datasets . . . 54
4.4.2 Baselines . . . 54
4.4.3 Evaluation and experimental protocol . . . 54
4.4.4 Results . . . 55
4.5 Deductive reasoning . . . 56
4.5.1 Experimental setup . . . 56
4.5.2 Results . . . 57
4.5.3 Comparison of the reasoning and prediction task . . . 57
4.6 Ablation studies . . . 59
4.6.1 Impact of role representation . . . 59
4.6.2 Bump vectors and regularisation . . . 60
4.6.3 Number of negative samples . . . 61

5 Related Work **63**

6 Conclusion and Future Work **66**
6.1 Summary . . . 66
6.2 Critical evaluation . . . 67
6.3 Future work . . . 67

Bibliography **69**

List of Abbreviations

AUC . . . . . Area under the curve

DL . . . . . Description logic

DLE . . . . . Decription logic embedding

FOL . . . . . First-order logic

FPR . . . . . False positive rate

GCI . . . . . General concept inclusion

KG . . . . . Knowledge graph

KGE . . . . . Knowledge graph embedding

Med . . . . . Median rank

MR . . . . . Mean rank

MRR . . . . . Mean reciprocal rank

OWL . . . . . Web Ontology Language

PPI . . . . . Protein-protein interaction

ROC . . . . . Receiver operating characteristic

TPR . . . . . True positive rate

1 Introduction

One of the fundamental problems in artificial intelligence is representing and reasoning with knowledge. This is an essential requirement for any system that is to exhibit any sort of intelligent behaviour, whether it is a self-driving car planning an optimal driving route or a digital assistant using information from the internet to answer a user query.

A variety of formalisms for knowledge representation have been developed within the field of artificial intelligence. A particular prominent such formalism are *description logics* (DL), a family of knowledge representation languages that not only allow stating facts about entities in some domain of interest, but also support expressing general background knowledge in the form of logical assertions. They have been used with great success in many different domains, ranging from biomedical and medical applications [Smith et al., 2007; Schulz et al., 2009; Hoehndorf et al., 2011] to the semantic web [Grau et al., 2008; Horrocks, 2008].

A crucial feature of DLs is that they are rooted in a precise formal semantics, which can be leveraged by a variety of reasoning algorithms [Tsarkov and Horrocks, 2006; Kazakov et al., 2014; Glimm et al., 2014] to perform logical inference, i.e. uncover information that implicitly follows from the recorded knowledge. Not only can this lead to new insights about the domain of interest, but reasoning is also essential to ensure the stated logical assertions are sensible and do not lead to contradictions.

However, while these reasoning algorithms have been indispensable for the application and success of DLs, the type of reasoning they can perform is inherently limited to strict logical inference as defined by the underlying formal semantics. Often, we do not want to limit ourselves to such rigid *deductive* reasoning, but instead want to derive conclusions that are *probable* from the given data, a task known as *inductive* reasoning.

In recent years, this type of inductive reasoning with knowledge has received a great deal of attention in the context of *knowledge graphs* (KG), a different formalism for knowledge representation that is closely related to DL. A key technique in this setting are so-called *knowledge graph embeddings* (KGE) [Q. Wang et al., 2017], an approach where the entities and facts in a KG are embedded in a continuous latent vector space in a way that preserves the underlying structure of the KG. A wide variety of such embedding techniques have been developed [Nickel et al., 2011; Yang et al., 2015; Trouillon et al., 2016; Schlichtkrull et al., 2018; Balazevic et al., 2019], and they have shown great potential across a range of tasks.

In light of these developments, recent work [Kulmanov et al., 2019; Mondal et al., 2021; Mohapatra et al., 2021; Xiong et al., 2022; Peng et al., 2022] has explored how similar embedding methods can be applied to the setting of DL. The resulting *description logic embeddings* (DLE) are useful in a variety of ways: on the one hand, they can complement classical reasoning algorithms by enabling novel kinds of inductive reasoning, both to predict missing information and to predict new background knowledge. On the other hand, they also have the potential to approximate the deductive reasoning of classical algorithms, possibly permitting substantial performance improvements.

Contribution. While a variety of different DLE techniques have been proposed and shown to be effective in practice, the current approaches still suffer from a major limitation: they are all based on the simple translational KGE model TransE [Bordes et al., 2013] or slight variations thereof, which is known to be inexpressive and unable to capture *one-to-many*, *many-to-one*, or *many-to-many* relationships [Z. Wang et al., 2014; Lin et al., 2015; Abboud et al., 2020]. Furthermore, the current evaluation strategy for

these models focuses only on the basic reasoning task of subsumption between named concepts and does not take complex concepts into account.

In this dissertation, we address these limitations by introducing a new DLE model and evaluation benchmark. Our contributions are as follows:

- We develop *Box^{2}EL*, a new *spatio-translational* DLE model based on the expressive BoxE KGE model [Abboud et al., 2020] and demonstrate how it overcomes the shortcomings of existing approaches. Furthermore, we show that Box^{2}EL is theoretically *sound*, i.e. corresponds to logical models of the underlying DL.
- We introduce a new benchmark for evaluating the inductive reasoning capabilities of DLE models based on predicting subsumptions between named and complex concepts.
- We perform an extensive empirical analysis of Box^{2}EL and report state-of-the-art results both on our new proposed benchmark and on experiments that have previously been considered. We conduct several ablation studies to highlight the contributions of different parts of our model.
- We analyse the ability of our and competing methods to make logical inferences in the embedding space and gain new insights about the relationship between inductive and deductive sub-symbolic reasoning.

Structure. The remainder of this dissertation is structured as follows: we review relevant background knowledge regarding KGs and define the concrete DL we are working with in Chapter 2. Subsequently, in Chapter 3 we introduce our novel DLE model Box^{2}EL and discuss its conceptual advantages over existing methods. We also explain how Box^{2}EL is trained, and prove that it is theoretically sound. Chapter 4 contains our empirical evaluation across three different settings and demonstrates the performance of our model in practice. Finally, we review related work in Chapter 5, before concluding this dissertation and outlining possible directions for future research in Chapter 6.

2 Background

In this chapter, we provide an account of relevant background material and introduce the basic terms and definitions our work builds upon. We first give a brief review of knowledge graphs, the link prediction task, and knowledge graph embedding techniques. Subsequently, we introduce description logics, which constitute the fundamental formalism that underlies most of our work, and discuss how they relate to knowledge graphs.

2.1 Knowledge graphs

Knowledge graphs (KG) are an effective means to represent and reason with knowledge about the world. They store information in terms of *relational data* consisting of *entities* and *relationships* between them.

2.1.1 Definition

We follow previous literature [S. Ji et al., 2022] and define a KG as a *directed*, *edge-labelled multi-graph*. Formally, a KG is a triple $G = (\mathcal{E}, \mathcal{R}, \mathcal{F})$, where $\mathcal{E}$ denotes the set of entities, $\mathcal{R}$ the set of relations, and $\mathcal{F} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ contains the facts in G. A fact (or *triple*) $(\mathsf{h}, \mathsf{r}, \mathsf{t}) \in \mathcal{F}$ connects the *head* entity h (sometimes also referred to as the *subject* of the fact) with the *tail* entity t (the *object*) via the relation r.

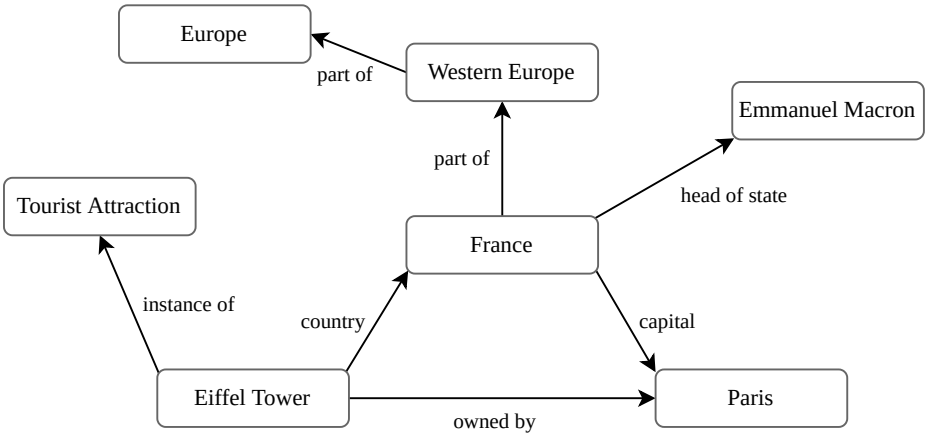


Figure 2.1: A fragment of the Wikidata knowledge graph [Vrandečić and Krötzsch, 2014].

Example 2.1.1. Consider the fragment of the Wikidata KG [Vrandečić and Krötzsch, 2014] depicted in Figure 2.1. In this example, we have the set of entities $\mathcal{E} = \{\text{Eiffel Tower, Tourist Attraction, France, Paris, Europe, Western Europe, Emmanuel Macron}\}$, the set of relations $\mathcal{R} = \{\text{instance of, country, owned by, part of, capital, head of state}\}$, and the set of facts $\mathcal{F}$ corresponds to the edges of the graph.

2.1.2 Link prediction

Modern KGs often contain vast amounts of data — for instance, Wikidata encompasses more than 99 million facts [Pintscher, 2022] — and are usually constructed and maintained in a semi-automated fashion, in which manually curated facts are combined with automated web information extraction techniques [Nickel et al., 2016]. However, despite containing an enormous amount of information and making use of the extensive amount of data available on the internet, existing KGs are still known to be inherently incomplete. For example, Freebase [Bollacker et al., 2008], the predecessor of Wikidata, does not specify the place of birth for more than 70% of the people it contains [West et al., 2014].

A central concern in KG curation is therefore to identify missing information in a given KG. This task can be formally specified as follows: given a KG $G = (\mathcal{E}, \mathcal{R}, \mathcal{F})$, let $\mathcal{F}^+ \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ denote the idealised set of all true facts over $\mathcal{E}$ and $\mathcal{R}$. The problem of *link prediction* (or *knowledge graph completion*) is to identify the true triples $\mathcal{F}^+ \setminus \mathcal{F}$ not contained in G.

2.1.3 Knowledge graph embeddings

A key approach to tackle the link prediction task that has recently emerged are *knowledge graph embeddings* (KGE) [Q. Wang et al., 2017]. The main idea of this technique is to embed the entities and relations of a KG in a continuous latent vector space that preserves the important statistical characteristics of the KG. Predicting missing links between entities then reduces to simple mathematical operations in this vector space.

While there exist a variety of different concrete KGE models, most of them can be formulated as a combination of three interacting parts [Q. Wang et al., 2017]:

- an *entity* and a *relation model* that specify how entities and relations are mapped into a continuous vector space;
- a *scoring function* that assigns scores to facts based on how likely they are to be true; and
- a *learning procedure* that sets up and solves an optimisation problem in order to find embeddings that produce higher scores for true facts than false facts.

Once a KGE model has been trained, i.e. once we have obtained entity and relation embeddings using the learning procedure, we can use the scoring function to predict the likelihood of facts that are not already contained in the KG.

Embedding models

A vast number of KGE models have been proposed in recent years, a detailed discussion of all of which is unfortunately beyond the scope of this dissertation. We instead refer the reader to the relevant literature, e.g. [Nickel et al., 2016; Q. Wang et al., 2017; S. Ji et al., 2022].

Broadly, the different approaches can be divided into three categories [Abboud et al., 2020]: *translational*, *bilinear*, and *neural* KGE models. Translational models such as TransE [Bordes et al., 2013] and RotatE [Sun et al., 2019] embed entities as points and compute scores as distances between these points. Bilinear models like DistMult [Yang et al., 2015] on the other hand employ a multiplicative approach and score triples

using matrix multiplications based on tensor factorisations. Finally, neural models, e.g. [Dettmers et al., 2018; Nathani et al., 2019], utilise sophisticated neural network architectures to differentiate true from false facts.

TransE. We discuss the classic translational model TransE [Bordes et al., 2013] in detail, since it serves as a foundation for many state-of-the-art description logic embedding models.

TransE embeds entities and relations as vectors in the same d-dimensional latent vector space $\mathbb{R}^d$. The embeddings of entities are interpreted as *points*, while relations are interpreted as *translations* of these points. For a triple $(\mathsf{h}, \mathsf{r}, \mathsf{t})$, the model tries to find embeddings such that

$$\boldsymbol{h} + \boldsymbol{r} \approx \boldsymbol{t}, \tag{2.1}$$

where $\boldsymbol{e}$ and $\boldsymbol{r}$ denote the embedding for entity e and relation r, respectively.

The likelihood of a fact should therefore depend on the distance of the translated head embedding and the tail embedding. Hence, the scoring function $s(\mathsf{h}, \mathsf{r}, \mathsf{t})$ is defined as

$$s(\mathsf{h}, \mathsf{r}, \mathsf{t}) = -\|\boldsymbol{h} + \boldsymbol{r} - \boldsymbol{t}\|.$$

Example 2.1.2. Consider again the KG depicted in Figure 2.1. Based on the information that Emmanuel Macron is the head of state of France and Paris is the capital of France, we should be able to predict the missing link $(\mathsf{Emmanuel\ Macron}, \mathsf{lives\ in}, \mathsf{Paris})$. Figure 2.2 illustrates how TransE could solve this link prediction problem in a two-dimensional embedding space.

We finally need to specify how TransE is trained to obtain the entity and relation embeddings. For now, assume we have access to a set of *negative* training examples $\mathcal{F}^-$, i.e. a set of triples that are definitely false. Let $\mathcal{F}^-_{(\mathsf{h},\mathsf{r},\mathsf{t})}$ denote the set of negative examples that differ from $(\mathsf{h}, \mathsf{r}, \mathsf{t})$ in only either the head or the tail. TransE then learns embeddings by optimising the following margin-based ranking loss:

$$\mathcal{L} = \sum_{(\mathsf{h},\mathsf{r},\mathsf{t}) \in \mathcal{F}} \sum_{(\mathsf{h}',\mathsf{r},\mathsf{t}') \in \mathcal{F}^-_{(\mathsf{h},\mathsf{r},\mathsf{t})}} \max\{\boldsymbol{0},\, s(\boldsymbol{h}', \boldsymbol{r}, \boldsymbol{t}') - s(\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t}) + \gamma\}, \tag{2.2}$$

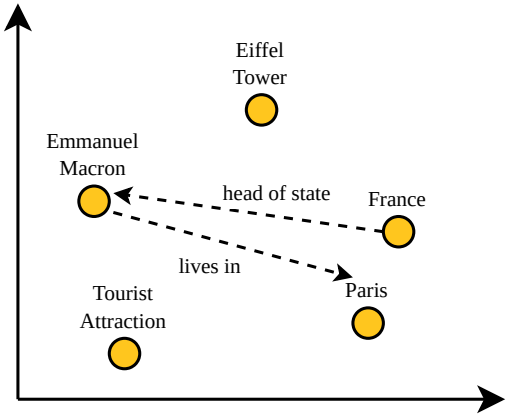


Figure 2.2: An illustration of the TransE embedding model. Yellow circles represent entity embeddings, while relation embeddings are illustrated by dotted arrows. Since the embedding of Emmanuel Macron + lives in is close to the embeddings of Paris and France, the model assigns a high score to the corresponding triples. Moreover, the model correctly captures the fact that Emmanuel Macron is the head of state of France. Incorrect triples such as (Emmanuel Macron, lives in, Eiffel Tower) are assigned a low score.

where $\gamma \geq 0$ is a margin hyperparameter.

The loss in Equation 2.2 encourages true triples to be scored higher than false triples and penalises the model if the opposite is the case. Therefore, in a trained model Equation 2.1 should approximately hold for true facts, but not for false facts.

However, in reality we do not usually have access to negative training examples, since KGs only encode true facts and we do not know whether triples not contained in the KG are true or false. To address this issue, a technique called *negative sampling* is commonly employed.

Negative sampling

Negative sampling is a general technique to obtain negative training examples from a KG and variations of it are used in the training process of most KGE models [Q. Wang et al., 2017]. The basic idea is to generate negative examples by *corrupting* existing facts in the KG by randomly replacing either their head or tail.

Formally, for a given triple $(\mathsf{h}, \mathsf{r}, \mathsf{t})$ define the set of head-corrupted and tail-corrupted

facts as follows:

$$\begin{aligned} HC_{(h,r,t)} &= \{\, (h', r, t) \mid h' \in \mathcal{E} \wedge (h', r, t) \notin \mathcal{F} \,\}, \\ TC_{(h,r,t)} &= \{\, (h, r, t') \mid t' \in \mathcal{E} \wedge (h, r, t') \notin \mathcal{F} \,\}. \end{aligned} \tag{2.3}$$

We can then produce a set of negative examples by sampling uniformly from $HC_{(h,r,t)}$ and $TC_{(h,r,t)}$ for every positive fact. A new set of negative samples is usually generated at every iteration of the training algorithm [Bordes et al., 2013].

Note that this procedure produces a set of corrupted triples that is not contained in the KG, but with no guarantee that the triples are actually false, i.e. not in $\mathcal{F}^+$. However, since the likelihood of a randomly corrupted fact to be false is much higher than the likelihood of it being true, the KGE model can still successfully learn to differentiate true from false facts. Somewhat more sophisticated negative sampling techniques exist that aim to reduce the number of wrongly generated examples [Z. Wang et al., 2014; Krompaß et al., 2015].

Evaluation metrics

In order to evaluate KGE models, a KG is first partitioned into a set of *training*, *validation*, and *testing* triples. As is customary in machine learning, models are trained on the training set, hyperparameters are chosen based on validation set performance, and models are finally evaluated on the testing set.

Given a trained KGE model and a test fact $\varphi = (h, r, t)$, we can evaluate the model performance by comparing the score assigned to φ to the score assigned to its corrupted counterparts in HC_φ and TC_φ [Bordes et al., 2011]. A link prediction model that successfully captures the semantics of the KG should assign a high score to the test fact φ and lower scores to the corrupted triples.

Formally, let $\mathrm{rk}_h(\varphi)$ denote the *rank* of φ within the set $HC_\varphi \cup \{\varphi\}$ ordered in descending order using the model's scoring function. Similarly, let $\mathrm{rk}_t(\varphi)$ be the rank of φ with respect to $TC_\varphi \cup \{\varphi\}$. The *mean rank* is then simply defined as the average rank of all facts in the testing set $\mathcal{F}_{\text{test}}$, i.e.

$$\frac{1}{2|\mathcal{F}_{\text{test}}|} \sum_{\varphi \in \mathcal{F}_{\text{test}}} \Big(\mathrm{rk}_h(\varphi) + \mathrm{rk}_t(\varphi) \Big).$$

Since the mean rank is susceptible to outliers [Hoyt et al., 2022], often the *median rank* or alternatively the *mean reciprocal rank* of the test facts, defined as

$$\frac{1}{2|\mathcal{F}_{\text{test}}|} \sum_{\varphi \in \mathcal{F}_{\text{test}}} \left(\frac{1}{\text{rk}_h(\varphi)} + \frac{1}{\text{rk}_t(\varphi)} \right),$$

is reported instead.

The previous metrics are good measures of average model performance; however, they do not necessarily capture how KGEs are used in practice. In particular, we will often only consider the first k most highly ranked facts for some small integer k to find potential missing links in the data, and are not interested in how the model performs beyond these most highly ranked triples. In this scenario, *hits at k* (also hits@k), which measures the fraction of test triples with rank $\leq k$, is a more appropriate metric [Hoyt et al., 2022]. It is defined as

$$\frac{1}{2|\mathcal{F}_{\text{test}}|} \sum_{\varphi \in \mathcal{F}_{\text{test}}} \left(\mathbb{1}[\text{rk}_h(\varphi) \leq k] + \mathbb{1}[\text{rk}_t(\varphi) \leq k] \right),$$

where $\mathbb{1}[a \leq b]$ is the indicator function that takes on the value of 1 if $a \leq b$ and 0 otherwise.

Another evaluation metric that is commonly used especially in biomedical applications [Kulmanov and Hoehndorf, 2017; Alshahrani et al., 2017] is based on *receiver operating characteristic* (ROC) curves (see e.g. [Fawcett, 2006]). To compute the ROC curve for a KGE model, we regard it as a binary classifier parametrised by a threshold value k, that assigns a label of *true* to a candidate fact if its rank is less than or equal to k and *false* otherwise. We can then compute the *true positive rate* (TPR), which is defined as the proportion of all true triples that are correctly labelled as *true*, and the *false positive rate* (FPR), i.e. the number of triples incorrectly labelled as *true* over the number of false triples. Plotting the TPR against the FPR for varying thresholds k yields the ROC curve for a model.

As illustrated in Figure 2.3, ROC curves can be used to compare the performance of different embedding models. It is furthermore often convenient to summarise an ROC curve with a single number by calculating the *area under the curve* (AUC), i.e.

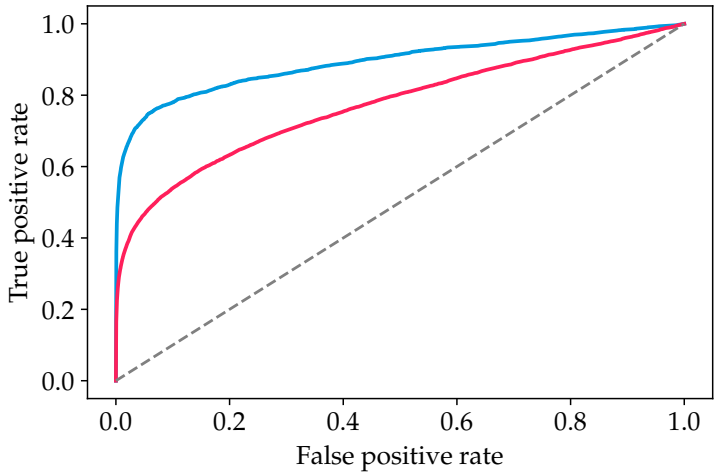


Figure 2.3: ROC curves of two different models. Generally, the closer a curve is to the top left corner, the better the performance of the corresponding model [Fawcett, 2006]. In this example, the model represented by the blue line clearly outperforms the model represented by the red line. The grey line depicts the performance of a classifier that randomly guesses whether a test fact is true or false.

the integral of the ROC curve. The AUC can be interpreted probabilistically as the probability that a model can correctly identify the true fact out of a randomly selected true and a randomly selected false fact [Fawcett, 2006].

The evaluation metrics as defined above are known as the *filtered* versions of these metrics, since true facts are removed from the corrupted facts in Definition 2.3. Generally, this is considered to be the more reliable evaluation strategy [Bordes et al., 2013], but sometimes *raw* metrics are also reported, for which true facts are not filtered out from the corrupted facts.

2.2 Description logics

We now turn our attention to *description logics* (DL), a different, but related, paradigm for knowledge representation. While DLs can represent facts in a similar fashion to KGs, they also allow for the specification of *logical axioms*, which greatly increases their expressive power. Our discussion of DLs closely follows [Baader et al., 2017].

2.2.1 The description logic $\mathcal{EL}^{++}$

There exist a wide variety of DLs, which mostly differ in which language constructs they provide for stating logical axioms. For the purpose of this dissertation, we will focus on a subset of the DL $\mathcal{EL}^{++}$ [Baader et al., 2005].

As with any DL, the statements $\mathcal{EL}^{++}$ allows us to make about some domain of interest encompass *individuals*, *concepts*, and *roles*. Individuals correspond to some notion of objects in the domain of interest, concepts represent sets of objects, and roles are binary relations between objects. In the following, we will first introduce the syntax and precise semantics of $\mathcal{EL}^{++}$ concepts, before we show how they can be used together with logical axioms to represent knowledge about the world.

Syntax of $\mathcal{EL}^{++}$ concepts

Let $\Sigma = (\mathbf{C}, \mathbf{R}, \mathbf{I})$ be a signature of pairwise disjoint sets of concept names $\mathbf{C}$, role names $\mathbf{R}$, and individual names $\mathbf{I}$. The set of $\mathcal{EL}^{++}$ concepts over Σ is then inductively defined as follows:

- Every named concept $C \in \mathbf{C}$ is an $\mathcal{EL}^{++}$ concept.
- The concepts $\top$ (top) and $\bot$ (bottom) are $\mathcal{EL}^{++}$ concepts.
- For every individual $a \in \mathbf{I}$, the *nominal* $\{a\}$ is an $\mathcal{EL}^{++}$concept.
- If C and D are $\mathcal{EL}^{++}$ concepts, their *conjunction* $C \sqcap D$ is an $\mathcal{EL}^{++}$ concept.
- If C is an $\mathcal{EL}^{++}$ concept and $r \in \mathbf{R}$ is a role name, the *existential restriction* $\exists r.C$ is an $\mathcal{EL}^{++}$ concept.

Note that we have omitted *concrete domains* from the above definition, since we will not make use of them in this dissertation.

We often distinguish between two types of concepts: *named* (or *atomic*) concepts are of the form $C \in \mathbf{C}$, $\top$, or $\bot$ and constitute the basic building blocks of *complex* (also called *compound*) concepts, which involve one or more $\mathcal{EL}^{++}$ constructors.

Semantics of $\mathcal{EL}^{++}$ concepts

We next need to specify the semantics of concepts, i.e. how they correspond to sets of objects. To this end, we introduce the notion of an interpretation.

Definition 2.2.1 (Interpretation)**.** An *interpretation* is a tuple $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of a non-empty set $\Delta^{\mathcal{I}}$, the *interpretation domain*, and a mapping function $\cdot^{\mathcal{I}} \colon \Sigma \to \Delta^{\mathcal{I}}$, that maps

- individuals $a \in \mathbf{I}$ to objects $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$;
- concept names $C \in \mathbf{C}$ to subsets $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$; and
- role names $r \in \mathbf{R}$ to binary relations $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

The mapping $\cdot^{\mathcal{I}}$ is extended to arbitrary concepts as follows:

$$\begin{aligned} \top^{\mathcal{I}} &= \Delta^{\mathcal{I}}, \\ \bot^{\mathcal{I}} &= \emptyset, \\ \{a\}^{\mathcal{I}} &= \{a^{\mathcal{I}}\}, \\ (C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}}, \\ (\exists r.C)^{\mathcal{I}} &= \{\, x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}}. (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}} \,\}. \end{aligned}$$

An interpretation thus establishes a precise correspondence between syntactical $\mathcal{EL}^{++}$ concepts and sets of objects in the interpretation domain.

Example 2.2.1. Consider the following interpretation $\mathcal{I}$ for a family domain:

$$\begin{aligned} \Delta^{\mathcal{I}} &= \{w, x, y, z\} & \mathsf{Male}^{\mathcal{I}} &= \{x, y\} \\ \mathsf{hasChild}^{\mathcal{I}} &= \{(w, x), (y, z)\} & \mathsf{Female}^{\mathcal{I}} &= \{w, z\} \\ \mathsf{Parent}^{\mathcal{I}} &= \{w, y\} & \mathsf{Father}^{\mathcal{I}} &= \{y\} \\ \mathsf{Child}^{\mathcal{I}} &= \{x, z\} & \mathsf{Mother}^{\mathcal{I}} &= \{w\} \end{aligned}$$

We have that the complex concept $(\mathsf{Male} \sqcap \exists \mathsf{hasChild}.\mathsf{Female})^{\mathcal{I}} = \{y\}$, since $\mathsf{Male}^{\mathcal{I}} = \{x, y\}$ and $(\exists \mathsf{hasChild}.\mathsf{Female})^{\mathcal{I}} = \{y\}$.

Ontologies

The primary purpose of $\mathcal{EL}^{++}$ concepts is to build an *ontology*, or *knowledge base*, that encodes facts about the world. These facts can be divided into two categories: *terminological statements* correspond to logical axioms and *data assertions* encode information about individuals. In DL terminology, a set of terminological statements is called a *TBox* and a set of data assertions is referred to as an *ABox*.

The kind of logical statements we can make in $\mathcal{EL}^{++}$ are called *general concept inclusion* (GCI) axioms and specify that all objects in one concept must also be contained in another concept. We write $A \sqsubseteq B$ for possibly complex concepts A and B to denote that A must be contained in B. An $\mathcal{EL}^{++}$ TBox is then simply defined as a finite set of GCIs. Furthermore, we say an interpretation $\mathcal{I}$ *satisfies* a GCI $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, and an interpretation that satisfies each GCI in a TBox $\mathcal{T}$ is called a *model* of $\mathcal{T}$, denoted as $\mathcal{I} \models \mathcal{T}$.

Example 2.2.2. Consider the following TBox:

$$\mathcal{T} = \{\mathsf{Male} \sqcap \mathsf{Female} \sqsubseteq \bot, \tag{2.4}$$

$$\exists \mathsf{hasChild}.\top \sqsubseteq \mathsf{Parent}, \tag{2.5}$$

$$\mathsf{Father} \sqsubseteq \mathsf{Male} \sqcap \exists \mathsf{hasChild}.\top, \tag{2.6}$$

$$\mathsf{Mother} \sqsubseteq \mathsf{Female} \sqcap \exists \mathsf{hasChild}.\top\}. \tag{2.7}$$

It can be easily checked that the interpretation $\mathcal{I}$ from Example 2.2.1 satisfies every GCI in $\mathcal{T}$ and is therefore a model of $\mathcal{T}$. In contrast, the interpretation $\mathcal{J}$, defined by $\mathsf{Parent}^{\mathcal{J}} = \{w\}$ and otherwise equivalently to $\mathcal{I}$, is not a model of $\mathcal{T}$ since it violates Axiom 2.5.

$\mathcal{EL}^{++}$ ABoxes on the other hand consist of two different types of data assertions:

- *concept assertions* of the form $C(a)$ for a possibly complex $\mathcal{EL}^{++}$ concept C and an individual $a \in \mathbf{I}$; and
- *role assertions* of the form $r(a, b)$ for a role $r \in \mathbf{R}$ and individuals $a, b \in \mathbf{I}$.

Concept assertions specify that certain individuals must be included in certain concepts, while role assertions denote relationships between individuals. Formally, an interpretation $\mathcal{I}$ *satisfies* a concept assertion $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and a role assertion $r(a,b)$ if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$. An interpretation that satisfies all concept and role assertions in an ABox $\mathcal{A}$ is called a *model* of $\mathcal{A}$, which we denote as $\mathcal{I} \models \mathcal{A}$.

We are now ready to define $\mathcal{EL}^{++}$ ontologies and their models.

Definition 2.2.2 (Ontology and models)**.** An $\mathcal{EL}^{++}$ ontology is a tuple $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ of an $\mathcal{EL}^{++}$ TBox $\mathcal{T}$ and an $\mathcal{EL}^{++}$ ABox $\mathcal{A}$. An interpretation that is both a model of $\mathcal{T}$ and $\mathcal{A}$ is called a *model* of $\mathcal{O}$.

Remark 1. We use the term *ontology* in order to be consistent with recent work in DLEs (e.g. [Kulmanov et al., 2019; Özçep et al., 2020; Chen et al., 2021]). In traditional DL literature, the term *knowledge base* is more commonly used [Baader et al., 2003; Baader et al., 2017].

Remark 2. Formally, $\mathcal{EL}^{++}$ also allows *role inclusion* axioms, but we will not consider them in this dissertation.

Note that the distinction between ABox and TBox is not mathematically meaningful, since any ABox axiom can directly be translated into a semantically equivalent TBox axiom as follows [Kulmanov et al., 2019]:

$$C(a) \rightsquigarrow \{a\} \sqsubseteq C$$
$$r(a,b) \rightsquigarrow \{a\} \sqsubseteq \exists r.\{b\}$$

However, separating ABox and TBox is often useful from a conceptual point of view.

2.2.2 Reasoning in $\mathcal{EL}^{++}$

We have seen that ontologies enable us to formally represent facts about data and logical axioms, and how interpretations provide them with a formal semantics. *Reasoning* algorithms leverage this formal semantics to make logical inferences that implicitly follow from a given ontology. This can be useful in two different ways [Baader et al., 2017]:

- During the construction phase of an ontology, i.e. when we model some domain of interest, reasoning allows us to ensure that we do not make any errors in our modelling. For example, when introducing a new concept assertion we may want to check that it does not violate any existing axioms.

- Once we have constructed an ontology, we can make use of reasoning algorithms to uncover hidden knowledge about the domain of interest that was not explicit before. For example, we might discover that two different concepts we included in our modelling are actually semantically equivalent.

Besides the two examples above, there exist a variety of other standard reasoning tasks for DL ontologies. For the purpose of this dissertation, we focus on the central task of *subsumption*, to which all other reasoning problems in $\mathcal{EL}^{++}$ can be reduced [Baader et al., 2005].

Definition 2.2.3 (Subsumption)**.** Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}^{++}$ ontology. We say that a concept C is *subsumed* by a concept D with respect to $\mathcal{O}$, written $\mathcal{O} \models C \sqsubseteq D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{O}$. The problem of *subsumption* is to determine whether $\mathcal{O} \models C \sqsubseteq D$ for two given concepts C and D.

Example 2.2.3. Consider the ontology $\mathcal{O} = (\mathcal{T}, \emptyset)$ with the TBox $\mathcal{T}$ from Example 2.2.2 and an empty ABox. We have that $\mathcal{O} \models \mathsf{Father} \sqsubseteq \mathsf{Parent}$. To see why, consider an arbitrary model $\mathcal{I}$ of $\mathcal{O}$ and an element $x \in \mathsf{Father}^{\mathcal{I}}$. Since $\mathcal{I}$ is a model, due to Axiom 2.6, we have that $x \in \mathsf{Male}^{\mathcal{I}} \cap (\exists \mathsf{hasChild}.\top)^{\mathcal{I}}$. It follows that $x \in (\exists \mathsf{hasChild}.\top)^{\mathcal{I}}$. But then, due to Axiom 2.5, $x \in \mathsf{Parent}^{\mathcal{I}}$.

Due to the central role of reasoning in constructing and using ontologies, much of the research in DL has been focused on developing efficient reasoning algorithms and tools. A key tradeoff that has to be made in this regard is between the expressivity of a particular DL and the complexity of reasoning in it. While the subsumption problem is known to be NP-complete or harder for many DLs [Baader et al., 2003], crucially, it is decidable in polynomial time for $\mathcal{EL}^{++}$ [Baader et al., 2005]. This has enabled $\mathcal{EL}^{++}$ to

be used as the basis of many particularly large ontologies, such as they are commonly found in the life sciences, e.g. [Ashburner et al., 2000; Schulz et al., 2009].

2.2.3 $\mathcal{EL}^{++}$ and first-order logic

The model-theoretic definition of the semantics of $\mathcal{EL}^{++}$ we have given is also known as the *direct semantics*. Alternatively, one can also specify the semantics of $\mathcal{EL}^{++}$ (and many other DLs) via a translation to first-order logic (FOL). This is interesting for two reasons: first, it shows that $\mathcal{EL}^{++}$ can be regarded as a decidable fragment of FOL, and second, it allows us to directly apply decidability and complexity results for (fragments of) FOL to DLs. While we refer the interested reader to [Baader et al., 2017, Chapter 2.6] for the exact details of this translation, we provide an example to illustrate the basic principle.

Example 2.2.4. Axiom 2.4 and Axiom 2.5 from the TBox in Example 2.2.2 can be translated into FOL sentences as follows:

$$\mathsf{Male} \sqcap \mathsf{Female} \sqsubseteq \bot \;\rightsquigarrow\; \forall x.\,(\mathsf{Male}(x) \wedge \mathsf{Female}(x) \implies \bot),$$
$$\exists \mathsf{hasChild}.\top \sqsubseteq \mathsf{Parent} \;\rightsquigarrow\; \forall x.\,(\exists y.\,\mathsf{hasChild}(x,y) \implies \mathsf{Parent}(x)).$$

Note how named concepts are translated to unary FOL predicates, role names to binary FOL predicates, and TBox axioms to universally quantified FOL sentences.

2.2.4 Relationship to knowledge graphs

Another interesting perspective on DLs is to regard them as extensions of KGs with logical background information [Kulmanov et al., 2019]. In particular, given an $\mathcal{EL}^{++}$ ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, we can represent the relational part of $\mathcal{A}$ as a KG $G_{\mathcal{A}} = (\mathcal{E}, \mathcal{R}, \mathcal{F})$ by setting $\mathcal{E} = \mathbf{I}$, $\mathcal{R} = \mathbf{R}$, and $\mathcal{F} = \{\,(a, r, b) \mid r(a,b) \in \mathcal{A}\,\}$. The other information encoded in $\mathcal{O}$ can then be seen as logical background knowledge about the entities and facts in $G_{\mathcal{A}}$.

This perspective offers an interesting insight into the connection between link prediction and logical reasoning. A reasoning algorithm that predicts role assertions of the form $r(a,b)$ (or equivalently, concept subsumptions of the form $\{a\} \sqsubseteq \exists r.\{b\}$) can

be regarded as predicting missing links in $G_{\mathcal{A}}$ that follow logically from the background knowledge. However, reasoning is not limited to only relational knowledge but can also be used to infer new background information about $G_{\mathcal{A}}$.

In contrast, most statistical link prediction techniques such as KGEs do not take logical background knowledge into account. Instead, they predict missing links solely on their statistical likelihood based on the observed connections between entities. While this approach does not provide any guarantees regarding the correctness of the predictions, it has the advantage that it is not limited to inferences that rigorously follow from a well-defined semantics.

In the next chapter, we will see how KGEs can be extended to the domain of DLs in order to enable similar kinds of statistical inference and overcome some of the drawbacks of purely logical reasoning.

3

A Spatio-Translational Embedding Model for the Description Logic $\mathcal{EL}^{++}$

In this chapter, we develop our main contribution: *Box*^{2}*EL*, a novel embedding model for the DL $\mathcal{EL}^{++}$ that represents roles using both *spatial* and *translational* representations. We begin by introducing the basic ideas behind description logic embeddings and subsequently discuss our model in comprehensive detail. Finally, we show that our model is *sound* in that it corresponds to logical models of $\mathcal{EL}^{++}$.

3.1 Description logic embeddings and geometric models

Description logic embeddings (DLE) extend the idea behind KGEs to the domain of DL. The basic principle is the same: given an ontology $\mathcal{O}$ with signature $\Sigma = (\mathbf{C}, \mathbf{R}, \mathbf{I})$, we want to embed the classes, roles, and individuals in $\mathcal{O}$ in a continuous latent vector space $\mathbb{R}^n$, in which we can then perform simple geometric operations to perform tasks such as subsumption prediction. However, in contrast to the KG setting, the embeddings we learn for an ontology must not only be based on statistical similarities, but crucially also have to preserve its logical semantics.

We follow the approach proposed by Kulmanov et al. [2019] and expanded upon in subsequent work [Mondal et al., 2021; Mohapatra et al., 2021; Xiong et al., 2022; Peng et al., 2022], which involves leveraging the axioms in the $\mathcal{EL}^{++}$ ontology $\mathcal{O}$ to

learn embeddings that correspond to *geometric models* of $\mathcal{O}$; that is, (logical) models with an interpretation domain $\Delta = \mathbb{R}^n$. If we are able to find such embeddings, they will by definition preserve the semantics of $\mathcal{O}$ and are as such well-suited for sub-symbolic reasoning tasks.

In order to define the geometric models we want to learn, we have to specify how concepts, roles, and individuals are mapped to the geometric embedding space $\mathbb{R}^n$. As a simplification, we omit mapping the individuals explicitly by first eliminating the ABox from the ontology using the transformation rules described in Section 2.2.1. Let us denote the set of all atomic concepts including nominals as

$$\mathbf{CI} = \mathbf{C} \cup \bigcup_{a \in \mathbf{I}} \{\{a\}\}.$$

Formally, we now have to define how elements of $\mathbf{CI}$ and $\mathbf{R}$ are represented in $\mathbb{R}^n$.

3.2 Concept representation

Concepts correspond to sets of objects in the interpretation domain, and we thus have to model them as subsets of $\mathbb{R}^n$. Existing DLE methods usually represent concepts as *regions* in the embedding space. Different such regions have been proposed, ranging from convex cones [Özçep et al., 2020] to linear subspaces of complex vector spaces [Garg et al., 2019]. In the context of $\mathcal{EL}^{++}$ embeddings, the two most prominent approaches in the literature model concepts as either *n-balls* [Kulmanov et al., 2019; Mondal et al., 2021; Mohapatra et al., 2021] or *boxes*, i.e. *axis-aligned hyperrectangles* [Xiong et al., 2022; Peng et al., 2022].

In Box^{2}EL we employ the latter representation of concepts as boxes, since they have the conceptual advantage over n-balls that they are closed under intersection [Xiong et al., 2022; Peng et al., 2022]. That is, the intersection of two boxes is guaranteed to also be a box, whereas intersecting two n-balls may yield a shape that is not an n-ball. This property of intersectional closure is useful for representing the conjunction of concepts in the embedding space, as we demonstrate in the following example.

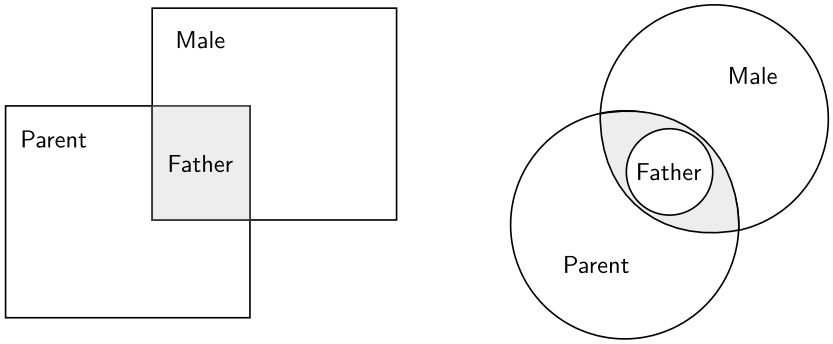


Figure 3.1: Intersections of concepts in the embedding space. *(Left)* Intersecting concepts represented as boxes in the embedding space yields a box, and the axiom $\mathsf{Parent} \sqcap \mathsf{Male} \sqsubseteq \mathsf{Father}$ can therefore naturally be represented. *(Right)* The intersection of two concepts represented as n-balls may not be an n-ball, which leads to problems when modelling conjunction. Adapted from [Xiong et al., 2022].

Example 3.2.1 (Xiong et al., 2022)**.** Suppose we want to represent the following TBox in the embedding space:

$$\begin{aligned}\mathcal{T} = \{&\mathsf{Parent} \sqcap \mathsf{Male} \sqsubseteq \mathsf{Father},\\ &\mathsf{Father} \sqsubseteq \mathsf{Male},\\ &\mathsf{Father} \sqsubseteq \mathsf{Parent}\}.\end{aligned}$$

As illustrated in Figure 3.1, when representing concepts as boxes, we can naturally model the concept Father as the intersection of Parent and Male. On the other hand, a concept representation based on n-balls cannot accurately capture the semantics of $\mathcal{T}$.

We now formally define boxes and a number of operations on them.

Definition 3.2.1 (Box)**.** An n-dimensional *box* A is a subset of $\mathbb{R}^n$ for which there exist vectors $\boldsymbol{l}_A \in \mathbb{R}^n$ and $\boldsymbol{u}_A \in \mathbb{R}^n$ with $\boldsymbol{l}_A \leq \boldsymbol{u}_A$, such that

$$A = \{\, \boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{l}_A \leq \boldsymbol{x} \leq \boldsymbol{u}_A \,\},$$

where $\leq$ is applied element-wise. The vectors $\boldsymbol{l}_A$ and $\boldsymbol{u}_A$ are the *lower* and *upper* corner of A, respectively. Sometimes we will write $[\boldsymbol{l}_A, \boldsymbol{u}_A]$ as shorthand for A. We denote the set of n-dimensional boxes as $\mathbb{B}^n$.

Centres and offsets. Given a box A, we can calculate its *centre* $c(A)$ and *offset* $o(A)$ as follows:

$$c(A) = \frac{\boldsymbol{l}_A + \boldsymbol{u}_A}{2} \tag{3.1}$$

and

$$o(A) = \frac{\boldsymbol{u}_A - \boldsymbol{l}_A}{2}. \tag{3.2}$$

We can also go into the other direction and recover the lower and upper corner of a box from $c(A)$ and $o(A)$. It follows immediately from the definitions that

$$\begin{aligned} & c(A) - o(A) \\ & \quad = \frac{\boldsymbol{l}_A + \boldsymbol{u}_A}{2} - \frac{\boldsymbol{u}_A - \boldsymbol{l}_A}{2} \\ & \quad = \boldsymbol{l}_A \end{aligned} \tag{3.3}$$

and similarly

$$\boldsymbol{u}_A = c(A) + o(A). \tag{3.4}$$

Translation. The *translation* of a box A along a vector $\boldsymbol{t} \in \mathbb{R}^n$ is defined as

$$A + \boldsymbol{t} = [\boldsymbol{l}_A + \boldsymbol{t},\ \boldsymbol{u}_A + \boldsymbol{t}].$$

Intersection. Given two boxes A and B, their intersection $A \cap B$ can be computed as follows:

$$\begin{aligned} \boldsymbol{l}_{A \cap B} &= \max\{\boldsymbol{l}_A, \boldsymbol{l}_B\}, \\ \boldsymbol{u}_{A \cap B} &= \min\{\boldsymbol{u}_A, \boldsymbol{u}_B\}, \end{aligned}$$

where max and min are applied element-wise. This is illustrated in Figure 3.2.

Model parameters. Box^{2}EL represents every concept in $\mathbf{CI}$ as a box given by its centre and offset. Formally, we denote all parameters of Box^{2}EL by a vector $\boldsymbol{\theta}$ and define the function $\text{Box}_{\boldsymbol{\theta}} \colon \mathbf{CI} \to \mathbb{B}^n$, which returns the box embedding of a given concept. In total, we require $2n(|\mathbf{C}| + |\mathbf{I}|)$ parameters to store all concept embeddings.

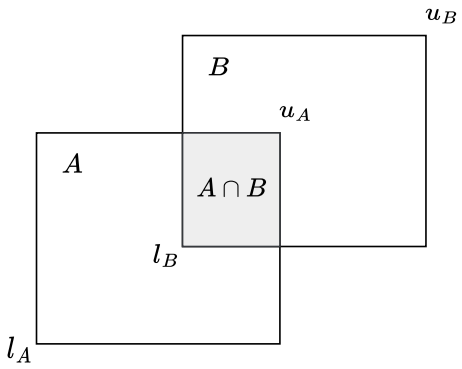


Figure 3.2: Computing the intersection of two boxes. The intersection of A and B is indicated by the shaded area, whose lower corner is given by $\max\{\boldsymbol{l}_A, \boldsymbol{l}_B\} = \boldsymbol{l}_B$ and whose upper corner is $\min\{\boldsymbol{u}_A, \boldsymbol{u}_B\} = \boldsymbol{u}_A$. Since min and max are applied element-wise, the corners of the intersection box $A \cap B$ need not in general be equal to one of the corners of either A or B.

3.3 Role representation

We next have to specify how to represent roles in the embedding space. Since our aim is to learn geometric models, we need to map roles to binary relations $r \subseteq \mathbb{R}^n \times \mathbb{R}^n$. All existing $\mathcal{EL}^{++}$ embedding models we are aware of represent these binary relations by making use of some variation of the TransE KGE model [Bordes et al., 2013] described in Section 2.1.3. However, this has several limitations as we show next.

3.3.1 Limitations of translational embedding models

Recall that in the KG setting, TransE represents entities and relations as n-dimensional vectors and learns embeddings such that

$$\boldsymbol{h} + \boldsymbol{r} \approx \boldsymbol{t}$$

for triples that are likely to be true. This approach can readily be applied to DLEs by similarly representing roles as translation vectors that induce a binary relation

$$\{\, (\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{R}^n \times \mathbb{R}^n \mid \boldsymbol{x} + \boldsymbol{r} = \boldsymbol{y} \,\},$$

where $\boldsymbol{r}$ is the translation vector for the role r. Analogously to TransE, an axiom of the form $C \sqsubseteq \exists r.D$ then holds in the embedding space if

$$\mathrm{Box}_{\boldsymbol{\theta}}(C) + \boldsymbol{r} \subseteq \mathrm{Box}_{\boldsymbol{\theta}}(D).$$

While this role representation is intuitive and relatively effective, it inherits one of the fundamental limitations of TransE: its inability to model *one-to-many*, *many-to-one*, or *many-to-many* relationships, as noted in e.g. [Z. Wang et al., 2014; Lin et al., 2015; Abboud et al., 2020]. We illustrate the problem with an example.

Example 3.3.1. Let $\mathcal{T}$ be the following $\mathcal{EL}^{++}$ TBox:

$$\begin{aligned}\mathcal{T} = \{&\mathsf{Mother} \sqcap \mathsf{Father} \sqsubseteq \bot,\\ &\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Mother},\\ &\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Father}\}.\end{aligned}$$

The TransE role representation requires that $\mathrm{Box}_{\boldsymbol{\theta}}(\mathsf{Child}) + \mathsf{hasParent} \subseteq \mathrm{Box}_{\boldsymbol{\theta}}(\mathsf{Mother})$ and $\mathrm{Box}_{\boldsymbol{\theta}}(\mathsf{Child}) + \mathsf{hasParent} \subseteq \mathrm{Box}_{\boldsymbol{\theta}}(\mathsf{Father})$. However, since the first axiom in $\mathcal{T}$ states that Mother and Father must be disjoint, this can only be fulfilled if $\mathrm{Box}_{\boldsymbol{\theta}}(\mathsf{Child}) = \emptyset$. The embedding model thus clearly does not align with the semantics of $\mathcal{T}$.

Along similar lines, TransE is incapable of capturing symmetric relationships of the form $\{C \sqsubseteq \exists r.D,\ D \sqsubseteq \exists r.C\}$ without making the embeddings of C and D equal [Sun et al., 2019]. Clearly, we need a more expressive representation of roles in the embedding space to accurately capture the range of constructs expressible in $\mathcal{EL}^{++}$.

3.3.2 Traditional extensions of TransE

In the KGE literature, a variety of extensions to TransE have been proposed to resolve the issues identified above [Z. Wang et al., 2014; Lin et al., 2015; G. Ji et al., 2015; G. Ji et al., 2016]. The general idea behind most of these extensions is to separate the embedding space of entities and relations. For example, the TransH model [Z. Wang et al., 2014] associates every relation with a hyperplane and a translation operation on that hyperplane. Scores for a triple $(\mathsf{h}, \mathsf{r}, \mathsf{t})$ are then computed by first calculating the projections $\boldsymbol{h}_\perp$ and $\boldsymbol{t}_\perp$ of h and t onto the hyperplane associated with r and then applying the normal TransE scoring function in that projection space, i.e.

$$s(\mathsf{h}, \mathsf{r}, \mathsf{t}) = -\|\boldsymbol{h}_\perp + \boldsymbol{r} - \boldsymbol{t}_\perp\|.$$

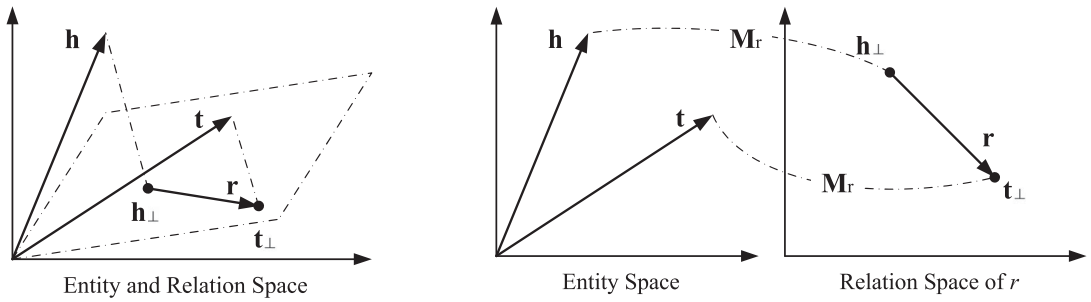


Figure 3.3: An illustration of TransH and TransR [Q. Wang et al., 2017]. *(Left)* TransH associates every relation with a hyperplane and projects entity embeddings onto that hyperplane before applying the TransE scoring function. *(Right)* TransR learns a projection matrix $\boldsymbol{M}_r$ for every relation, which is used to similarly project entity embeddings into the relation space of r.

With this formulation, the embedding $\boldsymbol{e}_\perp$ of an entity crucially depends on the relation r. To see how this addresses the shortcomings of TransE, assume that the relation r is a *one-to-many* relation. While TransE would require that the embeddings for every possible tail of a tuple (h, r) be equal, in TransH, we only have that the *projected* tails $\boldsymbol{t}_\perp$ must be equal. For a different relation that is not a *one-to-many* relation, the model can therefore still learn distinct embeddings for the possible tails of (h, r).

Other approaches such as TransR [Lin et al., 2015] and its variations follow this idea of learning embeddings in two separate vector spaces and projecting entities into the relevant relation space before applying a translational scoring function. An illustration of TransH and TransR is given in Figure 3.3.

While these approaches have been employed successfully in the KGE setting, there are several issues when trying to apply them to DLEs. The most immediately obvious problem is the question of how to preserve the representation of concepts when projecting them to the relation space. For instance, if we try to project an n-dimensional box onto a hyperplane using the TransH approach, we will end up with an $(n-1)$-dimensional shape that may not be a box anymore.

The same issue arises when mapping concepts back from the relation space to the space of concept embeddings, which is required for existentially restricted concepts. To illustrate, consider the concept $\exists r.D$. Using TransE, we can easily and naturally represent this concept in the embedding space by translating the embedding of D

backwards along $\boldsymbol{r}$, i.e. as $\text{Box}_{\boldsymbol{\theta}}(D) - \boldsymbol{r}$. If we try to use TransH to represent roles on the other hand, we would have to apply this translation along the reverse of $\boldsymbol{r}$ in the corresponding *relation space* and then map the result back to the space of concept embeddings. However, the resulting region in the concept space is not guaranteed to be a box and thus does not conform to our representation of concepts.

While it is possible to find ways around this problem, for example by heavily limiting the expressiveness of the representation of roles by only allowing axis-aligned hyperplanes, we adopt a different approach that does not require making such compromises for Box^{2}EL.

3.3.3 A spatio-translational model for roles

Instead of using one of the direct replacements of TransE, we adapt the expressive KGE model *BoxE* [Abboud et al., 2020] to the domain of DLEs. We first describe BoxE in the KG setting, before we show how its relational model can be used in the context of DLEs.

BoxE

BoxE is a KGE model that, like TransE, embeds entities as points in the embedding space $\mathbb{R}^n$. Instead of using a translational approach to represent relations, BoxE models relations using *spatial* representations. In particular, each relation is associated with two *boxes* in the embedding space: a *head* and a *tail* box. A triple $(\mathsf{h}, \mathsf{r}, \mathsf{t})$ is considered to be true if

$$\boldsymbol{h} \in \boldsymbol{r}_h \quad \text{and} \quad \boldsymbol{t} \in \boldsymbol{r}_t,$$

where $\boldsymbol{e}$ denotes the embedding for entity e, and $\boldsymbol{r}_h$ and $\boldsymbol{r}_t$ are the head and tail boxes of r, respectively.

However, this purely spatial representation of relations is still quite restricted in its current form, since it relates *every* entity whose embedding lies in the head box of a relation to *every* entity with an embedding in the corresponding tail box. To illustrate, if we try to model the simple set of facts $\{(\mathsf{a}, \mathsf{r}, \mathsf{b}), (\mathsf{c}, \mathsf{r}, \mathsf{d})\}$ with this approach, the model will also consider the facts $(\mathsf{a}, \mathsf{r}, \mathsf{d})$ and $(\mathsf{c}, \mathsf{r}, \mathsf{b})$ to be true.

To overcome this limitation, BoxE introduces an additional *bump vector* $\boldsymbol{b}_e$ for each entity $\mathtt{e}$. For a fact $(\mathtt{h}, \mathtt{r}, \mathtt{t})$, these bump vectors intuitively modify the position of $\mathtt{h}$ and $\mathtt{t}$ by "bumping" the corresponding points in the vector space before they are compared to the head or tail box of $\mathtt{r}$, respectively. Formally, with the introduction of bump vectors a fact is now considered true if

$$\boldsymbol{h} + \boldsymbol{b}_t \in \boldsymbol{r}_h \quad \text{and} \quad \boldsymbol{t} + \boldsymbol{b}_h \in \boldsymbol{r}_t.$$

The embeddings of entities are therefore dynamic and depend on the particular triple that is being considered. It is easy to see that bump vectors allow us to correctly represent the set of facts from above without making any additional facts true.

This *spatio-translational* approach of embedding relations by combining a spatial box representation with translational bumps turns out to yield a very strong KGE model that overcomes the limitations of TransE in the KG setting. Furthermore, one can show that BoxE does not suffer from any similar kinds of shortcomings — it is a *fully expressive* KGE model, i.e. can correctly capture an arbitrary set of facts in a KG [Abboud et al., 2020].

Adapting BoxE to the DLE setting

The relational model of BoxE can almost directly be applied to the DLE setting: we similarly associate every role $r \in \mathbf{R}$ with a head box $\mathrm{Head}_{\boldsymbol{\theta}}(r)$ and a tail box $\mathrm{Tail}_{\boldsymbol{\theta}}(r)$, inducing the binary relation

$$\mathrm{Head}_{\boldsymbol{\theta}}(r) \times \mathrm{Tail}_{\boldsymbol{\theta}}(r) \subseteq \mathbb{R}^n \times \mathbb{R}^n$$

in the embedding space. We furthermore introduce bump vectors $\mathrm{Bump}_{\boldsymbol{\theta}}(C)$ for every concept $C \in \mathbf{CI}$, which, as in BoxE, enable a dynamic representation of concept embeddings. An axiom of the form $C \sqsubseteq \exists r.D$ is thus considered to hold if

$$\mathrm{Box}_{\boldsymbol{\theta}}(C) + \mathrm{Bump}_{\boldsymbol{\theta}}(D) \subseteq \mathrm{Head}_{\boldsymbol{\theta}}(r)$$

and

$$\mathrm{Box}_{\boldsymbol{\theta}}(D) + \mathrm{Bump}_{\boldsymbol{\theta}}(C) \subseteq \mathrm{Tail}_{\boldsymbol{\theta}}(r),$$

and we can similarly represent other axioms in the embedding space.

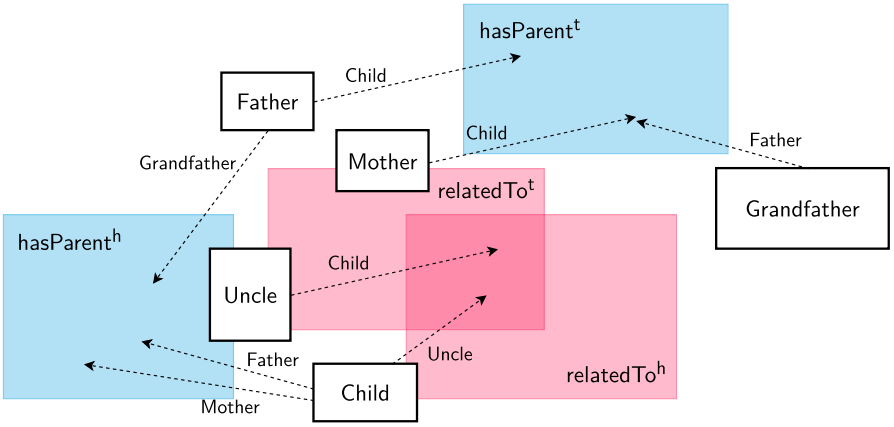


Figure 3.4: An illustration of Box^{2}EL. White boxes represent concept embeddings, whereas role embeddings are illustrated as coloured boxes and labelled as r^h or r^t for the head or tail box of r, respectively. Bump vectors are drawn as arrows and labelled with the corresponding concept. In this configuration, the axiom $\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Father}$ is modelled as true, since the box embedding of Child bumped by the bump vector of Father lies in $\mathsf{hasParent}^h$, and similarly the embedding of Father bumped by Child's bump vector is in $\mathsf{hasParent}^t$. Conversely, the axiom $\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Grandfather}$ does not hold.

Since we use boxes not only for the representation of concepts, but also to represent relations, we call our method *Box^{2}EL*.

Example 3.3.2. Consider the following TBox $\mathcal{T}$:

$$\mathcal{T} = \{\mathsf{Mother} \sqcap \mathsf{Father} \sqsubseteq \bot, \quad \mathsf{Father} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Grandfather},$$
$$\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Mother}, \quad \mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Father},$$
$$\mathsf{Child} \sqsubseteq \exists \mathsf{relatedTo}.\mathsf{Uncle}, \quad \mathsf{Uncle} \sqsubseteq \exists \mathsf{relatedTo}.\mathsf{Child}\}.$$

Figure 3.4 illustrates a Box^{2}EL model that correctly represents the axioms in $\mathcal{T}$.

The previous example demonstrates the expressive power of Box^{2}EL and shows how it is able to overcome the shortcomings of TransE. In particular, note that we can now successfully model *one-to-many* relationships such as hasParent and symmetric relationships like relatedTo.

Model complexity. In order to represent the head and tail boxes for every relation and a bump vector per concept, we require $4n|\mathbf{R}| + n(|\mathbf{C}| + |\mathbf{I}|)$ parameters. Together with the parameters needed to store the concept embeddings (see Section 3.2), the total space complexity of Box^{2}EL is thus $O\big(n(3(|\mathbf{C}| + |\mathbf{I}|) + 4|\mathbf{R}|)\big)$.

3.4 Training procedure

So far, we have defined how Box^{2}EL represents concepts and roles in the embedding space, and gained some intuition how these representations could be used to encode the axioms of an ontology. We now describe the training procedure that learns embeddings for a given ontology in detail. Subsequently, we will formally show that the learnt embeddings corresponds to a geometric model of the ontology.

3.4.1 Normal forms

In order to learn embeddings for an $\mathcal{EL}^{++}$ ontology $\mathcal{O}$, we first eliminate the ABox, as previously discussed. Afterwards, we transform every axiom in $\mathcal{O}$ into a normal form using the normalisation procedure described in [Baader et al., 2005]. The first four normal forms involve subsumptions between concepts:

$$C \sqsubseteq D \tag{NF1}$$

$$C \sqcap D \sqsubseteq E \tag{NF2}$$

$$C \sqsubseteq \exists r.D \tag{NF3}$$

$$\exists r.C \sqsubseteq D \tag{NF4}$$

The remaining normal forms specify that concepts are not satisfiable:

$$C \sqcap D \sqsubseteq \bot \tag{NF5}$$

$$\exists r.C \sqsubseteq \bot \tag{NF6}$$

$$C \sqsubseteq \bot \tag{NF7}$$

Crucially, the normalised ontology is a *conservative extension* of $\mathcal{O}$, that is, every model of the normalised ontology is also a model of $\mathcal{O}$ [Baader et al., 2005]. In particular, if we learn a geometric model for the normalised ontology, it will also be a model of $\mathcal{O}$.

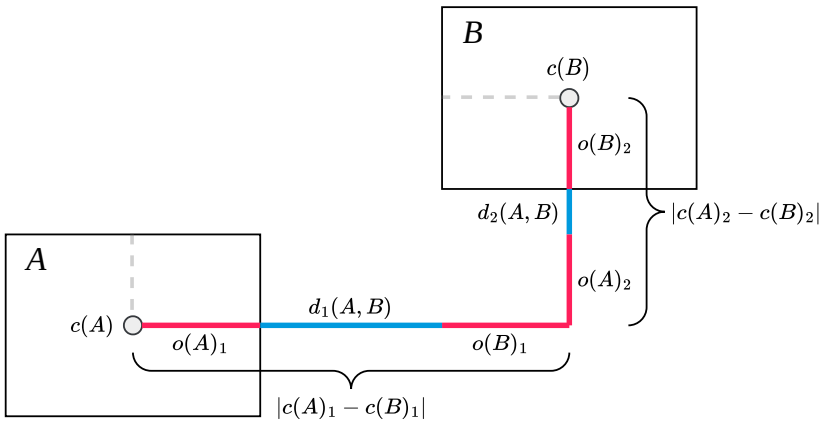


Figure 3.5: Calculating the distance between two boxes. The blue lines indicate the element-wise distances $d_1(A, B)$ and $d_2(A, B)$ between A and B. They are obtained by subtracting the offsets $o(A)$ and $o(B)$, represented by red lines, from the element-wise distances of the centres of the boxes. Inspired by [Peng et al., 2022].

3.4.2 Loss functions

We follow the framework of Kulmanov et al. [2019] and introduce a separate loss function for every normal form. During training, we then optimise the corresponding loss function for every axiom in the ontology.

There exist two main approaches for defining loss functions in the literature: *distance*- and *volume*-based formulations. In the distance-based approach, utilised for example in [Kulmanov et al., 2019; Peng et al., 2022], we aim to minimise the *distance* between the embeddings of related concepts. The volume-based definition, due to Xiong et al. [2022], on the other hand uses the *volume* of concept embeddings (or rather a modified *soft* volume) as the basis of the various loss functions. Since we empirically find the distance-based formulation to work better, as is also suggested by previous results [Peng et al., 2022], we employ the first approach.

Distance between boxes. In the following, we will often need to calculate the element-wise distance $d(A, B)$ between two boxes A and B. As illustrated in Figure 3.5, it can be computed as

$$d(A, B) = |c(A) - c(B)| - o(A) - o(B).$$

Inclusion loss. We first define a general inclusion loss $\mathcal{L}_{\subseteq}(A, B)$ that encourages the box A to be contained in the box B for two arbitrary boxes $A, B \in \mathbb{B}^n$, which will be helpful when defining the axiom-specific loss functions.

In order for A to be completely contained in B, for every dimension the side that is the furthest away from B needs to be inside B. From Figure 3.5, we can see that this exactly the case when

$$d_k(A, B) + 2o(A)_k \leq 0$$

for every dimension $1 \leq k \leq n$. Consequently, we define the inclusion loss $\mathcal{L}_{\subseteq}(A, B)$ as

$$\mathcal{L}_{\subseteq}(A, B) = \|\max\{\mathbf{0},\, d(A, B) + 2o(A) - \gamma\}\|,$$

where the max function and the subtraction of γ is applied element-wise. Note that we have introduced a margin hyperparameter γ, which allows the loss to become 0 even if A is not precisely contained in B, as long as it lies within γ-distance in each dimension.

We now formally show that this loss ensures that A lies within B.

Proposition 3.4.1. *Let A and B be boxes in $\mathbb{B}^n$ and $\gamma \leq 0$. If $\mathcal{L}_{\subseteq}(A, B) = 0$, then $A \subseteq B$.*

Proof. We show the proposition by proving that $\boldsymbol{l}_B \leq \boldsymbol{l}_A$ and $\boldsymbol{u}_A \leq \boldsymbol{u}_B$. Assume $\mathcal{L}_{\subseteq}(A, B) = 0$. We have that

$$d(A, B) + 2o(A) - \gamma \leq 0$$
$$|c(A) - c(B)| + o(A) - o(B) - \gamma \leq 0$$

and thus

$$|c(A) - c(B)| + o(A) - o(B) \leq \gamma \leq 0.$$

Now fix an arbitrary dimension k such that $1 \leq k \leq n$. We distinguish two cases:

Case 1: $c(A)_k \geq c(B)_k$. We eliminate the absolute value function and use Equation 3.4 to obtain

$$\boldsymbol{u}_{A,k} - \boldsymbol{u}_{B,k} \leq 0$$
$$\boldsymbol{u}_{A,k} \leq \boldsymbol{u}_{B,k}.$$

Since $c(A)_k \geq c(B)_k$, we furthermore have by Equation 3.1

$$\begin{aligned} \frac{\boldsymbol{l}_{A,k} + \boldsymbol{u}_{A,k}}{2} &\geq \frac{\boldsymbol{l}_{B,k} + \boldsymbol{u}_{B,k}}{2} \\ \boldsymbol{l}_{A,k} &\geq \boldsymbol{l}_{B,k} + \underbrace{\boldsymbol{u}_{B,k} - \boldsymbol{u}_{A,k}}_{\geq 0} \\ \boldsymbol{l}_{A,k} &\geq \boldsymbol{l}_{B,k}. \end{aligned}$$

Case 2: $c(A)_k \leq c(B)_k$. Similarly to the first case, we eliminate the absolute value function and use Equation 3.3 to obtain

$$\begin{aligned} -\boldsymbol{l}_{A,k} + \boldsymbol{l}_{B,k} &\leq 0 \\ \boldsymbol{l}_{B,k} &\leq \boldsymbol{l}_{A,k}. \end{aligned}$$

Because $c(A)_k \leq c(B)_k$ and using Equation 3.1, we have

$$\begin{aligned} \frac{\boldsymbol{l}_{A,k} + \boldsymbol{u}_{A,k}}{2} &\leq \frac{\boldsymbol{l}_{B,k} + \boldsymbol{u}_{B,k}}{2} \\ \underbrace{\boldsymbol{l}_{A,k} - \boldsymbol{l}_{B,k}}_{\geq 0} + \boldsymbol{u}_{A,k} &\leq \boldsymbol{u}_{B,k} \\ \boldsymbol{u}_{A,k} &\leq \boldsymbol{u}_{B,k}. \end{aligned}$$

Now, consider an arbitrary point $\boldsymbol{a} \in A$. By Definition 3.2.1 we have that $\boldsymbol{l}_A \leq \boldsymbol{a} \leq \boldsymbol{u}_A$. But then

$$\boldsymbol{l}_B \leq \boldsymbol{l}_A \leq \boldsymbol{a} \leq \boldsymbol{u}_A \leq \boldsymbol{u}_B$$

and thus $\boldsymbol{a} \in B$. □

Disjoint loss. We define a further generic loss function $\mathcal{L}_d(A, B)$ that ensures that boxes A and B are disjoint. Intuitively, the loss makes the element-wise distance $d(A, B)$ greater than 0 in all dimensions. It is defined as follows:

$$\mathcal{L}_d(A, B) = \|\max\{\mathbf{0}, -(d(A, B) + \gamma)\}\|. \tag{3.5}$$

Similarly to before, we prove that the loss corresponds to our intuition.

Proposition 3.4.2. *Let A and B be boxes in $\mathbb{B}^n$ and $\gamma \leq 0$. If $\mathcal{L}_d(A, B) = 0$, then $A \cap B = \emptyset$.*

Proof. The proof is similar to that of Proposition 3.4.1. Assume $\mathcal{L}_d(A, B) = 0$. We have that

$$-(d(A,B) + \gamma) \leq 0$$
$$-(|c(A) - c(B)| - o(A) - o(B) + \gamma) \leq 0$$

and therefore

$$|c(A) - c(B)| - o(A) - o(B) \geq -\gamma \geq 0.$$

We again fix a dimension k such that $1 \leq k \leq n$ and distinguish two cases:

Case 1: $c(A)_k \geq c(B)_k$. By eliminating the absolute value function and using Equations 3.3 and 3.4 we obtain

$$\begin{aligned} \boldsymbol{l}_{A,k} - \boldsymbol{u}_{B,k} &\geq 0 \\ \boldsymbol{l}_{A,k} &\geq \boldsymbol{u}_{B,k}. \end{aligned} \tag{3.6}$$

Case 2: $c(A)_k \leq c(B)_k$. Analogously to Case 1, we have

$$\begin{aligned} \boldsymbol{l}_{B,k} - \boldsymbol{u}_{A,k} &\geq 0 \\ \boldsymbol{l}_{B,k} &\geq \boldsymbol{u}_{A,k}. \end{aligned} \tag{3.7}$$

Now consider an arbitrary point $\boldsymbol{a} \in A$. From the case analysis above, we know that either $\boldsymbol{l}_{A,k} \geq \boldsymbol{u}_{B,k}$ or $\boldsymbol{l}_{B,k} \geq \boldsymbol{u}_{A,k}$. However, in both cases $\boldsymbol{a}$ cannot be in B. □

We are now ready to state the loss functions for the different normal forms in $\mathcal{EL}^{++}$ ontologies. Since we also represent concepts as boxes and use a distance-based approach similar to Peng et al. [2022], some of our loss functions are equivalent to theirs.

First normal form (NF1). Given an axiom $C \sqsubseteq D$, we want to learn embeddings such that $\text{Box}_{\boldsymbol{\theta}}(C) \subseteq \text{Box}_{\boldsymbol{\theta}}(D)$, since this corresponds to the semantics of concept inclusion. Therefore, we define the loss for the first normal form as simply the inclusion loss:

$$\mathcal{L}_1(C, D; \boldsymbol{\theta}) = \mathcal{L}_{\subseteq}(\text{Box}_{\boldsymbol{\theta}}(C), \text{Box}_{\boldsymbol{\theta}}(D)).$$

Second normal form (NF2). For an axiom of the form $C \sqcap D \sqsubseteq E$, we similarly require that the intersection of the box embeddings of C and D is a subset of the box associated with E. The intersection of two boxes can be easily computed as discussed in Section 3.2 and so we have

$$\mathcal{L}_2(C, D, E; \boldsymbol{\theta}) = \mathcal{L}_{\subseteq}\Big(\text{Box}_{\boldsymbol{\theta}}(C) \cap \text{Box}_{\boldsymbol{\theta}}(D), \text{Box}_{\boldsymbol{\theta}}(E)\Big).$$

However, this formulation is problematic since it can be easily minimised to 0 by setting $\text{Box}_{\boldsymbol{\theta}}(C)$ and $\text{Box}_{\boldsymbol{\theta}}(D)$ to be disjoint. While disjoint embeddings for C and D would technically not violate the semantics, usually an axiom of the form $C \sqcap D \sqsubseteq \bot$ would have been used directly if it had been the intention that C and D should be disjoint. Therefore, we introduce the following *non-empty loss* for arbitrary boxes A and B

$$\mathcal{L}_{NE}(A, B) = \|\max\{\mathbf{0}, \max\{\boldsymbol{l}_A, \boldsymbol{l}_B\} - \min\{\boldsymbol{u}_A, \boldsymbol{u}_B\}\}\|,$$

which encourages $A \cap B$ to be non-empty. Intuitively, the loss ensures that all elements of the offset vector $o(A \cap B)$ are positive.

Overall, the loss for axioms in the second normal form is given by

$$\mathcal{L}_2(C, D, E; \boldsymbol{\theta}) = \mathcal{L}_{\subseteq}\Big(\text{Box}_{\boldsymbol{\theta}}(C) \cap \text{Box}_{\boldsymbol{\theta}}(D), \text{Box}_{\boldsymbol{\theta}}(E)\Big) + \mathcal{L}_{NE}\Big(\text{Box}_{\boldsymbol{\theta}}(C), \text{Box}_{\boldsymbol{\theta}}(D)\Big).$$

Third normal form (NF3). The third normal form involves existential restriction, and we thus need to define a loss that takes the novel role representation of Box^{2}EL into account. From the discussion in Section 3.3.3, we have that for an axiom of the form $C \sqsubseteq \exists r.D$ we need to learn embeddings such that $\text{Box}_{\boldsymbol{\theta}}(C) + \text{Bump}_{\boldsymbol{\theta}}(D) \subseteq \text{Head}_{\boldsymbol{\theta}}(r)$ and $\text{Box}_{\boldsymbol{\theta}}(D) + \text{Bump}_{\boldsymbol{\theta}}(C) \subseteq \text{Tail}_{\boldsymbol{\theta}}(r)$. This requirement is captured by the following loss function:

$$\begin{aligned}\mathcal{L}_3(C, r, D; \boldsymbol{\theta}) = \frac{1}{2}\Big(&\mathcal{L}_{\subseteq}(\text{Box}_{\boldsymbol{\theta}}(C) + \text{Bump}_{\boldsymbol{\theta}}(D), \text{Head}_{\boldsymbol{\theta}}(r)) \\ &+ \mathcal{L}_{\subseteq}(\text{Box}_{\boldsymbol{\theta}}(D) + \text{Bump}_{\boldsymbol{\theta}}(C), \text{Tail}_{\boldsymbol{\theta}}(r))\Big).\end{aligned}$$

Fourth normal form (NF4). For an axiom of the form $\exists r.C \sqsubseteq D$, we need to ensure that all points in the embedding space that are connected to C via the role r are contained in $\text{Box}_{\boldsymbol{\theta}}(D)$. It can be easily seen from our geometric representation that the set of these points is contained in $\text{Head}_{\boldsymbol{\theta}}(r) - \text{Bump}_{\boldsymbol{\theta}}(C)$. We therefore define the loss for the fourth normal form as

$$\mathcal{L}_4(r, C, D; \boldsymbol{\theta}) = \mathcal{L}_{\subseteq}(\text{Head}_{\boldsymbol{\theta}}(r) - \text{Bump}_{\boldsymbol{\theta}}(C), \text{Box}_{\boldsymbol{\theta}}(D)).$$

Fifth normal form (NF5). Axioms of the fifth normal form $C \sqcap D \sqsubseteq \bot$ state that the concepts C and D have to be disjoint. Consequently, we define the corresponding loss as

$$\mathcal{L}_5(C, D; \boldsymbol{\theta}) = \mathcal{L}_d(\text{Box}_{\boldsymbol{\theta}}(C), \text{Box}_{\boldsymbol{\theta}}(D)).$$

Sixth normal form (NF6). The sixth normal form requires that a concept $\exists r.C$ be unsatisfiable. As with most previous approaches [Kulmanov et al., 2019; Peng et al., 2022], one limitation of our method is that we can not accurately capture this requirement in the embedding space. As an approximation, we define the loss for the sixth normal form as

$$\mathcal{L}_6(r, C; \boldsymbol{\theta}) = \|o(\text{Head}_{\boldsymbol{\theta}}(r))\|.$$

While this does ensure that $\exists r.C$ becomes unsatisfiable, it obviously does not precisely correspond to the desired semantics, since the loss will also make any concept $\exists r.C'$ with $C' \neq C$ unsatisfiable. However, we note that this normal form does not seem to be very common in practice; indeed, we do not find any NF6 axioms in any of the datasets we consider in our empirical evaluation in Chapter 4.

Seventh normal form (NF7). Finally, for axioms of the form $C \sqsubseteq \bot$ the following loss encourages $\text{Box}_{\boldsymbol{\theta}}(C)$ to be empty:

$$\mathcal{L}_7(C; \boldsymbol{\theta}) = \|o(\text{Box}_{\boldsymbol{\theta}}(C))\|.$$

Total loss. We have now defined a loss function for every possible axiom in a normalised ontology $\mathcal{O}$. The total loss of the embeddings $\boldsymbol{\theta}$ with respect to $\mathcal{O}$ is then simply given as the sum of the squares of the individual loss functions for every axiom in $\mathcal{O}$. In practice, we augment this loss using a negative sampling procedure and a regularisation term, as we describe next.

3.4.3 Negative sampling

While the embeddings could in theory be directly optimised with the loss functions we have specified, it is common to additionally employ a form of negative sampling during training in order to further improve the quality of the learnt embeddings [Kulmanov et al., 2019; Xiong et al., 2022; Peng et al., 2022]. We follow previous work and generate negative samples for axioms in the third normal form analogously to negative sampling in KGEs. In particular, for an axiom of the form $C \sqsubseteq \exists r.D$ we generate a set of corrupted axioms by replacing either C or D with a randomly selected different concept.

An intuitive choice regarding the loss function for negative training examples would be to use the disjoint loss from Equation 3.5, which ensures that the element-wise distance $d(A, B)$ between two boxes A and B is positive in all dimensions. However, in practice we find a loss formulation based on the *minimal distance* between A and B to yield better results.

Minimal distance between boxes. Let A and B be boxes in $\mathbb{B}^n$. Recall that the function $d(A, B)$ computes the element-wise distance between A and B. As illustrated in Figure 3.6, the minimal distance between any two points in A and B can be computed as

$$\|\max\{\mathbf{0},\, d(A, B)\}\|.$$

For our loss, we again add the margin hyperparameter γ to the minimal distance, yielding the following function μ:

$$\mu(A, B) = \|\max\{\mathbf{0},\, d(A, B) + \gamma\}\|.$$

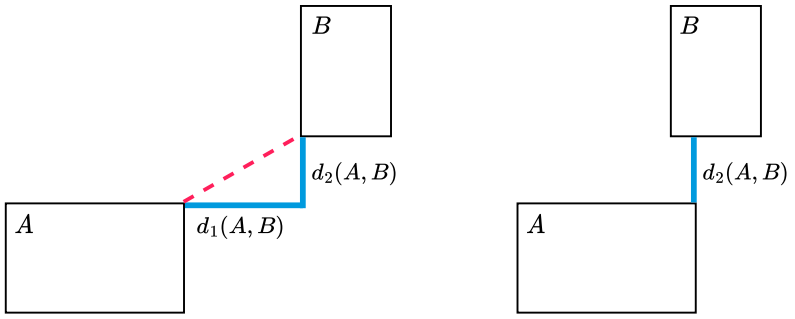


Figure 3.6: Computing the minimal distance between two boxes. *(Left)* If the element-wise distance $d(A,B)$ is greater than 0 in all dimensions, the minimal distance is simply given by $\|d(A,B)\|$. *(Right)* In this case, we have $d_1(A,B) < 0$ and we correspondingly need to set the first element of $d(A,B)$ to 0 before computing the norm.

We now introduce another hyperparameter, the *negative sampling distance* $\delta > 0$, and define the loss for a negative training example $C \not\sqsubseteq \exists r.D$ as

$$\begin{aligned}\mathcal{L}_{\not\sqsubseteq}(C,r,D) = \ & \left(\delta - \mu(\mathrm{Box}_{\boldsymbol{\theta}}(C) + \mathrm{Bump}_{\boldsymbol{\theta}}(D),\ \mathrm{Head}_{\boldsymbol{\theta}}(r))\right)^2 \\ & + \left(\delta - \mu(\mathrm{Box}_{\boldsymbol{\theta}}(D) + \mathrm{Bump}_{\boldsymbol{\theta}}(C),\ \mathrm{Tail}_{\boldsymbol{\theta}}(r))\right)^2.\end{aligned}$$

This loss encourages the minimal distance between the bumped embedding of C and the head box of r, as well as the minimal distance between the bumped embedding of D and the tail box of r, to be close to the negative sampling distance δ. As desired, it thus makes the negative training example less likely to be induced by the model. Our formulation is based on a similar loss that can be found in the implementation of ELBE [Peng et al., 2022].

Negative sampling procedure. In contrast to existing DLE methods, which generate a single set of negative samples in the beginning of the training process, we follow the approach more common in the KGE setting and generate new negative training examples every epoch. This has the advantage that our model learns to differentiate negative samples from positive ones in general, and not just for a specific fixed set of negative training examples.

We also experiment with generating $\omega > 1$ negative samples per NF3 axiom, which further improves the performance of the model as we demonstrate in Section 4.6.3.

Algorithm 1 Training procedure of Box^{2}EL

Require:
An $\mathcal{EL}^{++}$ ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$, the embedding dimensionality n, the margin γ, the negative sampling distance δ, the number of negative samples ω, the regularisation parameter λ, the step size η, and the number of training epochs e.

procedure TRAIN($\mathcal{T}$, $\mathcal{A}$, n, γ, δ, ω, λ, η, e)
  $\mathcal{T} \leftarrow \mathcal{T} \cup$ ELIMINATEABOX($\mathcal{A}$)
  $\mathcal{T} \leftarrow$ NORMALISE($\mathcal{T}$)
  $\boldsymbol{\theta} \leftarrow \mathcal{U}(-1, 1)$ ▷ initialise all embeddings randomly
  for $i \in \{1, \ldots, e\}$ **do**
    $B \leftarrow$ SAMPLEMINIBATCH($\mathcal{T}$)
    $N \leftarrow \emptyset$
    for $j \in \{1, \ldots, \omega\}$ **do**
      $N \leftarrow N \cup$ SAMPLENEGATIVES(B)
    end for
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \frac{\eta}{|B|} \nabla_{\boldsymbol{\theta}} \mathcal{L}(B, N, \gamma, \delta, \lambda; \boldsymbol{\theta})$
  end for
end procedure

3.4.4 Regularisation

The final ingredient in our loss formulation is a regularisation term for the bump vectors. Intuitively, the bump vectors make our relation model very expressive — as can be seen by the fact that they are a key ingredient in the proof of the full expressiveness of BoxE [Abboud et al., 2020] — and we thus want to limit their power in order to prevent overfitting. We therefore add the following regularisation loss:

$$\mathcal{L}_r(\boldsymbol{\theta}) = \lambda \sum_{C \in \mathbf{CI}} \left\| \text{Bump}_{\boldsymbol{\theta}}(C) \right\|,$$

where λ is a regularisation hyperparameter.

3.4.5 Training algorithm

To learn embeddings for a given ontology, we first eliminate the ABox and normalise the TBox, as previously discussed. We then start with a random initialisation of the embeddings and minimise the sum of the loss terms of all normal forms via mini-batch gradient descent. Similar to previous work [Kulmanov et al., 2019; Xiong et al., 2022], we formally specify the training algorithm in Algorithm 1.

3.5 Soundness

The loss function that is minimised during the training of Box^{2}EL intuitively encodes the axioms of an ontology in the embedding space by requiring certain geometric relationships to hold between the representations of concepts and roles. We now show that the geometric interpretation we learn indeed corresponds to a logical model of the given ontology. Our proof is inspired by similar work for other embedding models [Kulmanov et al., 2019; Xiong et al., 2022].

Theorem 3.5.1 (Soundness). *Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an $\mathcal{EL}^{++}$ ontology. If there exists a Box^{2}EL model with parameters $\boldsymbol{\theta}$ and a $\gamma \leq 0$ such that $\mathcal{L}(\mathcal{O}; \boldsymbol{\theta}) = 0$, then $\mathcal{O}$ has a model.*

Proof. We first perform the standard steps of eliminating the ABox and normalising the axioms in $\mathcal{O}$. Let $\mathcal{O}'$ denote the resulting ontology.

Consider the following geometric interpretation $\mathcal{I}_{\boldsymbol{\theta}} = (\Delta^{\mathcal{I}_{\boldsymbol{\theta}}}, \cdot^{\mathcal{I}_{\boldsymbol{\theta}}})$, induced by the trained Box^{2}EL model:

1. $\Delta^{\mathcal{I}_{\boldsymbol{\theta}}} = \mathbb{R}^n$,
2. for every concept name $C \in \mathbf{CI}$, let $C^{\mathcal{I}_{\boldsymbol{\theta}}} = \mathrm{Box}_{\boldsymbol{\theta}}(C)$,
3. for every role $r \in \mathbf{R}$, let $r^{\mathcal{I}_{\boldsymbol{\theta}}} = \mathrm{Head}_{\boldsymbol{\theta}}(r) \times \mathrm{Tail}_{\boldsymbol{\theta}}(r)$.

We show that $\mathcal{I}_{\boldsymbol{\theta}}$ is a model of $\mathcal{O}'$. First, note that $\mathcal{L}(\mathcal{O}; \boldsymbol{\theta}) = 0$ implies that $\mathcal{L}_r(\boldsymbol{\theta}) = 0$, and thus $\mathrm{Bump}_{\boldsymbol{\theta}}(C) = \mathbf{0}$ for any $C \in \mathbf{CI}$. We now show that $\mathcal{I}_{\boldsymbol{\theta}}$ satisfies every axiom $\alpha \in \mathcal{O}'$, distinguishing between the different normal forms. Implicitly, we make frequent use of Proposition 3.4.1, which we do not state explicitly for the sake of brevity.

Case 1: $\alpha = C \sqsubseteq D$. Since $\mathcal{L}_1(C, D; \boldsymbol{\theta}) = \mathcal{L}_{\subseteq}(\mathrm{Box}_{\boldsymbol{\theta}}(C), \mathrm{Box}_{\boldsymbol{\theta}}(D)) = 0$, we have that $\mathrm{Box}_{\boldsymbol{\theta}}(C) \subseteq \mathrm{Box}_{\boldsymbol{\theta}}(D)$. But then it immediately follows from the definition of $\mathcal{I}_{\boldsymbol{\theta}}$ that $C^{\mathcal{I}_{\boldsymbol{\theta}}} \subseteq D^{\mathcal{I}_{\boldsymbol{\theta}}}$.

Case 2: $\alpha = C \sqcap D \sqsubseteq E$. We have that $\mathcal{L}_2(C, D, E; \boldsymbol{\theta}) = 0$ and therefore it follows that $\mathrm{Box}_{\boldsymbol{\theta}}(C) \cap \mathrm{Box}_{\boldsymbol{\theta}}(D) \subseteq \mathrm{Box}_{\boldsymbol{\theta}}(E)$. Hence, we have $(C \sqcap D)^{\mathcal{I}_{\boldsymbol{\theta}}} = C^{\mathcal{I}_{\boldsymbol{\theta}}} \cap D^{\mathcal{I}_{\boldsymbol{\theta}}} = \mathrm{Box}_{\boldsymbol{\theta}}(C) \cap \mathrm{Box}_{\boldsymbol{\theta}}(D) \subseteq \mathrm{Box}_{\boldsymbol{\theta}}(E) = E^{\mathcal{I}_{\boldsymbol{\theta}}}$.

Case 3: $\alpha = C \sqsubseteq \exists r.D$. Let $x \in C^{\mathcal{I}_\theta} = \text{Box}_{\boldsymbol{\theta}}(C)$. Since $\mathcal{L}_3(C, r, D; \boldsymbol{\theta}) = 0$ and all bump vectors are $\mathbf{0}$, we have $\text{Box}_{\boldsymbol{\theta}}(C) \subseteq \text{Head}_{\boldsymbol{\theta}}(r)$ and therefore $x \in \text{Head}_{\boldsymbol{\theta}}(r)$. Similarly, for any $y \in D^{\mathcal{I}_\theta}$ we have $y \in \text{Tail}_{\boldsymbol{\theta}}(r)$. But then $(x, y) \in r^{\mathcal{I}_\theta}$ and therefore $x \in (\exists r.D)^{\mathcal{I}_\theta}$.

Case 4: $\alpha = \exists r.C \sqsubseteq D$. Let $x \in (\exists r.C)^{\mathcal{I}_\theta}$. Hence, there exist a $y \in C^{\mathcal{I}_\theta}$ such that $(x, y) \in r^{\mathcal{I}_\theta}$. By the definition of $r^{\mathcal{I}_\theta}$, we must therefore have $x \in \text{Head}_{\boldsymbol{\theta}}(r)$. Since $\mathcal{L}_4(r, C, D; \boldsymbol{\theta}) = 0$, furthermore $\text{Head}_{\boldsymbol{\theta}}(r) \subseteq \text{Box}_{\boldsymbol{\theta}}(D)$ and therefore $x \in D^{\mathcal{I}_\theta}$.

Case 5: $\alpha = C \sqcap D \sqsubseteq \bot$. We have $\mathcal{L}_d(\text{Box}_{\boldsymbol{\theta}}(C), \text{Box}_{\boldsymbol{\theta}}(D)) = 0$, so by Proposition 3.4.2 we have that $(C \cap D)^{\mathcal{I}_\theta} = \text{Box}_{\boldsymbol{\theta}}(C) \cap \text{Box}_{\boldsymbol{\theta}}(D) = \emptyset \subseteq \bot^{\mathcal{I}_\theta}$.

Case 6: $\alpha = \exists r.C \sqsubseteq \bot$. The loss $\mathcal{L}_6(r, C; \boldsymbol{\theta}) = 0$ implies that $\text{Head}_{\boldsymbol{\theta}}(r) = \emptyset$. Therefore $r^{\mathcal{I}_\theta} = \emptyset$, which means $(\exists r.C)^{\mathcal{I}_\theta} = \emptyset$ and hence $(\exists r.C)^{\mathcal{I}_\theta} \subseteq \bot^{\mathcal{I}_\theta}$.

Case 7: $\alpha = C \sqsubseteq \bot$. We have that $\mathcal{L}_7(C) = 0$, from which we immediately obtain $\text{Box}_{\boldsymbol{\theta}}(C) = \emptyset$. Thus, $C^{\mathcal{I}_\theta} \subseteq \bot^{\mathcal{I}_\theta}$.

We have shown that $\mathcal{I}_{\boldsymbol{\theta}}$ satisfies every axiom in $\mathcal{O}'$, and is therefore a model of $\mathcal{O}'$. But since $\mathcal{O}'$ is a conservative extension of $\mathcal{O}$ [Baader et al., 2005], it follows that $\mathcal{I}_{\boldsymbol{\theta}}$ is also a model of $\mathcal{O}$. □

4
Empirical Evaluation

In this chapter, we perform an extensive empirical evaluation of Box^{2}EL in a variety of different settings, and demonstrate that the theoretical advantages of our model manifest themselves in practice. Furthermore, we present a novel benchmark for predicting subsumptions between named and complex concepts, and use it to evaluate the inductive reasoning capabilities of Box^{2}EL and of a variety of standard models.

We begin by first giving a brief overview of our implementation of Box^{2}EL. Subsequently, we introduce a simple proof of concept ontology to demonstrate the workings of Box^{2}EL and its conceptual advantages over competing models. We proceed with our empirical evaluation in the three different settings of subsumption prediction, link prediction, and deductive reasoning. Finally, we present a number of ablation studies that highlight the contribution of different parts of our model.

4.1 Implementation

We implemented Box^{2}EL in the PyTorch machine learning framework [Paszke et al., 2019], using the publicly available code of ELBE [Peng et al., 2022] as a starting point for our implementation. Our code is roughly organised into the following subsystems:

- The implementation of the model itself, which largely follows the mathematical description given in Chapter 3. In particular, the model implementation specifies the loss functions for the different normal forms.
- The training subsystem, which implements the training algorithm given in Algorithm 1.
- The evaluation subsystem, which takes a trained Box^{2}EL model and evaluates it on a validation or testing dataset, computing standard ranking-based metrics.
- Finally, a variety of data loaders handle parsing ontologies given in the Web Ontology Language (OWL) format [Grau et al., 2008] and transforming them to an internal tensor-based representation.

The normalisation procedure that transforms an $\mathcal{EL}^{++}$ ontology into a set of axioms in one of the normal forms is handled as a pre-processing step. We use the implementation provided by Kulmanov et al. [2019], which internally makes use of the jcel reasoner [Mendez, 2012].

4.2 Proof of concept: family ontology

In order to validate our implementation and demonstrate the expressiveness of our novel role representation, we evaluate Box^{2}EL on the following proof of concept ontology from the family domain, which is adapted from [Kulmanov et al., 2019]:

$$\mathsf{Father} \sqsubseteq \mathsf{Male} \qquad \mathsf{Mother} \sqsubseteq \mathsf{Female}$$

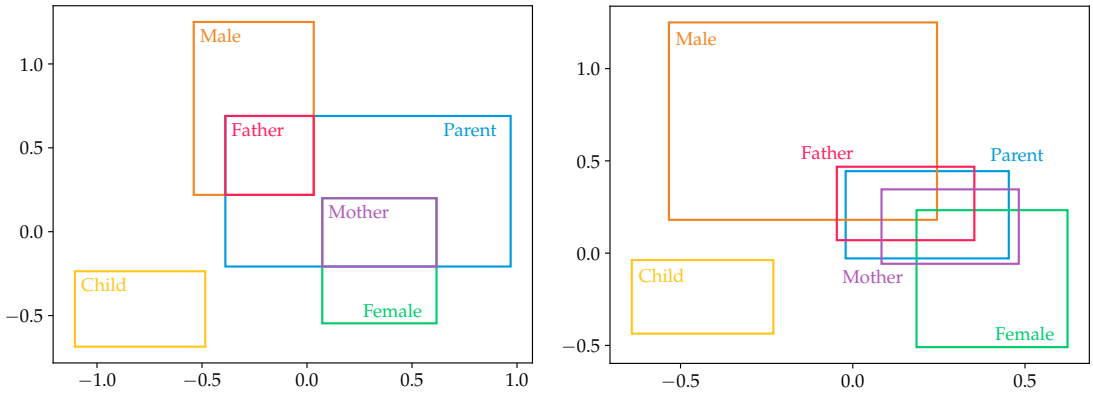
$$\mathsf{Father} \sqsubseteq \mathsf{Parent} \qquad \mathsf{Mother} \sqsubseteq \mathsf{Parent}$$

$$\mathsf{Male} \sqcap \mathsf{Parent} \sqsubseteq \mathsf{Father} \qquad \mathsf{Female} \sqcap \mathsf{Parent} \sqsubseteq \mathsf{Mother}$$

$$\mathsf{Male} \sqcap \mathsf{Female} \sqsubseteq \bot \qquad \mathsf{Parent} \sqcap \mathsf{Child} \sqsubseteq \bot$$

$$\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Mother} \qquad \mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Father}$$

$$\mathsf{Parent} \sqsubseteq \exists \mathsf{hasChild}.\mathsf{Child}$$



(a) Box^{2}EL **(b)** ELBE

Figure 4.1: Visualisation of the embeddings learnt by Box^{2}EL (left) and ELBE (right) for the proof of concept ontology. While Box^{2}EL can accurately represent the axioms in the ontology, the limitations of TransE as a model for roles prevent ELBE from learning correct embeddings.

This ontology contains a variety of interesting features: first, the axioms $\mathsf{Male} \sqcap \mathsf{Parent} \sqsubseteq \mathsf{Father}$ and $\mathsf{Female} \sqcap \mathsf{Parent} \sqsubseteq \mathsf{Mother}$ require a concept representation that can accurately model intersections in the embedding space. Second, the role hasParent forms a *one-to-many* relationship between the concepts Child and Mother/Father.

In order to be able to visualise the learnt embeddings, we train Box^{2}EL with an embedding dimensionality of $n = 2$. We set the margin $\gamma = 0$, apply a regularisation of $\lambda = 1$, and do not employ any negative sampling. We select similar hyperparameters for training ELBE [Peng et al., 2022], a comparable state-of-the-art $\mathcal{EL}^{++}$ embedding model that also interprets concepts as boxes and uses TransE to represent roles. Furthermore, we add the following *visualisation loss* term to the objective function to ensure the learnt boxes have a big enough volume for plotting:

$$\mathcal{L}_V = \frac{1}{n|\mathbf{CI}|} \sum_{C \in \mathbf{CI}} \sum_{1 \leq i \leq n} \max\{0,\, 0.2 - o(\mathrm{Box}_{\boldsymbol{\theta}}(C))_i\}.$$

The resulting embeddings of both models are depicted in Figure 4.1.

We can clearly see that Box^{2}EL is able to successfully learn embeddings that align with the axioms in the ontology. In particular, the embeddings fulfill all disjointness axioms and correctly represent the relationship between the concepts Father, Male, Mother, Female, and Parent.

In contrast, we find that the embeddings learnt by ELBE violate several of the axioms in the ontology. This is due to the inability of the underlying TransE model to correctly represent *one-to-many* relationships: because the ontology contains the axioms $\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Mother}$ as well as $\mathsf{Child} \sqsubseteq \exists \mathsf{hasParent}.\mathsf{Father}$, the model is forced to let the embeddings of Mother and Father overlap.

4.3 Subsumption prediction

Having demonstrated the effectiveness of our method on a proof of concept example, we next consider a variety of large-scale real-world datasets. We first focus on the task of *subsumption prediction*, i.e. predicting subsumptions that are not necessarily entailed by the given ontology.

4.3.1 Datasets

We evaluate Box^{2}EL on the following biomedical ontologies:

- **GALEN** [Rector et al., 1996], a clinical ontology that comprises a wide collection of medical terminology, ranging from specific diseases and treatments to general structures and processes.
- **Gene Ontology (GO)** [Ashburner et al., 2000], which represents genes and their associated functions in a unified way across species.
- **Anatomy** [Mungall et al., 2012] (also called *Uberon*), a multi-species ontology that captures a wide variety of anatomical structures and the relationships that hold between them.

The size of these ontologies in terms of the number of classes, roles, and axioms is summarised in Table 4.1. Note that none of the datasets we consider contain axioms in the sixth or seventh normal form.

Ontology	Classes	Roles	NF1	NF2	NF3	NF4	NF5
GALEN	23,142	397	27,874	13,595	28,118	13,597	0
GO	45,895	9	85,471	12,131	20,324	12,129	30
Anatomy	106,363	157	122,022	2,121	152,289	2,143	184

Table 4.1: Sizes of the different ontologies we consider. The number of classes, roles, and axioms in each normal form is reported.

4.3.2 Subsumptions between named and complex concepts

The ontologies we have introduced have previously been used to evaluate $\mathcal{EL}^{++}$ embedding methods on the task of subsumption prediction [Mondal et al., 2021; Xiong et al., 2022]. In these previous works, the axioms in the first normal form are first split into a training, validation, and testing set in a proportion of 70%/20%/10%, respectively.

However, when inspecting the data contained in the relevant splits provided with the implementation of these previous methods, we find that the validation set is in fact contained entirely within the training set, i.e. the data is actually split in a 90%/10% fashion. This lack of an independent validation makes it very difficult to perform a proper hyperparameter search without overfitting on the training data.

Moreover, the existing benchmark only takes axioms in the first normal form into account, i.e. subsumptions between named concepts of the form $C \sqsubseteq D$. In order to enable a more thorough analysis of the capabilities of DLE models, we propose to also evaluate their ability to predict subsumptions between named and *complex* concepts. To this end, and to address the issue with the validation set identified earlier, we develop a new prediction benchmark for the three datasets described above.

Our novel benchmark consists of training (80%), validation (10%), and testing (10%) sets for axioms in *all* the normal forms NF1–NF4. This enables the evaluation of DLE models regarding subsumption prediction between named concepts (NF1), named concepts and conjunctions (NF2), and named concepts and existentially restricted concepts (NF3 and NF4). We furthermore ensure that all classes and roles that occur in the validation and testing sets also occur in the training set, and verify that the validation set is not contained in the training set. We provide the exact data splits

we used together with our implementation in order to make the benchmark available to the wider research community.

4.3.3 Baselines

We compare our proposed method Box^{2}EL with two representative state-of-the-art $\mathcal{EL}^{++}$ embedding methods: ELEm [Kulmanov et al., 2019] and ELBE [Peng et al., 2022]. Both of these methods use TransE as the underlying model for roles and employ a distance-based approach for their loss functions. The main difference between them is that ELEm represents concepts as n-balls, whereas ELBE uses a box representation equivalent to ours.

We also attempted to evaluate BoxEL [Xiong et al., 2022] on our benchmark, but were unfortunately unable to reproduce results similar to what is reported in the paper, even after contacting the authors. For this reason we omit BoxEL from our comparison.

Furthermore, we also do not consider any traditional KGE methods in our experiments, since they have been shown to be considerably outperformed by DLEs [Mondal et al., 2021; Xiong et al., 2022] and are not applicable in the setting of complex concepts.

4.3.4 Evaluation protocol

As in previous work [Mondal et al., 2021; Xiong et al., 2022], we evaluate the subsumption prediction performance of the embedding models with a variety of ranking-based metrics on the testing set. This is similar to the evaluation of KGE models, as discussed in Section 2.1.3. In order to compute these metrics, we first need to define *scoring functions* for the embedding models we consider.

Scoring functions

As in the KGE setting, a scoring function $s(\cdot)$ assigns scores to candidate predictions such that the scores for axioms that are likely to be true based on the learnt embeddings are higher than those that are considered false. We need to define scoring functions for candidate predictions in all four normal forms.

First and second normal form. For an axiom $C \sqsubseteq D$ in NF1, we define the score based on the distance between the embeddings of C and D, i.e. for Box^{2}EL we have

$$s(C \sqsubseteq D) = -\|c(\text{Box}_{\boldsymbol{\theta}}(C) - c(\text{Box}_{\boldsymbol{\theta}}(D)))\|.$$

The same formulation can be used for the baseline methods. Similarly, for NF2 axioms $C \sqcap D \sqsubseteq E$, we define the score as the negative distance of the embedding of E to the intersection of C and D in the embedding space.

Note how these scoring functions closely follow the loss functions for the first two normal forms we have defined in Section 3.4.2. If the embedding model successfully captures the semantics of the ontology it was trained on, we expect the loss for axioms in the testing set to be low, or equivalently the score to be high.

Third normal form. For axioms in the third and fourth normal form the scoring function differs between Box^{2}EL and the baseline methods, because of the different role representation. For Box^{2}EL, we define the score for a subsumption $C \sqsubseteq \exists r.D$ as

$$\begin{aligned} s(C \sqsubseteq \exists r.D) = &- \left\|c(\text{Box}_{\boldsymbol{\theta}}(C) + \text{Bump}_{\boldsymbol{\theta}}(D)) - c(\text{Head}_{\boldsymbol{\theta}}(r))\right\| \\ &- \left\|c(\text{Box}_{\boldsymbol{\theta}}(D) + \text{Bump}_{\boldsymbol{\theta}}(C)) - c(\text{Tail}_{\boldsymbol{\theta}}(r))\right\|, \end{aligned}$$

again closely following the corresponding loss function.

In the baseline methods, the score is computed similarly to TransE:

$$s(C \sqsubseteq \exists r.D) = -\|c(\text{Box}_{\boldsymbol{\theta}}(C)) + \boldsymbol{r} - c(\text{Box}_{\boldsymbol{\theta}}(D))\|.$$

Fourth normal form. Finally, for an axiom $\exists r.C \sqsubseteq D$ in NF4, the score assigned by Box^{2}EL is given by

$$s(\exists r.C \sqsubseteq D) = -\left\|c(\text{Head}_{\boldsymbol{\theta}}(r) - \text{Bump}_{\boldsymbol{\theta}}(C)) - c(\text{Box}_{\boldsymbol{\theta}}(D))\right\|,$$

and for the baseline methods we define

$$s(\exists r.C \sqsubseteq D) = -\|c(\text{Box}_{\boldsymbol{\theta}}(C)) - \boldsymbol{r} - c(\text{Box}_{\boldsymbol{\theta}}(D))\|.$$

Evaluation metrics

The scoring functions defined above can be used to compute several ranking-based metrics, analogously to how evaluation is performed for KGE models. To illustrate, consider a testing axiom $C \sqsubseteq D$ in the first normal form. We fix the concept C and rank the set of corrupted axioms

$$\mathcal{C}_{C \sqsubseteq D} = \{ C \sqsubseteq D' \mid D' \in \mathbf{CI} \}$$

using the scoring function. Finally, we record the rank of the true subsumption $C \sqsubseteq D$. If our model performs well, it should assign a high score to the true subsumption and the corresponding rank should therefore be low.

We similarly compute ranks for testing axioms in the other normal forms. Finally, we report the following standard metrics introduced in Section 2.1.3: hits at k, where $k \in \{1, 10, 100\}$, the median rank (Med), the mean reciprocal rank (MRR), the mean rank (MR), and the area under the ROC curve (AUC). We compute these metrics for all axioms in each normal form individually, as well as in a combined setting in which the ranks of all normal forms are taken together.

Filtering. Recall from our discussion in Section 2.1.3 that ranks can be computed in either a *raw* or a *filtered* fashion, where we filter out true axioms from the set of corrupted axioms. For subsumption prediction, we follow previous work [Mondal et al., 2021; Xiong et al., 2022] and only report the raw ranking-based metrics instead of the filtered versions, which are challenging to compute for the sizes of the datasets we consider.

However, since the filtered metrics are generally considered to be more reliable [Bordes et al., 2013], we implement a simple and efficient approximation of filtering for NF1 and NF2 axioms as follows:

- For NF1 axioms $C \sqsubseteq D$, filter out $C \sqsubseteq C$ from $\mathcal{C}_{C \sqsubseteq D}$.
- For NF2 axioms $C \sqcap D \sqsubseteq E$, filter out $C \sqcap D \sqsubseteq C$ and $C \sqcap D \sqsubseteq D$ from $\mathcal{C}_{C \sqcap D \sqsubseteq E}$.

4.3.5 Experimental protocol

We train the embedding models on the training set of the considered ontology, consisting of 80% of the axioms in NF1–NF4. Optimisation is performed with the Adam optimiser [Kingma and Ba, 2015] and a learning rate in $\{1e^{-2}, 5e^{-3}, 1e^{-3}, 5e^{-4}\}$. We choose the learning rate and all other hyperparameters based on validation set performance. The values we consider for the other hyperparameters are $n \in \{50, 100, 200\}$, $\gamma \in \{0, 0.05, 0.1\}$, $\delta \in \{1, 2, 3\}$, $\omega \in \{1, 2, 3\}$, and $\lambda \in \{0, 0.05, 0.1\}$.

Training is performed for a maximum of $e = 10,000$ epochs. We evaluate the models on a fraction of the validation set every 100 epochs and choose the embeddings that achieve the best performance for final evaluation on the testing set. The results we report are averages across 5 runs with different random seeds, which we provide in our implementation to ensure our results are reproducible. All experiments were conducted on a machine with an Intel Xeon Bronze 3204 processor with 12 cores at a clock speed of 1.90 GHz, 128 GB of RAM, and an NVIDIA Quadro RTX 8000 GPU.

4.3.6 Results

We report the results of the embedding methods on the GALEN, GO, and Anatomy ontologies in Tables 4.2 to 4.4.

General findings

We first observe that all methods we consider perform reasonably well across the different normal forms. This is to our knowledge the first result to demonstrate that the learnt embeddings are not only useful for comparing named concepts, but are also expressive enough to perform predictive reasoning with complex concepts. As expected, performance is generally better for NF1, which involves only named concepts, but interestingly this is not always the case: for example in Anatomy we find that all models achieve stronger results for NF3 axioms than for NF1 axioms.

Comparing the different methods, we find ELEm and ELBE to perform similarly well, although ELEm is better in general, especially when it comes to complex concepts

Table 4.2: Subsumption prediction results on GALEN. NF*k* refers to the ranking metrics computed only on the ranks achieved on axioms in normal form *k*. The 'Combined' row lists the metrics computed on all ranks across normal forms.

Normal form	Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
NF1	ELEm	0.01	0.16	0.40	430	0.06	3568	0.85
	ELBE	**0.03**	0.24	0.47	138	**0.10**	2444	0.89
	Box^{2}EL	0.02	**0.25**	**0.55**	**62**	0.09	**2039**	**0.91**
NF2	ELEm	0.01	0.07	0.17	5106	0.03	7432	0.68
	ELBE	0.03	0.06	0.11	6476	0.04	8068	0.65
	Box^{2}EL	**0.05**	**0.13**	**0.22**	**3468**	**0.08**	**7246**	**0.69**
NF3	ELEm	0.02	0.14	0.28	1479	0.05	4831	0.79
	ELBE	0.03	0.14	0.25	2154	0.07	5072	0.78
	Box^{2}EL	**0.08**	**0.19**	**0.31**	**1060**	**0.12**	**4530**	**0.80**
NF4	ELEm	0.00	0.05	0.18	3855	**0.02**	**6793**	**0.71**
	ELBE	0.00	0.03	0.07	7563	0.01	8884	0.62
	Box^{2}EL	0.00	**0.08**	**0.19**	**3426**	**0.02**	6806	**0.71**
Combined	ELEm	0.01	0.12	0.29	1662	0.05	5153	0.78
	ELBE	0.02	0.14	0.27	1865	0.06	5303	0.77
	Box^{2}EL	**0.04**	**0.18**	**0.36**	**643**	**0.09**	**4511**	**0.81**

Table 4.3: Subsumption prediction results on GO. NF*k* refers to the ranking metrics computed only on the ranks achieved on axioms in normal form *k*. The 'Combined' row lists the metrics computed on all ranks across normal forms.

Normal form	Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
NF1	ELEm	0.01	0.13	0.35	590	0.05	6433	0.86
	ELBE	0.01	0.10	0.24	1156	0.04	5657	0.88
	Box^{2}EL	**0.03**	**0.16**	**0.59**	**61**	**0.08**	**2616**	**0.94**
NF2	ELEm	0.12	0.49	0.63	11	0.24	4508	0.90
	ELBE	0.01	0.05	0.09	6456	0.02	9421	0.80
	Box^{2}EL	**0.22**	**0.65**	**0.77**	**5**	**0.36**	**1546**	**0.97**
NF3	ELEm	**0.06**	**0.40**	**0.52**	**54**	**0.15**	6292	0.86
	ELBE	0.02	0.15	0.30	959	0.07	7131	0.84
	Box^{2}EL	0.00	0.14	0.51	90	0.04	**5074**	**0.89**
NF4	ELEm	**0.01**	**0.49**	0.60	**12**	**0.12**	6272	0.86
	ELBE	0.00	0.07	0.12	9049	0.02	12868	0.72
	Box^{2}EL	0.00	0.45	**0.66**	14	0.10	**4960**	**0.89**
Combined	ELEm	0.03	**0.24**	0.43	272	0.09	6204	0.86
	ELBE	0.01	0.10	0.22	1838	0.04	6986	0.85
	Box^{2}EL	**0.04**	0.23	**0.60**	**50**	**0.10**	**3151**	**0.93**

Table 4.4: Subsumption prediction results on Anatomy. NF*k* refers to the ranking metrics computed only on the ranks achieved on axioms in normal form *k*. The 'Combined' row lists the metrics computed on all ranks across normal forms.

Normal form	Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
NF1	ELEm	**0.07**	**0.30**	0.57	43	**0.14**	9059	0.91
	ELBE	0.05	0.24	0.55	68	0.11	5177	0.95
	Box^{2}EL	0.04	0.25	**0.62**	**39**	0.11	**4367**	**0.96**
NF2	ELEm	0.03	0.18	0.42	394	0.08	11592	0.89
	ELBE	0.02	0.11	0.26	1394	0.05	4885	0.96
	Box^{2}EL	**0.13**	**0.34**	**0.55**	**66**	**0.20**	**2465**	**0.98**
NF3	ELEm	0.12	0.47	0.69	13	0.23	4686	0.96
	ELBE	0.04	0.44	0.70	16	0.18	5408	0.95
	Box^{2}EL	**0.30**	**0.62**	**0.75**	**4**	**0.41**	**2612**	**0.98**
NF4	ELEm	0.00	0.03	0.23	813	0.01	10230	0.91
	ELBE	0.00	0.02	0.06	6261	0.01	15187	0.86
	Box^{2}EL	0.00	**0.07**	**0.25**	**615**	**0.02**	**6166**	**0.94**
Combined	ELEm	0.10	0.40	0.64	22	0.19	6464	0.94
	ELBE	0.04	0.36	0.63	29	0.15	5400	0.95
	Box^{2}EL	**0.19**	**0.48**	**0.69**	**13**	**0.29**	**3312**	**0.97**

and especially on GO. Box^{2}EL consistently outperforms the baseline methods on all datasets, almost always achieving the best results in the combined setting, and in most cases when considering the different normal forms in isolation. The performance gains are usually significant: for example, we find that the median rank of Box^{2}EL is more than 60% lower than the second best-performing method on GALEN, more than 80% lower on GO, and more than 40% lower on Anatomy.

When it comes to absolute performance scores, we see that the results of Box^{2}EL are promising, especially on Anatomy and on some normal forms for GO. For instance, on Anatomy we manage to achieve a median rank of 13 in the combined setting, and a hits at 1 ratio of 19%. With these strong results, we believe our model can be used in practice to investigate potentially missing axioms in real-world ontologies.

Detailed discussion

Box^{2}EL. Clearly, the novel role representation introduced in Box^{2}EL greatly improves the quality of the learnt embeddings. This becomes especially evident when comparing

the performance of Box^{2}EL to that of the similar model ELBE, which mainly differs in the fact that it uses TransE to model roles. We see that our new approach for representing roles not only generally improves prediction performance for NF3 and NF4 axioms, but also for the first two normal forms. This can be explained by the fact that the different normal forms are used to optimise the *same* embeddings; i.e. if Box^{2}EL can better represent an axiom of the form $C \sqsubseteq \exists r.D$, it will learn better embeddings for C and D, therefore also improving prediction quality for axioms in NF1 or NF2.

Notably, while Box^{2}EL consistently outperforms the baseline methods for NF3 and NF4 axioms on GALEN and Anatomy, we find that ELEm performs better on most metrics on GO, despite relying on the weaker TransE role representation. While we can only hypothesise why this might be the case, we conjecture that it is related to the fact that GO is the ontology with the fewest number of roles (see Table 4.1). We also note that GO has the fewest number of *one-to-many* and *many-to-one* relationships among all the datasets. The impact of the role representation is further discussed in Section 4.6.1.

Baseline methods. As discussed above, we find ELEm to outperform ELBE in most of our experiments in this particular subsumption prediction setting. Interestingly, this is also the case for axioms in NF2, despite the fact that ELBE uses a box representation for concepts, which is advantageous for modelling conjunction in the embedding space, as discussed in Section 3.2. However, since our own model Box^{2}EL also represents concepts as boxes and performs the strongest overall, it is unlikely that the comparatively poor performance of ELBE is caused by its concept representation.

Furthermore, we note that we generally achieve much better results with the baseline methods than have previously been reported [Mondal et al., 2021; Xiong et al., 2022]. This is despite the fact that our benchmark requires reasoning with complex concepts, and is thus more challenging.

A possible explanation is that our choice of hyperparameters differs from the literature: we generally use a much higher embedding dimension of $n = 200$, which has previously only be used by Mondal et al. [2021] on the Anatomy dataset. When

reducing the dimensionality of the embeddings, our results significantly worsen and are closer to what has been reported before. However, all previous evaluations claim that they included the dimensionality of 200 in their hyperparameter search, and we do not know exactly why they did not manage to produce similar results as ours.

We do note that, as far as we are aware, none of the previous studies evaluated the performance of the models on the validation set during the training stage. In contrast, recall from Section 4.3.5 that we evaluate the embeddings on the validation set every 100 epochs and choose the best performing model overall, which we find has a significant positive effect on our results.

Possibly, previous studies did not evaluate models during training because they found the frequent computation of ranks to be a major performance bottleneck. We circumvent this problem in two ways: first, we only perform evaluation on the first $1,000$ validation examples, instead of on the whole validation set. Second, we implement an efficient form of batched ranking that can directly be executed in parallel on the GPU, in comparison to the loop-based CPU ranking procedures we find in previous implementations. Overall, this allows us to compute ranks in a matter of a few seconds, compared to several minutes with the previous approaches.

Metrics. Lastly, we want to point out that throughout our experiments the median ranks often are several order of magnitudes smaller than the mean ranks. This suggests that we frequently encounter outliers on which our methods partially fail and yield high ranks. For this reason, we consider median rank and MRR, which are explicitly designed to counteract the influence of outliers, to be the more reliable evaluation metrics.

4.4 Link prediction

We next evaluate our model on the task of *link prediction*, i.e. predicting role assertions of the form $r(a, b)$. Recall that the first step of our training algorithm is to eliminate the ABox from a given ontology, and predicting links is therefore equivalent to predicting subsumptions of the form $\{a\} \sqsubseteq \exists r.\{b\}$.

While this means we can directly apply ideas from subsumption prediction to the link prediction task, the focus between the two settings is very different. In subsumption prediction, our goal is to predict new *axioms*, i.e. logical background knowledge about the domain of interest. In contrast, in link prediction, we want to predict new relational *facts* about real-world individuals.

4.4.1 Datasets

We consider the protein-protein interaction (PPI) prediction task introduced by Kulmanov et al. [2019]. They provide two ontologies of PPIs in human and yeast organisms, constructed by combining the STRING database of PPIs [Szklarczyk et al., 2021] with the Gene Ontology (GO) [Ashburner et al., 2000]. The proteins and their interactions recorded in STRING form the ABox of the constructed ontologies, while GO acts as the TBox, and is enriched with additional information about the association of proteins with functions. The task is to predict subsumptions of the form $\{P_1\} \sqsubseteq \exists \mathsf{interacts}.\{P_2\}$ between proteins P_1 and P_2.

4.4.2 Baselines

We compare Box^{2}EL with the state-of-the-art $\mathcal{EL}^{++}$ embedding methods ELEm [Kulmanov et al., 2019], ELBE [Peng et al., 2022], EmEL^{++} [Mondal et al., 2021], and BoxEL [Xiong et al., 2022]. ELEm and ELBE were already introduced earlier (see Section 4.3.3). EmEL^{++} is similar to ELEm, but considers additional role axioms that are part of $\mathcal{EL}^{++}$. BoxEL represents concepts as boxes and roles as affine transformations, which is comparable to the TransE model used by ELBE. We do not re-evaluate the models, but instead report the relevant best results from the literature.

4.4.3 Evaluation and experimental protocol

In order to evaluate our method, we use the 80%/10%/10% training, testing, and validation split of the PPI data provided by Kulmanov et al. [2019]. We report the

Table 4.5: PPI prediction results on the yeast and human datasets. Columns annotated with (F) contain filtered metrics, other columns contain raw metrics. All baseline results except for BoxEL are from [Peng et al., 2022]. The results for BoxEL are from the original paper [Xiong et al., 2022].

Dataset	Model	H@10	H@10 (F)	H@100	H@100 (F)	MR	MR (F)	AUC	AUC (F)
Yeast	ELEm	0.10	0.23	0.50	0.75	247	187	0.96	0.97
	EmEL^{++}	0.08	0.17	0.48	0.65	336	291	0.94	0.95
	BoxEL	0.09	0.20	0.52	0.73	423	379	0.93	0.94
	ELBE	**0.11**	0.26	0.57	0.77	201	154	0.96	0.97
	Box^{2}EL	0.10	**0.30**	**0.62**	**0.84**	**180**	**130**	**0.97**	**0.98**
Human	ELEm	**0.09**	0.22	0.43	0.70	658	572	0.96	0.96
	EmEL^{++}	0.04	0.13	0.38	0.56	772	700	0.95	0.95
	BoxEL	0.07	0.10	0.42	0.63	1574	1530	0.93	0.93
	ELBE	**0.09**	0.22	0.49	0.72	434	362	0.97	**0.98**
	Box^{2}EL	0.08	**0.24**	**0.52**	**0.79**	**314**	**241**	**0.98**	**0.98**

same ranking-based metrics as in the subsumption prediction setting, computed with the same scoring function, i.e.

$$s(\{P_1\} \sqsubseteq \exists \mathsf{interacts}.\{P_2\}) = -\left\|c(\mathrm{Box}_{\boldsymbol{\theta}}(\{P_1\}) + \mathrm{Bump}_{\boldsymbol{\theta}}(\{P_2\})) - c(\mathrm{Head}_{\boldsymbol{\theta}}(\mathsf{interacts}))\right\| - \left\|c(\mathrm{Box}_{\boldsymbol{\theta}}(\{P_2\}) + \mathrm{Bump}_{\boldsymbol{\theta}}(\{P_1\})) - c(\mathrm{Tail}_{\boldsymbol{\theta}}(\mathsf{interacts}))\right\|.$$

Since the number of proteins is much smaller than the number of classes in the datasets we considered in the subsumption prediction setting, we are now able to efficiently compute all metrics in a raw and filtered (F) fashion. The experimental protocol is the same as before (see Section 4.3.5).

4.4.4 Results

Table 4.5 lists the results of Box^{2}EL and the baseline methods on the yeast and human PPI prediction datasets. We see that Box^{2}EL outperforms the current state of the art on all metrics except raw hits at 1, usually with a significantly better performance of several percentage points. The results once again demonstrate the expressiveness of our novel role representation, which is especially important in the link prediction setting, where all subsumptions we predict are in third normal form.

4.5 Deductive reasoning

Our evaluation so far has been concerned with the setting of *inductive reasoning*, or prediction. We now examine how well our model is able to approximate *deductive reasoning* in the embedding space.

4.5.1 Experimental setup

We consider the same three ontologies we have used for the subsumption prediction task (Section 4.3): GALEN, GO, and Anatomy. However, instead of splitting the datasets into separate training, validation, and testing sets, we now train our models on the entire ontology including all axioms.

For evaluation, we use the standard ELK reasoner [Kazakov et al., 2014] to create a set of inferences for each of the ontologies we consider. These inferences correspond to subsumptions between named concepts that logically follow from the given ontology. We again point out the difference to the subsumption prediction setting: instead of *predicting* axioms that are statistically and semantically likely to be missing from the ontology, we now evaluate the ability of the embedding models to *infer* subsumptions that logically follow from the axioms in the ontology.

We report the results of Box^{2}EL and the same baseline methods considered previously. The evaluation and experimental protocol is equivalent to the subsumption prediction setting (Sections 4.3.4 and 4.3.5). However, note that the inferences we evaluate the models on only include named concepts (i.e. all test subsumptions are in NF1). We split off 10% off the inference set and use it for validation.

A similar experiment evaluating the deductive reasoning capabilities of $\mathcal{EL}^{++}$ embedding models has been previously conducted by Mondal et al. [2021]. However, in contrast to our approach, the inferences they evaluate their models on are drawn from an arbitrary training set containing only 80% of the axioms from the ontology. These inferences will thus inherently be incomplete, whereas our setup includes all subsumptions that follow from the complete ontology. Furthermore, and more gravely, upon inspecting their data we find that the inferences they use for evaluation in fact are

Table 4.6: Deductive reasoning results on GALEN, GO, and Anatomy.

Dataset	Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
GALEN	ELEm	0.00	0.04	0.20	1807	0.01	4405	0.81
	ELBE	0.00	0.06	0.16	1961	0.02	4115	0.82
	Box^{2}EL	**0.01**	**0.08**	**0.24**	**1030**	**0.03**	**2825**	**0.88**
GO	ELEm	0.00	0.04	0.22	1629	0.02	7377	0.84
	ELBE	0.00	0.06	0.21	935	0.02	3846	0.92
	Box^{2}EL	0.00	**0.08**	**0.50**	**100**	**0.04**	**1569**	**0.97**
Anatomy	ELEm	0.00	0.07	0.28	901	0.02	7958	0.93
	ELBE	0.00	0.08	0.32	336	0.03	2312	0.98
	Box^{2}EL	0.00	**0.09**	**0.47**	**120**	**0.04**	**1178**	**0.99**

all also contained in their training set. Therefore, any model that manages to overfit on the training data will be able to achieve strong results on their benchmark.

4.5.2 Results

The results for the deductive reasoning task are listed in Table 4.6. We observe that all methods exhibit some capability for sub-symbolic deductive reasoning, although the results are generally worse than in the subsumption prediction setting. On GALEN, ELEm performs better than ELBE, whereas ELBE is the better baseline method on GO and Anatomy. Our own model Box^{2}EL outperforms the other methods on all metrics across the three datasets, with significant performance gains especially for hits at 100, median rank, and mean rank.

4.5.3 Comparison of the reasoning and prediction task

While the results we were able to achieve are promising, it may seem counterintuitive at first that the embedding methods generally perform worse on the reasoning than the prediction task. Whereas the latter involves predicting axioms that do not necessarily have any direct semantic relation to the training data, in the prediction setting, all testing subsumptions follow logically from the ontology that was used for training.

In order to explain why the embedding models still perform comparatively worse on the reasoning task, it is instructive to investigate some of the training and testing data in

detail. For example, in the GALEN ontology, we find the following subsumption in the inference set:

$$\mathsf{SodiumLactate} \sqsubseteq \mathsf{SodiumCompound}.$$

In order to arrive at this inference, a reasoning algorithm such as ELK has to perform the following derivations, where we abbreviate SodiumLactate as SL and ChemicalSubstance as CS in the first derivation[1]:

$$\frac{\mathsf{SL} \sqsubseteq \mathsf{NAMEDComplexChemical} \qquad \mathsf{NAMEDComplexChemical} \sqsubseteq \mathsf{CS}}{\mathsf{SodiumLactate} \sqsubseteq \mathsf{ChemicalSubstance}} \tag{4.1}$$

$$\mathsf{SodiumLactate} \sqsubseteq \exists \mathsf{isMadeOf}.\mathsf{Sodium} \tag{4.2}$$

$$\frac{(4.1) \qquad (4.2) \qquad \mathsf{ChemicalSubstance} \sqcap \exists \mathsf{isMadeOf}.\mathsf{Sodium} \sqsubseteq \mathsf{SodiumCompound}}{\mathsf{SodiumLactate} \sqsubseteq \mathsf{SodiumCompound}} \tag{4.3}$$

In order to perform the same reasoning in the embedding space, the embeddings learnt by our model have to be highly accurate for a number of different concepts and roles such as SodiumLactate, isMadeOf, Sodium, and SodiumCompound, to name but a few examples from the last derivation. Furthermore, recall that the scoring function we use measures only the distance between concepts. It is quite likely in this example that the embedding of SodiumLactate is closer to NAMEDComplexChemical than SodiumCompound, simply because of the axiom $\mathsf{SodiumLactate} \sqsubseteq \mathsf{NAMEDComplexChemical}$ in the training data, increasing the rank of the desired subsumption.

In contrast, in the prediction setting, we find that the following axiom is contained in the testing data:

$$\mathsf{SodiumLactate} \sqsubseteq \mathsf{NAMEDComplexChemical}.$$

While this axiom does not occur in the training data and cannot be logically inferred

[1] The derivations are meant to be read from top to bottom, similar to a natural deduction-style proof.

from it, the following axioms do occur in the training data:

$$\begin{aligned}
\mathsf{SodiumLactate} &\sqsubseteq \exists \mathsf{isMadeOf.Sodium} \\
\mathsf{SodiumBicarbonate} &\sqsubseteq \exists \mathsf{isMadeOf.Sodium} \\
\mathsf{SodiumCitrate} &\sqsubseteq \exists \mathsf{isMadeOf.Sodium} \\
\mathsf{SodiumBicarbonate} &\sqsubseteq \mathsf{NAMEDComplexChemical} \\
\mathsf{SodiumCitrate} &\sqsubseteq \mathsf{NAMEDComplexChemical}.
\end{aligned}$$

It seems quite likely that our model will be able to exploit this statistical information to learn an embedding for SodiumLactate that is close to NAMEDComplexChemical, yielding a low rank for the desired axiom.

In conclusion, reasoning is a harder task than prediction because it requires a number of steps involving a variety of concepts. Furthermore, the scoring functions used in current embedding models are not designed for the purpose of reasoning. In the prediction setting on the other hand, embedding models can exploit statistical information to make predictions that do not necessarily logically follow from the training data.

Nevertheless, existing embedding models still perform quite well on the deductive reasoning task, and we are confident that future research will be able to address the shortcomings we have identified above.

4.6 Ablation studies

The previous experiments show that Box^{2}EL is a strong $\mathcal{EL}^{++}$ embedding model that achieves state-of-the-art results on several benchmarks. We now conduct a variety of ablation studies to investigate the performance impact of different parts of our model. All studies are conducted on the GALEN ontology for the subsumption prediction task.

4.6.1 Impact of role representation

The central novel contribution of our method is its role representation based on BoxE. We have illustrated the conceptual advantages of this role representation and argued that it is a key ingredient for the performance of Box^{2}EL. To strengthen these claims,

Table 4.7: Impact of the role representation on the performance of Box^{2}EL. We compare our original model with a version where the BoxE-based role representation has been replaced with TransE (Box^{2}EL-TE). The results are for subsumption prediction on GALEN.

Normal form	Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
NF1	Box^{2}EL-TE	**0.02**	0.22	0.45	159	0.08	2417	0.90
	Box^{2}EL	**0.02**	**0.25**	**0.55**	**62**	**0.09**	**2039**	**0.91**
NF2	Box^{2}EL-TE	0.02	0.05	0.13	5314	0.03	7510	0.68
	Box^{2}EL	**0.05**	**0.13**	**0.22**	**3468**	**0.08**	**7246**	**0.69**
NF3	Box^{2}EL-TE	0.01	0.08	0.19	2544	0.03	5623	0.76
	Box^{2}EL	**0.08**	**0.19**	**0.31**	**1060**	**0.12**	**4530**	**0.80**
NF4	Box^{2}EL-TE	0.00	0.02	0.09	4260	0.01	7092	0.69
	Box^{2}EL	0.00	**0.08**	**0.19**	**3426**	**0.02**	**6806**	**0.71**
Combined	Box^{2}EL-TE	0.01	0.11	0.25	1557	0.05	5099	0.78
	Box^{2}EL	**0.04**	**0.18**	**0.36**	**643**	**0.09**	**4511**	**0.81**

we conduct an ablation study in which we replace our role representation with TransE, similar to ELBE [Peng et al., 2022], and keep the rest of our model exactly the same. The results are given in Table 4.7.

We observe that the model based on BoxE outperforms the TransE-based model on all metrics, in most cases by a large margin. Furthermore, we again see that the different role representation not only improves results for axioms involving roles (i.e. axioms in NF3 or NF4), but consistently across the different normal forms. As noted earlier, this is due to the fact that the axioms in different normal forms are used to optimise the same concept embeddings. Overall, we conclude that the novel role representation is indeed crucially important for the performance of our model.

4.6.2 Bump vectors and regularisation

The second ablation study we conduct concerns the details of our role representation. As in BoxE, a central feature of our representation are bump vectors, which enable the embeddings of concepts to dynamically adapt to different roles. In Table 4.8, we investigate the performance of a model that does not use bump vectors, but instead requires the embeddings of concepts to directly lie in a given head or tail box, without previously having been "bumped".

Table 4.8: Impact of bump vectors and regularisation on the performance of Box^{2}EL. We compare our original model with a version without bump vectors (Box^{2}EL-NB) and a version without regularisation (Box^{2}EL-NR). The results are for subsumption prediction on GALEN, in the combined setting.

Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
Box^{2}EL-NB	0.00	0.03	0.12	7336	0.01	8673	0.63
Box^{2}EL-NR	**0.04**	0.16	0.33	877	0.08	4789	0.79
Box^{2}EL	**0.04**	**0.18**	**0.36**	**643**	**0.09**	**4511**	**0.81**

Furthermore, recall from Section 3.4.4 that we employ a regularisation term to prevent the bump vectors from becoming too large, in order to counteract overfitting. We examine the importance of regularisation by training an unregularised model. The results are also given in Table 4.8.

First, we observe that the model without bump vectors significantly underperforms, which leads us to the conclusion that they are an important component of our model. Furthermore, we see that regularisation consistently improves our results across all metrics, suggesting that it successfully reduces overfitting.

4.6.3 Number of negative samples

Finally, we investigate the impact of the number of negative samples on the performance of our model. In Table 4.9, we report the results achieved by various Box^{2}EL models trained with $\omega \in \{0, 1, 2, 3\}$ negative samples per axiom in NF3.

Comparing the results of the model that was trained without any negative sampling to the others, we see that it performs significantly worse. This demonstrates that negative sampling is essential to learn strong embeddings. Increasing the number ω of negative samples improves the results on some metrics, but not on all. We find that a number of 2–3 negative samples generally performs best.

Table 4.9: Impact of the number of negative samples on the performance of Box^{2}EL. The model Box^{2}EL-*k* denotes Box^{2}EL trained with *k* negative samples per axiom in NF3. The results are for subsumption prediction on GALEN, in the combined setting.

Model	Hits@1	Hits@10	Hits@100	Med	MRR	MR	AUC
Box^{2}EL-0	0.00	0.02	0.10	7805	0.01	8925	0.61
Box^{2}EL-1	**0.04**	**0.19**	0.35	694	**0.09**	**4501**	**0.81**
Box^{2}EL-2	**0.04**	0.18	**0.36**	643	**0.09**	4511	**0.81**
Box^{2}EL-3	**0.04**	0.18	**0.36**	**635**	0.08	4513	**0.81**

5
Related Work

Having introduced our novel method Box^{2}EL and demonstrated its effectiveness on a wide variety of benchmarks, we now compare our approach to related work. We begin with a review of related DLE methods, subsequently draw comparisons to KGEs, and finally give a brief overview of other approaches in the broad field of neuro-symbolic reasoning.

Description logic embedding models. A number of methods for learning embeddings of various DLs have been proposed in recent years. We build upon the framework for learning geometric models of $\mathcal{EL}^{++}$ ontologies introduced by Kulmanov et al. [2019] and refined in subsequent work [Mondal et al., 2021; Mohapatra et al., 2021; Xiong et al., 2022; Peng et al., 2022]. The primary difference between this previous work and our approach is our novel role representation based on BoxE. Additionally, some of or loss functions differ slightly from the literature, and we employ a different negative sampling procedure. Furthermore, none of the methods have so far been evaluated on predicting complex concepts.

Our concept representation based on boxes has previously been used in [Xiong et al., 2022; Peng et al., 2022]. In contrast, the other $\mathcal{EL}^{++}$ embedding models represent concepts as n-balls, which has conceptual disadvantages. Mondal et al.

[2021] incorporate further $\mathcal{EL}^{++}$ axioms involving roles into their model, which we do not consider. Mohapatra et al. [2021] aim to solve the same problem as we, namely overcoming the limitations of TransE when it comes to *one-to-many*, *many-to-one*, or *many-to-many* relationships, but their model is only a simple adaption of TransE and comparable to the role representation used by Xiong et al. [2022].

Going beyond $\mathcal{EL}^{++}$, Özçep et al. [2020] introduce a cone-based model for the more expressive DL $\mathcal{ALC}$; however, their contribution is mainly theoretical since they do not provide an implementation or experiments. Embed2Reason [Garg et al., 2019] is another embedding approach for learning $\mathcal{ALC}$ based on quantum logic [Birkhoff and Von Neumann, 1936]. In contrast to our work, its focus lies on ABox instead of subsumption reasoning.

Finally, there exist a variety of other embedding approaches for OWL ontologies that differ from our approach in that they require textual annotation data and cannot model logical structure directly [Smaili et al., 2018; Smaili et al., 2019; Chen et al., 2021].

Knowledge graph embedding models. A vast number of KGE models have been proposed recently [Nickel et al., 2011; Yang et al., 2015; Trouillon et al., 2016; Schlichtkrull et al., 2018; Balazevic et al., 2019], an overview of which can be found in e.g. [Q. Wang et al., 2017; S. Ji et al., 2022]. However, since they are concerned with KGs, most of these methods can be thought of as only modelling the relational part of the ABox of an ontology, whereas we learn embeddings for both the ABox and TBox.

Some KGE methods take background knowledge into account and are thus more similar to our approach, especially in the setting of using DLEs for link prediction. However, while most of these methods focus on embedding logical rules concerning relations [Rocktäschel et al., 2015; Q. Wang et al., 2015; Guo et al., 2016; Nayyeri et al., 2020; Nayyeri et al., 2021], our work is mainly concerned with subsumptions between concepts. Furthermore, link prediction is only one possible application of DLEs, which can also be used to predict new logical background knowledge itself in the form of subsumptions.

Neuro-symbolic reasoning. Our work fits within the broad body of research that tries to combine symbolic and sub-symbolic reasoning techniques, sometimes called *neuro-symbolic reasoning*. While embedding-based methods such as KGEs and DLEs are one possible approach to integrate these two paradigms, there has been a vast amount of work that makes use of different techniques. These include deep learning based approaches [Eberhart et al., 2019; Hohenecker and Lukasiewicz, 2020] and differentiable proving techniques inspired by classic symbolic methods [Rocktäschel and Riedel, 2017; Minervini et al., 2018], to name but a few.

6 Conclusion and Future Work

6.1 Summary

We have introduced Box^{2}EL, a novel $\mathcal{EL}^{++}$ embedding method that adapts the relation representation of BoxE [Abboud et al., 2020] to the setting of DLs. We motivated the need for a novel role representation by illustrating the limitations of TransE [Bordes et al., 2013], which forms the basis of all current $\mathcal{EL}^{++}$ embedding models.

Subsequently, we gave a detailed description of the training procedure of Box^{2}EL. As in previous approaches, we learn ontology embeddings by optimising a variety of loss functions corresponding to TBox axioms. We justified these loss functions intuitively, and provided a formal proof that they are sound, i.e. give rise to embeddings that correspond to logical models of the given ontology.

Our empirical results on several real-world ontologies in the three different settings of subsumption prediction, link prediction, and deductive reasoning show that Box^{2}EL consistently outperforms state-of-the-art $\mathcal{EL}^{++}$ embedding models, demonstrating the expressiveness of our novel role representation. Furthermore, we presented a new benchmark to evaluate the inductive reasoning capability of DLE models regarding predicting subsumptions between named and complex concepts. Overall, our results

align with previous work and show that DLEs are a powerful set of techniques that can be successfully applied in a wide variety of different settings.

6.2 Critical evaluation

We briefly outline a few aspects of our work that could have been improved.

The first two points concern our evaluation and the benchmark we have introduced. While this novel benchmark does allow us to evaluate the subsumption prediction performance of DLE methods regarding complex concepts, the complex concepts we consider are only of the simplest form. In the beginning of this project, our goal was to evaluate current embedding techniques on more involved complex concepts and investigate at what point they might fail.

Furthermore, we have not been able to evaluate all current state-of-the-art DLE methods on our benchmark. In particular, EmEL^{++} [Mondal et al., 2021] and EmEL-var [Mohapatra et al., 2021] are missing from our evaluation, although both approaches contain interesting ideas that we initially wanted to evaluate as well.

Finally, we note that our soundness proof in Theorem 3.5.1 relies on regularising the bump vectors to 0. While regularisation is well-justified and significantly improves the performance of Box^{2}EL in practice (see Section 3.4.4), it would be desirable to have a different proof that does not require the model to be regularised. This is particularly the case because we have also shown that bump vectors play an essential part in the performance of our model (Section 4.6.2).

6.3 Future work

Our approach offers several interesting directions for future work.

Minor improvements. Recall that the first step in our training procedure is to eliminate the ABox from a given ontology. However, this means that individuals will be represented as boxes in the embedding space as well, while it makes conceptually more sense to treat them as points in the vector space. Previous results have confirmed

that this improves the quality of the learnt embeddings [Xiong et al., 2022], and it should be relatively straightforward to incorporate this idea to Box^{2}EL.

Furthermore, we found our model to be somewhat susceptible to the choice of hyperparameters. We believe that with more sophisticated techniques such as Bayesian optimisation, we might be able to find even better hyperparameters that further improve the performance of our model. Recent work has shown this to be the case for KGEs [Ruffinelli et al., 2019].

More expressive DLs. Another interesting idea for future work is to adapt our approach to more expressive DLs such as $\mathcal{ALC}$ or $\mathcal{SROIQ}$. A starting point might be recent work that investigates embeddings for more expressive DLs [Özçep et al., 2020; Garg et al., 2019].

Improving deductive reasoning performance. Finally, as we have noted in our evaluation, we find that the performance of our model is worse in the deductive reasoning than the subsumption prediction setting. It would be interesting to explore new ideas to address this issue and improve deductive reasoning performance, for example potentially by modifying the ranking function used in our model.

Bibliography

Abboud, R., İ. İ. Ceylan, T. Lukasiewicz and T. Salvatori (2020). 'BoxE: A Box Embedding Model for Knowledge Base Completion'. In: *Advances in Neural Information Processing Systems*. NeurIPS 2020. Vol. 33, pp. 9649–9661. URL: https://proceedings.neurips.cc/paper/2020/hash/6dbbe6abe5f14af882ff977fc3f35501-Abstract.html.

Alshahrani, M., M. A. Khan, O. Maddouri, A. R. Kinjo, N. Queralt-Rosinach and R. Hoehndorf (2017). 'Neuro-Symbolic Representation Learning on Biological Knowledge Graphs'. In: *Bioinformatics* 33.17, pp. 2723–2730. DOI: 10.1093/bioinformatics/btx275.

Ashburner, M., C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin and G. Sherlock (2000). 'Gene Ontology: Tool for the Unification of Biology'. In: *Nature Genetics* 25.1 (1), pp. 25–29. DOI: 10.1038/75556.

Baader, F., S. Brandt and C. Lutz (2005). *Pushing the EL Envelope*. LTCS-Report LCTS-05-01. Institute for Theoretical Computer Science, TU Dresden, 2005. URL: http://lat.inf.tu-dresden.de/research/reports.html.

Baader, F., D. Calvanese, D. L. McGuinness, D. Nardi and P. F. Patel-Schneider, eds. (2003). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press. ISBN: 978-0-521-78176-3.

Baader, F., I. Horrocks, C. Lutz and U. Sattler (2017). *An Introduction to Description Logic*. Cambridge University Press. ISBN: 978-0-521-69542-8.

Balazevic, I., C. Allen and T. Hospedales (2019). 'TuckER: Tensor Factorization for Knowledge Graph Completion'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. EMNLP-IJCNLP, pp. 5185–5194. DOI: 10.18653/v1/D19-1522.

Birkhoff, G. and J. Von Neumann (1936). 'The Logic of Quantum Mechanics'. In: *Annals of Mathematics* 37.4, pp. 823–843. DOI: 10.2307/1968621. JSTOR: 1968621.

Bollacker, K., C. Evans, P. Paritosh, T. Sturge and J. Taylor (2008). 'Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge'. In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. SIGMOD '08, pp. 1247–1250. DOI: 10.1145/1376616.1376746.

Bordes, A., N. Usunier, A. Garcia-Duran, J. Weston and O. Yakhnenko (2013). 'Translating Embeddings for Modeling Multi-relational Data'. In: *Advances in Neural Information Processing Systems*. NIPS 2013. Vol. 26. URL: https://proceedings.neurips.cc/paper/2013/hash/1cecc7a77928ca8133fa24680a88d2f9-Abstract.html.

Bordes, A., J. Weston, R. Collobert and Y. Bengio (2011). 'Learning Structured Embeddings of Knowledge Bases'. In: *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*. AAAI 2011. URL: https://www.aaai.org/ocs/index.php/AAAI/AAAI11/paper/view/3659.

Chen, J., P. Hu, E. Jimenez-Ruiz, O. M. Holter, D. Antonyrajah and I. Horrocks (2021). 'OWL2Vec*: Embedding of OWL Ontologies'. In: *Machine Learning* 110.7, pp. 1813–1845. DOI: 10.1007/s10994-021-05997-6.

Dettmers, T., P. Minervini, P. Stenetorp and S. Riedel (2018). 'Convolutional 2D Knowledge Graph Embeddings'. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1. URL: https://ojs.aaai.org/index.php/AAAI/article/view/11573.

Eberhart, A., M. Ebrahimi, L. Zhou, C. Shimizu and P. Hitzler (2019). 'Completion Reasoning Emulation for the Description Logic EL+'. In: *arXiv* (preprint). DOI: 10.48550/arXiv.1912.05063.

Fawcett, T. (2006). 'An Introduction to ROC Analysis'. In: *Pattern Recognition Letters* 27.8, pp. 861–874. DOI: 10.1016/j.patrec.2005.10.010.

Garg, D., S. Ikbal, S. K. Srivastava, H. Vishwakarma, H. Karanam and L. V. Subramaniam (2019). 'Quantum Embedding of Knowledge for Reasoning'. In: *Advances in Neural Information Processing Systems*. NeurIPS 2019. Vol. 32. URL: https://proceedings.neurips.cc/paper/2019/hash/cb12d7f933e7d102c52231bf62b8a678-Abstract.html.

Glimm, B., I. Horrocks, B. Motik, G. Stoilos and Z. Wang (2014). 'HermiT: An OWL 2 Reasoner'. In: *Journal of Automated Reasoning* 53.3, pp. 245–269. DOI: 10.1007/s10817-014-9305-1.

Grau, B. C., I. Horrocks, B. Motik, B. Parsia, P. Patel-Schneider and U. Sattler (2008). 'OWL 2: The next Step for OWL'. In: *Journal of Web Semantics*. Semantic Web Challenge 2006/2007 6.4, pp. 309–322. DOI: 10.1016/j.websem.2008.05.001.

Guo, S., Q. Wang, L. Wang, B. Wang and L. Guo (2016). 'Jointly Embedding Knowledge Graphs and Logical Rules'. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 192–202. DOI: 10.18653/v1/D16-1019.

Hoehndorf, R., M. Dumontier, A. Oellrich, S. Wimalaratne, D. Rebholz-Schuhmann, P. Schofield and G. V. Gkoutos (2011). 'A Common Layer of Interoperability for Biomedical Ontologies Based on OWL EL'. In: *Bioinformatics* 27.7, pp. 1001–1008. DOI: 10.1093/bioinformatics/btr058.

Hohenecker, P. and T. Lukasiewicz (2020). 'Ontology Reasoning with Deep Neural Networks'. In: *Journal of Artificial Intelligence Research* 68, pp. 503–540. DOI: 10.1613/jair.1.11661.

Horrocks, I. (2008). 'Ontologies and the Semantic Web'. In: *Communications of the ACM* 51.12, pp. 58–67. DOI: 10.1145/1409360.1409377.

Hoyt, C. T., M. Berrendorf, M. Galkin, V. Tresp and B. M. Gyori (2022). 'A Unified Framework for Rank-based Evaluation Metrics for Link Prediction in Knowledge Graphs'. In: *arXiv* (preprint). DOI: 10.48550/arXiv.2203.07544.

Ji, G., S. He, L. Xu, K. Liu and J. Zhao (2015). 'Knowledge Graph Embedding via Dynamic Mapping Matrix'. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*. ACL 2015, pp. 687–696. DOI: 10.3115/v1/p15-1067.

Ji, G., K. Liu, S. He and J. Zhao (2016). 'Knowledge Graph Completion with Adaptive Sparse Transfer Matrix'. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. AAAI'16. Vol. 30. 1. DOI: 10.1609/aaai.v30i1.10089.

Ji, S., S. Pan, E. Cambria, P. Marttinen and P. S. Yu (2022). 'A Survey on Knowledge Graphs: Representation, Acquisition, and Applications'. In: *IEEE Transactions on Neural Networks and Learning Systems* 33.2, pp. 494–514. DOI: 10.1109/TNNLS.2021.3070843.

Kazakov, Y., M. Krötzsch and F. Simančík (2014). 'The Incredible ELK'. In: *Journal of Automated Reasoning* 53.1, pp. 1–61. DOI: 10.1007/s10817-013-9296-3.

Kingma, D. P. and J. Ba (2015). 'Adam: A Method for Stochastic Optimization'. In: *3rd International Conference on Learning Representations*. ICLR 2015. DOI: 10.48550/arXiv.1412.6980.

Krompaß, D., S. Baier and V. Tresp (2015). 'Type-Constrained Representation Learning in Knowledge Graphs'. In: *The Semantic Web*. ISWC 2015. Lecture Notes in Computer Science, pp. 640–655. DOI: 10.1007/978-3-319-25007-6_37.

Kulmanov, M. and R. Hoehndorf (2017). 'Evaluating the Effect of Annotation Size on Measures of Semantic Similarity'. In: *Journal of Biomedical Semantics* 8.1 (1), pp. 1–10. DOI: 10.1186/s13326-017-0119-z.

Kulmanov, M., W. Liu-Wei, Y. Yan and R. Hoehndorf (2019). 'EL Embeddings: Geometric Construction of Models for the Description Logic EL++'. In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. IJCAI-19. DOI: 10.24963/ijcai.2019/845.

Lin, Y., Z. Liu, M. Sun, Y. Liu and X. Zhu (2015). 'Learning Entity and Relation Embeddings for Knowledge Graph Completion'. In: *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*. AAAI'15, pp. 2181–2187.

Mendez, J. (2012). 'Jcel: A Modular Rule-Based Reasoner'. In: *Proceedings of the 1st International Workshop on OWL Reasoner Evaluation*. ORE-2012. Vol. 858. CEUR Workshop Proceedings. URL: http://ceur-ws.org/Vol-858/ore2012_paper12.pdf.

Minervini, P., M. Bosnjak, T. Rocktäschel and S. Riedel (2018). 'Towards Neural Theorem Proving at Scale'. In: *arXiv* (preprint). DOI: 10.48550/arXiv.1807.08204.

Mohapatra, B., S. Bhatia, R. Mutharaju and G. Srinivasaraghavan (2021). 'EmELvar: A NeuroSymbolic Reasoner for the EL++ Description Logic'. In: *Proceedings of the Semantic Reasoning Evaluation Challenge*. SemREC 2021. Vol. 3123. CEUR Workshop Proceedings, pp. 44–51. URL: http://ceur-ws.org/Vol-3123/#paper6.

Mondal, S., S. Bhatia and R. Mutharaju (2021). 'EmEL++: Embeddings for EL++ Description Logic'. In: *Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering*. AAAI-MAKE 2021. Vol. 2846. CEUR Workshop Proceedings. URL: http://ceur-ws.org/Vol-2846/paper19.pdf.

Mungall, C. J., C. Torniai, G. V. Gkoutos, S. E. Lewis and M. A. Haendel (2012). 'Uberon, an Integrative Multi-Species Anatomy Ontology'. In: *Genome Biology* 13.1, R5. DOI: 10.1186/gb-2012-13-1-r5.

Nathani, D., J. Chauhan, C. Sharma and M. Kaul (2019). 'Learning Attention-based Embeddings for Relation Prediction in Knowledge Graphs'. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. ACL 2019, pp. 4710–4723. DOI: 10.18653/v1/P19-1466.

Nayyeri, M., C. Xu, M. M. Alam, J. Lehmann and H. Shariat Yazdi (2021). 'LogicENN: A Neural Based Knowledge Graphs Embedding Model with Logical Rules'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. DOI: 10.1109/TPAMI.2021.3121646.

Nayyeri, M., C. Xu, S. Vahdati, N. Vassilyeva, E. Sallinger, H. S. Yazdi and J. Lehmann (2020). 'Fantastic Knowledge Graph Embeddings and How to Find the Right Space for Them'. In: *The Semantic Web*. ISWC 2020. Lecture Notes in Computer Science, pp. 438–455. DOI: 10.1007/978-3-030-62419-4_25.

Nickel, M., K. Murphy, V. Tresp and E. Gabrilovich (2016). 'A Review of Relational Machine Learning for Knowledge Graphs'. In: *Proceedings of the IEEE* 104.1, pp. 11–33. DOI: 10.1109/JPROC.2015.2483592.

Nickel, M., V. Tresp and H.-P. Kriegel (2011). 'A Three-Way Model for Collective Learning on Multi-Relational Data'. In: *Proceedings of the 28th International Conference on Machine Learning*. ICML 2011. URL: https://openreview.net/forum?id=H14QEiZ_WS.

Özçep, Ö. L., M. Leemhuis and D. Wolter (2020). 'Cone Semantics for Logics with Negation'. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. IJCAI-20. Vol. 2, pp. 1820–1826. DOI: 10.24963/ijcai.2020/252.

Paszke, A., S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala (2019). 'PyTorch: An Imperative Style, High-Performance Deep Learning Library'. In: *Advances in Neural Information Processing Systems*. NeurIPS 2019. Vol. 32. URL: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Peng, X., Z. Tang, M. Kulmanov, K. Niu and R. Hoehndorf (2022). 'Description Logic EL++ Embeddings with Intersectional Closure'. In: *arXiv* (preprint). DOI: 10.48550/arXiv.2202.14018.

Pintscher, L. (2022). *Wikidata:Statistics*. URL: https://www.wikidata.org/wiki/Wikidata:Statistics (visited on 07/08/2022).

Rector, A. L., J. E. Rogers and P. Pole (1996). 'The GALEN High Level Ontology'. In: *Medical Informatics Europe '96*. Studies in Health Technology and Informatics, pp. 174–178. DOI: 10.3233/978-1-60750-878-6-174.

Rocktäschel, T. and S. Riedel (2017). 'End-to-End Differentiable Proving'. In: *Advances in Neural Information Processing Systems*. Vol. 30. URL: https://proceedings.neurips.cc/paper/2017/hash/b2ab001909a8a6f04b51920306046ce5-Abstract.html.

Rocktäschel, T., S. Singh and S. Riedel (2015). 'Injecting Logical Background Knowledge into Embeddings for Relation Extraction'. In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1119–1129. DOI: 10.3115/v1/N15-1118.

Ruffinelli, D., S. Broscheit and R. Gemulla (2019). 'You CAN Teach an Old Dog New Tricks! On Training Knowledge Graph Embeddings'. In: International Conference on Learning Representations, ICLR 2020. URL: https://openreview.net/forum?id=BkxSmlBFvr.

Schlichtkrull, M., T. N. Kipf, P. Bloem, R. van den Berg, I. Titov and M. Welling (2018). 'Modeling Relational Data with Graph Convolutional Networks'. In: *The Semantic Web*. ESWC 2018. Lecture Notes in Computer Science, pp. 593–607. DOI: 10.1007/978-3-319-93417-4_38.

Schulz, S., B. Suntisrivaraporn, F. Baader and M. Boeker (2009). 'SNOMED Reaching Its Adolescence: Ontologists' and Logicians' Health Check'. In: *International Journal of Medical Informatics*. MedInfo 2007 78, S86–S94. DOI: 10.1016/j.ijmedinf.2008.06.004.

Smaili, F. Z., X. Gao and R. Hoehndorf (2018). 'Onto2Vec: Joint Vector-Based Representation of Biological Entities and Their Ontology-Based Annotations'. In: *Bioinformatics* 34.13, pp. i52–i60. DOI: 10.1093/bioinformatics/bty259.

Smaili, F. Z., X. Gao and R. Hoehndorf (2019). 'OPA2Vec: Combining Formal and Informal Content of Biomedical Ontologies to Improve Similarity-Based Prediction'. In: *Bioinformatics* 35.12, pp. 2133–2140. DOI: 10.1093/bioinformatics/bty933.

Smith, B., M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall, N. Leontis, P. Rocca-Serra, A. Ruttenberg, S.-A. Sansone, R. H. Scheuermann, N. Shah, P. L. Whetzel and S. Lewis (2007). 'The OBO Foundry: Coordinated Evolution of Ontologies to Support Biomedical Data Integration'. In: *Nature Biotechnology* 25.11 (11), pp. 1251–1255. DOI: 10.1038/nbt1346.

Sun, Z., Z.-H. Deng, J.-Y. Nie and J. Tang (2019). 'RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space'. In: *International Conference on Learning Representations*. ICLR 2019. URL: https://openreview.net/forum?id=HkgEQnRqYQ.

Szklarczyk, D., A. L. Gable, K. C. Nastou, D. Lyon, R. Kirsch, S. Pyysalo, N. T. Doncheva, M. Legeay, T. Fang, P. Bork, L. J. Jensen and C. von Mering (2021). 'The STRING Database in 2021: Customizable Protein–Protein Networks, and Functional Characterization of User-Uploaded Gene/Measurement Sets'. In: *Nucleic Acids Research* 49.D1, pp. D605–D612. DOI: 10.1093/nar/gkaa1074.

Trouillon, T., J. Welbl, S. Riedel, E. Gaussier and G. Bouchard (2016). 'Complex Embeddings for Simple Link Prediction'. In: *Proceedings of The 33rd International Conference on Machine Learning*, pp. 2071–2080. URL: https://proceedings.mlr.press/v48/trouillon16.html.

Tsarkov, D. and I. Horrocks (2006). 'FaCT++ Description Logic Reasoner: System Description'. In: *Automated Reasoning*. IJCAR 2006. Lecture Notes in Computer Science, pp. 292–297. DOI: 10.1007/11814771_26.

Vrandečić, D. and M. Krötzsch (2014). 'Wikidata: A Free Collaborative Knowledgebase'. In: *Communications of the ACM* 57.10, pp. 78–85. DOI: 10.1145/2629489.

Wang, Q., Z. Mao, B. Wang and L. Guo (2017). 'Knowledge Graph Embedding: A Survey of Approaches and Applications'. In: *IEEE Transactions on Knowledge and Data Engineering* 29.12, pp. 2724–2743. DOI: 10.1109/TKDE.2017.2754499.

Wang, Q., B. Wang and L. Guo (2015). 'Knowledge Base Completion Using Embeddings and Rules'. In: *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*. IJCAI-15. URL: https://www.aaai.org/ocs/index.php/IJCAI/IJCAI15/paper/view/10798.

Wang, Z., J. Zhang, J. Feng and Z. Chen (2014). 'Knowledge Graph Embedding by Translating on Hyperplanes'. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 28.1. DOI: 10.1609/aaai.v28i1.8870.

West, R., E. Gabrilovich, K. Murphy, S. Sun, R. Gupta and D. Lin (2014). 'Knowledge Base Completion via Search-Based Question Answering'. In: *Proceedings of the 23rd International Conference on World Wide Web*. WWW'14, pp. 515–526. DOI: 10.1145/2566486.2568032.

Xiong, B., N. Potyka, T.-K. Tran, M. Nayyeri and S. Staab (2022). 'Box Embeddings for the Description Logic EL++'. In: *arXiv* (preprint). DOI: 10.48550/arXiv.2201.09919.

Yang, B., W.-t. Yih, X. He, J. Gao and L. Deng (2015). 'Embedding Entities and Relations for Learning and Inference in Knowledge Bases'. In: *3rd International Conference on Learning Representations*. ICLR 2015. DOI: 10.48550/arXiv.1412.6575.