DELE: Deductive $\mathcal{EL}^{++}$ Embeddings for Knowledge Base Completion

NeSy 2024 Special Issue, paper # 51

Olga Mashkova [a], Fernando Zhapa-Camacho [a] and Robert Hoehndorf [a,b,c,*]

[a] *Computer Science Program, Computer, Electrical, and Mathematical Sciences & Engineering Division, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia*

[b] *KAUST Center of Excellence for Smart Health, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia*

[c] *KAUST Center of Excellence for Generative AI, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia*

E-mail: robert.hoehndorf@kaust.edu.sa

Abstract.
Ontology embeddings map classes, relations, and individuals in ontologies into $\mathbb{R}^n$, and within $\mathbb{R}^n$ similarity between entities can be computed or new axioms inferred. For ontologies in the Description Logic $\mathcal{EL}^{++}$, several optimization-based embedding methods have been developed that explicitly generate models of an ontology. However, these methods suffer from some limitations; they do not distinguish between statements that are unprovable and provably false, and therefore they may use entailed statements as negatives. Furthermore, they do not utilize the deductive closure of an ontology to identify statements that are inferred but not asserted. We evaluated a set of embedding methods for $\mathcal{EL}^{++}$ ontologies, incorporating several modifications that aim to make use of the ontology deductive closure. In particular, we designed novel negative losses that account both for the deductive closure and different types of negatives and formulated evaluation methods for knowledge base completion. We demonstrate that our embedding methods improve over the baseline ontology embedding in the task of knowledge base or ontology completion.

Keywords: Ontology Embedding, Knowledge Base Completion, Description Logic $\mathcal{EL}^{++}$

1. Introduction

Several methods have been developed to embed Description Logic theories or ontologies in vector spaces [7, 8, 15, 21, 32, 34, 36, 46]. These embedding methods preserve some aspects of the semantics in the vector space, and may enable the computation of semantic similarity, inferring axioms that are entailed, and predicting axioms that are not entailed but may be added to the theory. For the lightweight Description Logic $\mathcal{EL}^{++}$, several geometric embedding methods have been developed [15, 21, 32, 34, 46]. They can be proven to "faithfully" approximate a model in the sense that, if a certain optimization objective is reached (usually, a loss function reduced to 0), the embedding method has constructed a model of the $\mathcal{EL}^{++}$theory. Geometric model construction enables the execution of various tasks. These tasks include knowledge base completion and subsumption prediction via either

*Corresponding author. E-mail: robert.hoehndorf@kaust.edu.sa.

testing the truth of a statement under consideration in a single (approximate) model or aggregating truth values over multiple models.

Advances on different geometric embedding methods have usually focused on the expressiveness of the embedding methods; originally, hyperballs [21] where used to represent the interpretation of concept symbols, yet hyperballs are not closed under intersection. Therefore, axis-aligned boxes were introduced [15, 36, 46]. Furthermore, $\mathcal{EL}^{++}$allows for axioms pertaining to relations, and several methods have extended the way in which relations are modeled [15, 21, 46]. However, there are several aspects of geometric embeddings that have not yet been investigated. In particular, for $\mathcal{EL}^{++}$, there are sound and complete reasoners with efficient implementations that scale to very large knowledge bases [19]; it may therefore be possible to utilize a deductive reasoner together with the embedding process to improve generation of embeddings that represent geometric models.

We evaluate geometric embedding methods and incorporate deductive inference into the training process. We use the *ELEmbeddings* [21], *ELBE* [36], and *Box*^{2}*EL* [15] models for our experiments; however, our results also apply to other geometric embedding methods for $\mathcal{EL}^{++}$.

Our main contributions are as follows:

- We propose loss functions that incorporate negative samples in all normal forms and account for deductive closure during training.
- We introduce a fast approximate algorithm for computing the deductive closure of an $\mathcal{EL}^{++}$theory and use it to improve negative sampling during model training.
- We formulate evaluation methods for knowledge base completion that account for the deductive closure during evaluation.

This is an extended version of our previous work [25]. Here, we include a more comprehensive treatment of computing the deductive closure and using the deductive closure with $\mathcal{EL}^{++}$embedding methods (e.g., for model evaluation). We include additional experiments on three benchmark datasets: besides protein function prediction and protein–protein interaction, we study the subsumption prediction task on Food Ontology [11] and GALEN ontology [39]. Furthermore, we consider two additional geometric ontology embedding models, *ELBE* [36] and *Box*^{2}*EL* [15], apart from *ELEmbeddings* [21] for which we also extend our methods. We make our code and data available at https://github.com/bio-ontology-research-group/DELE.

2. Preliminaries

2.1. Description Logic $\mathcal{EL}^{++}$

Let $\Sigma = (\mathbf{C}, \mathbf{R}, \mathbf{I})$ be a signature with set $\mathbf{C}$ of concept names, $\mathbf{R}$ of role names, and $\mathbf{I}$ of individual names. Given $A, B \in \mathbf{C}$, $r \in \mathbf{R}$, and $a, b \in \mathbf{I}$, $\mathcal{EL}^{++}$concept descriptions are constructed with the grammar $\bot \mid \top \mid A \sqcap B \mid \exists r.A \mid \{a\}$. ABox axioms are of the form $A(a)$ and $r(a, b)$, TBox axioms are of the form $A \sqsubseteq B$, and RBox axioms are of the form $r_1 \circ r_2 \circ \cdots \circ r_n \sqsubseteq r$. $\mathcal{EL}^{++}$ *generalized concept inclusions* (GCIs) and *role inclusions* (RIs) can be normalized to follow one of the forms listed in Table 1 [2]. If an ontology contains a non-empty ABox, then, first, concept assertion axioms of the form $A(a)$ are processed: for each complex concept description A, a new concept name C is introduced and axiom $A \equiv C$ is added to the TBox [35]. Normalization rules for TBox axioms [2] can be found in Table 2. They preserve the same models for ontologies [26] and the normalized ontology is a conservative extension of the original ontology [42]. Following previous works [15, 21, 36] we develop our methodology specifically for normalized $\mathcal{EL}^{++}$ontologies. The advantage of normalizing $\mathcal{EL}^{++}$ontologies is that their deductive closure is finite whereas the deductive closure of non-normalized $\mathcal{EL}^{++}$ontologies is not. Therefore, in our work, when we refer to the language $\mathcal{EL}^{++}$, we always intend the normalized $\mathcal{EL}^{++}$, i.e., where all valid axiom types are those listed in Table 1.

To define the semantics of an $\mathcal{EL}^{++}$ theory, we use [2] an *interpretation domain* $\Delta^{\mathcal{I}}$ and an *interpretation function* $\cdot^{\mathcal{I}}$. For every concept $A \in \mathbf{C}$, $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$; individual $a \in \mathbf{I}$, $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$; role $r \in \mathbf{R}$, $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Furthermore, the

Table 1

Normalized forms of $\mathcal{EL}^{++}$ generalized concept inclusions (GCIs) and role inclusions (RIs)

Acronym	Axiom type
GCI0	$C \sqsubseteq D$
GCI1	$C \sqcap D \sqsubseteq E$
GCI2	$C \sqsubseteq \exists R.D$
GCI3	$\exists R.C \sqsubseteq D$
GCI0-BOT	$C \sqsubseteq \bot$
GCI1-BOT	$C \sqcap D \sqsubseteq \bot$
GCI3-BOT	$\exists R.C \sqsubseteq \bot$
RI0	$r \sqsubseteq s$
RI1	$r_1 \circ r_2 \sqsubseteq s$

Table 2

Normalization rules for TBox for $\mathcal{EL}^{++}$ based on Baader et al. [2]. Here, $r, r_1, \ldots, r_k, s$ are role names, C, D, B are arbitrary concept descriptions, $\hat{C}, \hat{D}$ are complex concept descriptions (i.e., not concept names), A is a fresh concept name and u is a fresh role name.

Input	Output
$r_1 \circ \cdots \circ r_k \sqsubseteq s$	$r_1 \circ \cdots \circ r_{k-1} \sqsubseteq u,\ u \circ r_k \sqsubseteq s$
$C \sqcap \hat{D} \sqsubseteq E$	$\hat{D} \sqsubseteq A,\ C \sqcap A \sqsubseteq E$
$\exists r.\hat{C} \sqsubseteq D$	$\hat{C} \sqsubseteq A,\ \exists r.A \sqsubseteq D$
$\bot \sqsubseteq D$	$\emptyset$
$\hat{C} \sqsubseteq \hat{D}$	$\hat{C} \sqsubseteq A,\ A \sqsubseteq \hat{D}$
$B \sqsubseteq \exists r.\hat{C}$	$B \sqsubseteq \exists r.A,\ A \sqsubseteq \hat{C}$
$B \sqsubseteq C \sqcap D$	$B \sqsubseteq C,\ B \sqsubseteq D$

semantics for other $\mathcal{EL}^{++}$ constructs are the following (omitting concrete domains and role inclusions):

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

An interpretation $\mathcal{I}$ is a model for an axiom $C \sqsubseteq D$ if and only if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, for an axiom $B(a)$ if and only if $a^{\mathcal{I}} \in B^{\mathcal{I}}$; and for an axiom $r(a,b)$ if and only if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ [3]. The relation of semantic entailment, $\models$, is defined as a relation between a theory $\mathcal{T}$ and axiom ϕ: $\mathcal{T} \models \phi$ iff every model of $\mathcal{T}$ is also a model of ϕ ($Mod(\mathcal{T}) \subseteq Mod(\{\phi\})$) [43].

2.2. Knowledge Base Completion

The task of knowledge base completion is the addition (or prediction) of axioms which hold yet are not represented in the knowledge base. We call the task "ontology completion" when exclusively TBox axioms are predicted. The task of knowledge base completion may encompass both deductive [18, 40] and inductive [5, 12] inference processes and give rise to two subtly different tasks: adding only "novel" axioms to a knowledge base that are *not* in the deductive closure of the knowledge base, and adding axioms that are in the deductive closure as well as some "novel" axioms that are not deductively inferred; both tasks are related but differ in how they are evaluated.

Inductive inference, analogously to knowledge graph completion [9], predicts axioms based on patterns and regularities within the knowledge base. Knowledge base completion, or ontology completion, can be further distinguished based on the information that is used to predict "novel" axioms. We distinguish between two approaches to knowledge base completion: (1) knowledge base completion which relies solely on (formalized) information within the knowledge base to predict new axioms, and (2) knowledge base completion which incorporates side information, such as text, to enhance the prediction of new axioms. Here, we mainly consider the first case.

2.3. Deductive Closure

The *deductive closure* of a theory T refers to the smallest set containing all statements which can be inferred by deductive reasoning over T; for a given deductive relation $\vdash$, we call $T^{\vdash} = \{\phi \mid T \vdash \phi\}$ the deductive closure of T. In knowledge bases, the deductive closure is usually not identical to the asserted axioms in the knowledge base; it is also usually infinite. Representing the deductive closure is challenging since it is infinite, but, in $\mathcal{EL}^{++}$, any knowledge base can be normalized to one of the seven normal forms; therefore, we can compute the deductive closure with respect to these normal forms, and this set will be finite (as long as the sets of concept and role names are finite). For example, all entailed axioms of type $C \sqsubseteq D$ will be a subset of the set of all possible axioms of GCI0 type having cardinality $|\mathbf{C}|^2$ where $|\mathbf{C}|$ is the cardinality of the set of all concept names. Similarly, the cardinality of GCI0-BOT deductive closure will be limited by $|\mathbf{C}|$, GCI1 deductive closure cardinality will be limited by $|\mathbf{C}|^3$, and GCI1-BOT deductive closure cardinality by $|\mathbf{C}|^2$ since one of the concepts is fixed. GCI2 and GCI3 deductive closures cardinality will depend on the total number of relations $|\mathbf{R}|$ and will be limited by $|\mathbf{C}|^2 \cdot |\mathbf{R}|$, and, finally, the number of entailed axioms of GCI3-BOT type will not exceed $|\mathbf{C}| \cdot |\mathbf{R}|$.

3. Related Work

3.1. Graph-Based Ontology Embeddings

Graph-based ontology embeddings rely on a construction (projection) of graphs from ontology axioms mapping ontology classes, individuals and roles to nodes and labeled edges [48]. Embeddings for nodes and edge labels are optimized following two strategies: by generating random walks and using a sequence learning method such as Word2Vec [30] or by using Knowledge Graph Embedding (KGE) methods [44]. These type of methods have been shown effective on knowledge base and ontology completion [7] and have been applied to domain-specific tasks such as protein–protein interaction prediction [7] or gene–disease association prediction [1, 8]. Graph-based methods rely on adjacency information of the ontology structure but cannot easily handle logical operators and do not approximate ontology models. Therefore, graph-based methods are not "faithful", i.e., do not approximate models, do not allow determining whether statements are "true" in these models, and therefore cannot be used to perform semantic entailment.

3.2. Geometric-Based Ontology Embeddings

Multiple methods have been developed for the geometric construction of models for the $\mathcal{EL}^{++}$ language. ELEmbeddings [21] constructs an interpretation of concept names as sets of points lying within an open n-dimensional ball and generates an interpretation of role names as the set of pairs of points that are separated by a vector in $\mathbb{R}^n$, i.e., by the embedding of the role name. EmEL++ [32] extends ELEmbeddings with more expressive constructs such as role chains and role inclusions. Another extension of ELEmbeddings, EmELvar [31], introduces role embeddings which enables handling many-to-many relations and provides a perspective to extending the method to more expressive Description Logics. ELBE [36] and BoxEL [46] use n-dimensional axis-aligned boxes to represent concepts, which has an advantage over balls because the intersection of two axis-aligned boxes is a box whereas the intersection of two n-balls is not an n-ball. BoxEL additionally preserves ABox facilitating a more accurate representation of knowledge base's logical structure by ensuring, e.g., that an entity has the minimal volume. Box^{2}EL [15] represents ontology roles more expressively with two boxes encoding the semantics of the domain and codomain of roles. Box^{2}EL enables the expression of one-to-many relations as opposed to other methods. The box-based method TransBox [47] aims to effectively capture all $\mathcal{EL}^{++}$ logical operations allowing for precise representation of any arbitrarily complex concept description. The fact that any normalized $\mathcal{EL}^{++}$ theory has a finite deductive closure allows the definition of a canonical model where all and only entailed axioms are true. This has led to the development of strongly faithful method based on convex sets in n-dimensional real-valued space [23]. Its extension, FaithEL [22], interprets concepts and relations as subsets of unitary hypercubes. Although these geometric methods construct a "canonical" model of an ontology, they are not intended to predict new knowledge, which however is useful in real-world applications: knowledge bases may be incomplete, and ontology embedding methods designed for knowledge base completion would be able to find a tradeoff between the prediction of entailed and novel knowledge. Axis-aligned cone-shaped geometric model introduced in [34] deals with $\mathcal{ALC}$ ontologies and allows for full negation of concepts and existential quantification by construction of convex sets in $\mathbb{R}^n$. This work has not yet been implemented or evaluated in an application.

3.3. *Knowledge Base Completion Task*

Several recent advancements in the knowledge base completion rely on side information as included in Large Language Models (LLMs). Ji et al. [17] explores how pretrained language models can be utilized for incorporating one ontology into another, with the main focus on inconsistency handling and ontology coherence. HalTon [6] addresses the task of event ontology completion via simultaneous event clustering, hierarchy expansion and type naming utilizing BERT [10] for instance encoding. Li et al. [24] formulates knowledge base completion task as a Natural Language Inference (NLI) problem and examines how this approach may be combined with concept embeddings for identifying missing knowledge in ontologies. As for other approaches, Mežnar et al. [29] proposes a method that converts an ontology into a graph to recommend missing edges using structure-only link analysis methods, Shiraishi et al. [41] constructs matrix-based ontology embeddings which capture the global and local information for subsumption prediction. All these methods use side information from LLMs and would not be applicable, for example, in the case where a knowledge base is private or consists of only identifiers; we do not consider methods based on pre-trained LLMs here as baselines.

4. Negative sampling and Objective Functions

Currently available geometric ontology embedding models which construct a model of an ontology by optimizing some objective function usually sample negative examples during training phase [15, 21, 31, 32, 36, 47]. This operation prevents overgeneralization of learned embeddings and trivial satisfiability in case a model collapses [21, 47] by incorporating additional constraints within a model. Ontology embedding methods select negatives by replacing one of the concepts with a randomly chosen one (either from the set of all concept names, or a subset thereof). *ELEmbeddings*, *ELBE* and *Box*^{2}*EL* use a single loss for "negatives", i.e., axioms that are not included in the knowledge base; the loss is used only for axioms of the form $C \sqsubseteq \exists R.D$ (GCI2) which are randomly sampled; negatives are not sampled for other normal forms. Correspondingly, the embedding methods were primarily evaluated on predicting GCI2 axioms (*Box*^{2}*EL* was also evaluated on subsumption prediction); this evaluation procedure might have introduced biases towards axioms of type GCI2, and influenced the ability of geometric models to predict axioms of other types. Specifically, the lack of negative examples of other axiom types may lead to geometric models in which many axioms are true even if they are not entailed, leading to a decreased ability to find axioms that can be added to a theory in the task of knowledge base completion. Consequently, we also sample negatives for other normal forms and add "negative" losses (i.e., losses for the sampled "negatives") for all other normal forms.

4.1. *ELEmbeddings Negative Losses*

For negative loss construction in *ELEmbeddings*, we employ notations from the *ELEmbeddings* method where $r_\eta(c)$, $r_\eta(d)$, $r_\eta(e)$ and $f_\eta(c)$, $f_\eta(d)$, $f_\eta(e)$ denote the radius and the ball center associated with classes c, d, e, respectively; and $f_\eta(r)$ denotes the embedding vector associated with relation r. γ stands for a margin parameter, and ε is a small positive number. There is a geometric part as well as a regularization part for each new negative loss forcing class centers to lie on a unit ℓ_2-sphere.

For *ELEmbeddings*, as reflected in Eq. 1, we use the original GCI1-BOT loss for disjoint classes; although non-containment of ball corresponding to C within the ball corresponding to D is not equivalent to their disjointness, the loss aims to minimize the classes' overlap for better optimization:

$$loss_{C \not\sqsubseteq D}(c,d) = \max(0, r_\eta(c) + r_\eta(d) - \|f_\eta(c) - f_\eta(d))\| + \gamma) + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \quad (1)$$

The same logic applies for the negative loss in Eq. 2 where we minimize overlap between the translated ball corresponding to class C and the ball representing D:

$$\begin{aligned} loss_{\exists R.C \not\sqsubseteq D}(r,c,d) = \max(0, r_\eta(c) + r_\eta(d) - \|f_\eta(c) - f_\eta(r) - f_\eta(d))\| + \gamma) + \\ + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \end{aligned} \quad (2)$$

Negative loss 3 is constructed similarly to the $C \sqcap D \sqsubseteq E$ loss: the first part penalizes non-overlap of the classes C and D (we do not consider the disjointness case since, for every class X, we have $\bot \sqsubseteq X$); furthermore, for negative sampling of axioms of this type, we vary only the E part of GCI1 axioms from the ontology, so the intersection of C and D is non-empty by assumption:

$$\begin{aligned} loss_{C \sqcap D \not\sqsubseteq E}(c,d,e) = \max(0, -r_\eta(c) - r_\eta(d) + \|f_\eta(c) - f_\eta(d))\| - \gamma) + \\ + \max(0, r_\eta(c) - \|f_\eta(c) - f_\eta(e))\| + \gamma) + \max(0, r_\eta(d) - \|f_\eta(d) - f_\eta(e))\| + \gamma) + \\ + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| + |\|f_\eta(e)\| - 1| \end{aligned} \quad (3)$$

The second and the third part force the center corresponding to E not to lie in the intersection of balls associated with C and D. Here we do not consider constraints on the radius of the ball for the E class and focus only on the relative positions of the C, D and E class centers and the overlapping of n-balls representing C and D. Since the first part of the loss encourages classes to have a non-empty intersection, we use it as a negative loss for GCI1-BOT axioms:

$$loss_{C \sqcap D \not\sqsubseteq \bot}(c,d) = \max(0, -r_\eta(c) - r_\eta(d) + \|f_\eta(c) - f_\eta(d))\| - \gamma) + |\|f_\eta(c)\| - 1| + |\|f_\eta(d)\| - 1| \quad (4)$$

In the original method losses for axioms of type GCI0-BOT and GCI3-BOT force radii of unsatisfiable classes to become 0. For the correspondent negative losses (see Eq. 5 and Eq. 6) we use the interpretation for satisfiable classes as balls with non-zero radius, i.e., with a radius which equals to or greater than some small positive number ε:

$$loss_{C \not\sqsubseteq \bot}(c) = \max(0, \varepsilon - r_\eta(c)) \quad (5)$$

$$loss_{\exists R.C \not\sqsubseteq \bot}(r,c) = \max(0, \varepsilon - r_\eta(c)) \quad (6)$$

4.2. ELBE Negative Losses

ELBE is a model that relies on boxes instead of balls. Here, similarly, ε is a small positive number, $e_c(c)$, $e_c(d)$ and $e_o(c)$, $e_o(d)$ denote the box center and the box offset associated with classes c, d, respectively, $e_c(r)$ denotes the embedding vector associated with relation r, and $e_c(new)$, $e_o(new)$ correspond to the center and the offset of the box which is the result of intersection of boxes associated with concepts c and d, *margin* stands for a margin parameter.

Following the same method of negative loss construction for *ELEmbeddings*, we use GCI1-BOT loss as a negative loss for $C \sqsubseteq D$ axioms (see Eq. 7):

$$loss_{C \not\sqsubseteq D}(c,d) = \| \max(zeros, -|e_c(c) - e_c(d)| + e_o(c) + e_o(d) + margin)\| \tag{7}$$

Since axis-aligned hyperrectangles are closed under intersection, we also use GCI1-BOT for the intersection of boxes representing C and D concepts and the E box(see Eq. 8):

$$loss_{C \sqcap D \not\sqsubseteq E}(c,d,e) = \| \max(zeros, -|e_c(new) - e_c(e)| + e_o(new) + e_o(e) + margin)\| \tag{8}$$

This property also allows us to interpret each negative sample for $C \sqcap D \sqsubseteq \bot$ axioms as a box intersection with nonzero offset (see Eq. 9):

$$loss_{C \sqcap D \not\sqsubseteq \bot}(c,d) = \max(0, \varepsilon - \|e_o(new)\|) \tag{9}$$

Other negative losses have the form similar to the ones constructed for *ELEmbeddings*:

$$loss_{\exists R.C \not\sqsubseteq D}(r,c,d) = \|max(zeros, -|e_c(c) - e_c(r) - e_c(d))| + e_o(c) + e_o(d) + margin)\| \tag{10}$$

$$loss_{C \not\sqsubseteq \bot}(c) = \max(0, \varepsilon - \|e_o(c)\|) \tag{11}$$

$$loss_{\exists R.C \not\sqsubseteq \bot}(r,c) = \max(0, \varepsilon - \|e_o(c)\|) \tag{12}$$

4.3. *Box*^{2}*EL* Negative Losses

Box^2*EL* is also based on boxes but uses a different relation model compared to ELBE. Additionally making use of the notations from *Box*^{2}*EL* [15], ε is a small positive number, $Box(C)$, $Box(D)$, $Box(E)$ are boxes associated with classes c, d, e, respectively, γ denotes a margin parameter, δ is a parameter from the GCI2 negative loss, $Head(r)$ represents the head box of relation r interpretation, and $Bump(C)$ corresponds to a bump vector associated with concept C.

Equations 13 and 14 are constructed in a similar fashion as for *ELBE* based on the GCI1-BOT loss which penalizes the element-wise distance $\boldsymbol{d}$ between axis-aligned boxes:

$$loss_{C \not\sqsubseteq D}(c,d) = \| \max(\mathbf{0}, -(\boldsymbol{d}(Box(C), Box(D)) + \gamma))\| \tag{13}$$

$$loss_{C \sqcap D \not\sqsubseteq E}(c,d,e) = \| \max(\mathbf{0}, -(\boldsymbol{d}(Box(C) \cap Box(D), Box(E)) + \gamma))\| \tag{14}$$

Negative losses 15–17 encourage boxes to be non-empty:

$$loss_{C \not\sqsubseteq \bot}(c) = \max(0, \varepsilon - \|o(C)\|) \tag{15}$$

$$loss_{C \sqcap D \not\sqsubseteq \bot}(c,d) = \max(0, \varepsilon - \|o(Box(C) \cap Box(D))\|) \tag{16}$$

$$loss_{\exists R.C \not\sqsubseteq \bot}(r,c) = \max(0, \varepsilon - \|o(C)\|) \tag{17}$$

The GCI3 negative loss reflects the structure of the original GCI3 loss:

$$loss_{\exists R.C \not\sqsubseteq D}(r,c,d) = (\delta - \mu(Head(r) - Bump(C), Box(D)))^2 \tag{18}$$

ALGORITHM 1

An algorithm for computation of axioms in the deductive closure using inference rules; axioms in bold correspond to subclass/superclass axioms derived using ELK reasoner (here we use the transitive closure of the ELK inferences); plain axioms come from the knowledge base. The input of the algorithm is the set of normalized axioms of the knowledge base, the output is the set of entailed axioms computed with respect to all normal forms according to syntactic rules stated in the algorithm.

Input: An $\mathcal{EL}^{++}$ theory T
Output: An extended theory T'

for all $C \sqcap D \sqsubseteq E$ in the knowledge base **do**

$$\frac{C \sqcap D \sqsubseteq E \quad \boldsymbol{C' \sqsubseteq C} \quad \boldsymbol{D' \sqsubseteq D} \quad \boldsymbol{E \sqsubseteq E'}}{C' \sqcap D' \sqsubseteq E'}$$

end for
for all $C \sqsubseteq \exists R.D$ in the knowledge base **do**

$$\frac{C \sqsubseteq \exists R.D \quad \boldsymbol{C' \sqsubseteq C} \quad \boldsymbol{D \sqsubseteq D'} \quad R \sqsubseteq R'}{C' \sqsubseteq \exists R'.D'} \qquad \frac{C \sqsubseteq \exists R.D \quad D \sqsubseteq \exists R'.E \quad R \circ R' \sqsubseteq S}{C \sqsubseteq \exists S.E}$$

end for
for all $\exists R.C \sqsubseteq D$ in the knowledge base **do**

$$\frac{\exists R.C \sqsubseteq D \quad \boldsymbol{C' \sqsubseteq C} \quad \boldsymbol{D \sqsubseteq D'} \quad R' \sqsubseteq R}{\exists R'.C' \sqsubseteq D'}$$

end for
for all $C \sqcap D \sqsubseteq \bot$ in the knowledge base **do**

$$\frac{C \sqcap D \sqsubseteq \bot \quad \boldsymbol{C' \sqsubseteq C} \quad \boldsymbol{D' \sqsubseteq D}}{C' \sqcap D' \sqsubseteq \bot} \qquad \frac{C \sqcap D \sqsubseteq \bot}{C \sqcap D \sqsubseteq E}$$

end for
for all $\exists R.C \sqsubseteq \bot$ in the knowledge base **do**

$$\frac{\exists R.C \sqsubseteq \bot \quad \boldsymbol{C' \sqsubseteq C} \quad R' \sqsubseteq R}{\exists R'.C' \sqsubseteq \bot}$$

end for

5. Negative Sampling and Entailments

In the case of knowledge base completion where the deductive closure contains potentially many non-trivial entailed axioms, the random sampling approach for negatives may lead to suboptimal learning since some of the axioms treated as negatives may be entailed (and should therefore be true in any model, in particular the one constructed by the geometric embedding method). As an example, let us consider the simple ontology consisting of two axioms: $A \sqcap B \sqsubseteq C$ and $D \sqsubseteq B$. For the $A \sqcap B \sqsubseteq C$ axiom, random negative sampling will sample $A \sqcap B \sqsubseteq C'$ where

C' is one of A, B, C, D. Since the knowledge base makes the axioms $A \sqcap B \sqsubseteq A$, $A \sqcap B \sqsubseteq B$, and $A \sqcap B \sqsubseteq C$ true, in 75% of cases we will sample a negative for this axiom that is actually true in each model.

ALGORITHM 2
Additional entailed axioms. The input of the algorithm is the set of concept and/or role names, the output is the set of entailed axioms computed according to syntactic rules stated in the algorithm.

Input: An $\mathcal{EL}^{++}$ theory T
Output: An extended theory T'

for all concepts C, D, E, E' in the signature **do**

$$\frac{}{C \sqcap \bot \sqsubseteq E} \quad \frac{D \sqsubseteq \bot}{C \sqcap D \sqsubseteq E} \quad \frac{E \sqsubseteq E'}{C \sqcap E \sqsubseteq E'} \quad \frac{C \sqcap D \sqsubseteq \bot}{C \sqcap D \sqsubseteq E}$$

$$\frac{C \sqsubseteq E \quad D \sqsubseteq E \quad C' \sqsubseteq C \quad D' \sqsubseteq D \quad E \sqsubseteq E'}{C' \sqcap D' \sqsubseteq E'} \quad \frac{C \sqsubseteq C'}{C \sqcap \top \sqsubseteq C'}$$

end for
for all relations R and all concepts $D \neq \bot$ in the signature **do**

$$\frac{}{\bot \sqsubseteq \exists R.D} \quad \frac{C \sqsubseteq \bot}{C \sqsubseteq \exists R.D}$$

end for
for all relations R and all concepts $C \neq \bot$ in the signature **do**

$$\frac{}{\exists R.C \sqsubseteq \top}$$

end for

We suggest to filter selected negatives during training based on the deductive closure of the knowledge base: for each randomly generated axiom to be used as negative, we check whether it is present in the deductive closure and, if it is, we delete it. $\mathcal{EL}^{++}$reasoners such as ELK [19] compute subsumption hierarchies, i.e., all axioms of the form $C \sqsubseteq D$ in the deductive closure, but not entailed axioms for the other normal forms. We use the inferences computed by ELK (of the form $C \sqsubseteq D$ where C and D are concept names) to design an algorithm that computes (a part of) the deductive closure with respect to the $\mathcal{EL}^{++}$normal forms. The algorithm implements a sound but incomplete set of inference rules which can quickly generate a partial deductive closure with respect to all normal forms. Algorithm 1 contains inference rules for deriving entailed axioms of type GCI1, GCI2, GCI3, GCI1-BOT and GCI3-BOT from axioms explicitly represented within a knowledge base; GCI0 and GCI0-BOT axioms are precomputed by ELK. Algorithm 2 provides a set of additional rules depending on arbitrary classes and relations represented within a knowledge base after inferred axioms from Algorithm 1 are computed. The purpose of Algorithm 2 is to enrich the approximate deductive closure with axioms involving arbitrary relations and concepts or with axioms of new GCI type which may be missed by applying rules from Algorithm 1 since Algorithm 1 computes entailed axioms based on ontology axioms, concept hierarchy and role inclusions. For example, consider an ontology consisting of two axioms: $C \sqsubseteq E$ and $D \sqsubseteq E$. Since no GCI1 axioms are present in the ontology, no GCI1 axioms will be entailed by Algorithm 1. Algorithm 2 enables inference of a GCI1 axiom $C \sqcap D \sqsubseteq E$. Another example is an ontology comprised of axioms $C \sqsubseteq D$ and $D \sqcap E \sqsubseteq F$. The axiom $C \sqcap \bot \sqsubseteq D$, which is entailed, cannot be inferred by applying rules from Algorithm 1: the concept C is not a subclass of neither D or E. Although we can use ELK or similar reasoners to query for arbitrary entailed axioms, the algorithms we propose have an advantage over this method since it does not require the addition of a new concept to an ontology and recomputing the concept hierarchy. We show a detailed example of the algorithm in Appendix E based on the simple ontology example introduced in Section 6.4.

In the task of knowledge base completion with many non-trivial entailed axioms, the deductive closure can also be used to modify the evaluation metrics, or define novel evaluation metrics that distinguish between entailed and non-entailed axioms. So far, ontology embedding methods that have been applied to the task of knowledge base completion have used evaluation measures that are taken from the task of knowledge graph completion; in particular, they only evaluate knowledge base completion using axioms that are "novel" and not entailed. However, any entailed axiom will be true in all models of the knowledge base, and therefore also in the geometric model that is constructed by the embedding method.

We suggest to filter entailed axioms from training or test sets when the aim is to predict "novel" (i.e., non-entailed) knowledge. The geometric embedding methods generate models making all entailed axioms true in all models. It is expected that methods explicitly constructing models preferentially make entailed axioms true and rank them higher than non-entailed axioms. If the evaluation is based solely on non-entailed axioms, it will consider all similar inferred axioms false, and to avoid this, we may filter such axioms from the ranking list. The more axioms are filtered, the more entailed axioms are predicted by a model.

6. Experiments

6.1. Datasets

6.1.1. Gene Ontology & STRING Data

Following previous works [15, 21, 36] we use common benchmarks for knowledge-base completion, in particular a task that predicts protein–protein interactions (PPIs) based on the functions of proteins. We also use the same data for the task of protein function prediction. For these tasks we use two datasets, each of them consists of the Gene Ontology (GO) [50] with all its axioms, protein–protein interactions (PPIs) and protein function axioms extracted from the STRING database [28]; each dataset focuses on only yeast proteins. GO is formalized using OWL 2 EL [14].

For the PPI yeast network we use the built-in dataset `PPIYeastDataset` available in the mOWL [49] Python library (release 0.2.1) where axioms of interest are split randomly into train, validation and test datasets in ratio 90:5:5 keeping pairs of symmetric PPI axioms within the same dataset, and other axioms are placed into the training part; validation and test sets are made up of TBox axioms of type $\{P_1\} \sqsubseteq \exists interacts\_with.\{P_2\}$ where P_1, P_2 are protein names. The GO version released on 2021-10-20 and the STRING database version 11.5 were used. Alongside with the yeast *interacts_with* dataset we collected the yeast *has_function* dataset organized in the same manner with validation and test parts containing TBox axioms of type $\{P\} \sqsubseteq \exists has\_function.\{GO\}$. Based on the information in the STRING database, in PPI yeast, the *interacts_with* relation is symmetric and the dataset is closed against symmetric interactions. We normalize each train ontology using the updated implementation of the jcel [27] reasoner [1] where we take into consideration newly generated concept and role names. Although role inclusion axioms may be utilized within the Box^2EL framework we ignore them since neither *ELEmbeddings* nor *ELBE* incorporate these types of axioms. Table in the appendix A shows the number of GCIs of each type in the datasets and the number of concepts and roles after normalization. For more precise evaluation of novel knowledge prediction we remove entailed axioms from the test set for function prediction task based on the precomputed deductive closure of the train ontology (see Section 5).

[1] https://github.com/julianmendez/jcel/pull/12

6.1.2. Food ontology & GALEN Ontology

Food Ontology [11] contains structured information about foods formalized in $\mathcal{SRIQ}$ DL expressivity [7] involving terms from UBERON [33], NCBITaxon [13], Plant Ontology [16], and others. The GALEN ontology [39] represents biomedical concepts related to anatomy, diseases, and others [38]. For the Food Ontology, the data for subsumption prediction was extracted from the case studies used to evaluate OWL2Vec* [7][2]; the train part of the ontology was restricted to the $\mathcal{EL}$ fragment and normalized using the jcel [27] reasoner. In case of GALEN ontology, subsumption axioms were randomly split in ratio 90:5:5 among train, validation and test sets. Since the normalization procedure splits each complex axiom into a set of shorter axioms including subsumptions between atomic concepts from the signature, it may result in adding axioms represented in the validation or test part of the ontology to the train part. To avoid this, we filtered out such axioms from the original validation and test datasets after the train ontology for subsumption prediction was normalized. Additionally, as described in Section 6.1.1, we remove entailed axioms from the test dataset. Statistics about the number of axioms of each GCI type, relations and classes can be found in Appendix B for the Food Ontology and in Appendix C for the GALEN ontology.

6.2. Evaluation Scores and Metrics

For GO & STRING data, we predict GCI2 axioms of type $\{P_1\} \sqsubseteq \exists interacts\_with.\{P_2\}$ or $\{P\} \sqsubseteq \exists has\_function.\{GO\}$ depending on the dataset. On Food Ontology and GALEN ontology, we predict GCI0 axioms of type $C \sqsubseteq D$, C and D are arbitrary classes from the signature. For each axiom type, we use the corresponding loss expressions to score axioms. This is justified by the fact that objective functions are measures of truth for each axiom within constructed models.

The predictive performance is measured by the Hits@n metrics for $n = 1, 10, 100$, macro and micro mean rank, and the area under the ROC curve (AUC ROC). For rank-based metrics, we calculate the score of $C \sqsubseteq \exists R.D$ or $C \sqsubseteq D$ for every class C from the test set and for every D from the set $\mathbf{C}$ of all classes (or subclasses of a certain type, such as proteins or functions for domain-specific cases) and determine the rank of a test axiom $C \sqsubseteq \exists R.D$. For macro mean rank and AUC ROC, we consider all axioms from the test set; for micro metrics, we compute corresponding class-specific metrics averaging them over all classes in the signature:

$$micro\_MR_{C\sqsubseteq\exists R.D} = Mean(MR_C(\{C \sqsubseteq \exists R.D,\ D \in \mathbf{C}\})) \tag{19}$$

$$micro\_MR_{C\sqsubseteq D} = Mean(MR_C(\{C \sqsubseteq D,\ D \in \mathbf{C}\})) \tag{20}$$

$$micro\_AUC\_ROC_{C\sqsubseteq\exists R.D} = Mean(AUC\_ROC_C(\{C \sqsubseteq \exists R.D,\ D \in \mathbf{C}\})) \tag{21}$$

$$micro\_AUC\_ROC_{C\sqsubseteq D} = Mean(AUC\_ROC_C(\{C \sqsubseteq D,\ D \in \mathbf{C}\})) \tag{22}$$

Additionally, we remove axioms represented in the train set or deductive closures (see Section 5) to obtain corresponding filtered metrics (FHits@n, FMR, FAUC). In related work focusing on knowledge graph completion or knowledge base completion tasks [4, 21, 36, 45], filtered metrics are computed by removing axioms presented within the train set from the list of all ranked axioms. This filtration is applied to eliminate statements learnt by a model during training phase which are therefore likely to have lower rank and to evaluate the predictive performance of a model in a more fair setting.

6.3. Training Procedure

All models are optimized with respect to the sum of individual GCI losses (here we define the loss in most general case using all positive and all negative losses):

$$\begin{aligned}\mathcal{L} = {} & l_{C\sqsubseteq D} + l_{C\sqcap D\sqsubseteq E} + l_{C\sqsubseteq\exists R.D} + l_{\exists R.C\sqsubseteq D} + l_{C\sqsubseteq\bot} + l_{C\sqcap D\sqsubseteq\bot} + l_{\exists R.C\sqsubseteq\bot} + \\ & + l_{C\not\sqsubseteq D} + l_{C\sqcap D\not\sqsubseteq E} + l_{C\not\sqsubseteq\exists R.D} + l_{\exists R.C\not\sqsubseteq D} + l_{C\not\sqsubseteq\bot} + l_{C\sqcap D\not\sqsubseteq\bot} + l_{\exists R.C\not\sqsubseteq\bot}\end{aligned} \tag{23}$$

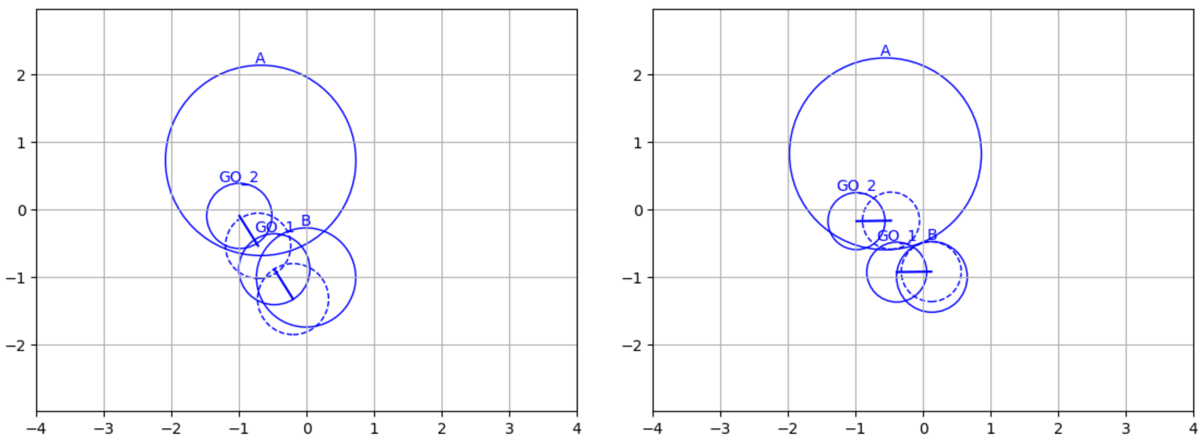
[2]https://github.com/KRR-Oxford/OWL2Vec-Star/tree/master/case_studies

All model architectures are built using the mOWL [49] library on top of mOWL's base models. All models were trained using the same fixed random seed.

All models are trained for 2,000 epochs for STRING & GO datasets and 800 epochs for the Food Ontology and GALEN datasets with batch size of 32,768. Training and optimization is performed using Pytorch with Adam optimizer [20] and ReduceLROnPlateau scheduler with patience parameter 10. We apply early stopping if validation loss does not improve for 20 epochs. For *ELEmbeddings*, hyperparameters are tuned using grid search over the following set: margin $\gamma \in \{-0.1, -0.01, 0, 0.01, 0.1\}$, embedding dimension $\{50, 100, 200, 400\}$, learning rate $\{0.01, 0.001, 0.0001\}$; since none of our datasets contains unsatisfiable classes, we do not tune the parameter ε appearing in GCI0-BOT and GCI3-BOT negative losses. For *ELBE*, grid search is performed over 60 randomly chosen subsets of the following hyperparameters: embedding dimension $\{25, 50, 100, 200\}$, margin $\{-0.1, -0.01, 0, 0.01, 0.1\}$, $\varepsilon \in \{0.1, 0.01, 0.001\}$ (for experiments with all negative losses involved), learning rate $\{0.01, 0.001, 0.0001\}$. The same strategy is applied to Box^2EL models for embedding dimension $\{25, 50, 100, 200\}$, margin $\gamma \in \{-0.1, -0.01, 0, 0.01, 0.1\}$, $\delta \in \{1, 2, 4\}$, $\varepsilon \in \{0.1, 0.01, 0.001\}$ (similarly, for experiments with all negative losses involved), regularization factor $\lambda \in \{0, 0.05, 0.1, 0.2\}$, and learning rate $\{0.01, 0.001, 0.0001\}$. For experiments with negatives filtration during training we use the same set of hyperparameters for random and filtered mode of negative sampling. See Appendix D for details on optimal hyperparameters used.

6.4. Results

We evaluate whether adding negative losses for all normal forms will allow for the construction of a better model and improve the performance in the task of knowledge base completion. We test the effect of the expanded negative sampling and negative losses first on a small ontology that can be embedded and visualized in 2D space, and then on a larger application. We formulate and add negative losses for all normal forms given by equations 1–17.



(a) GCI2 negative loss (b) All negative losses

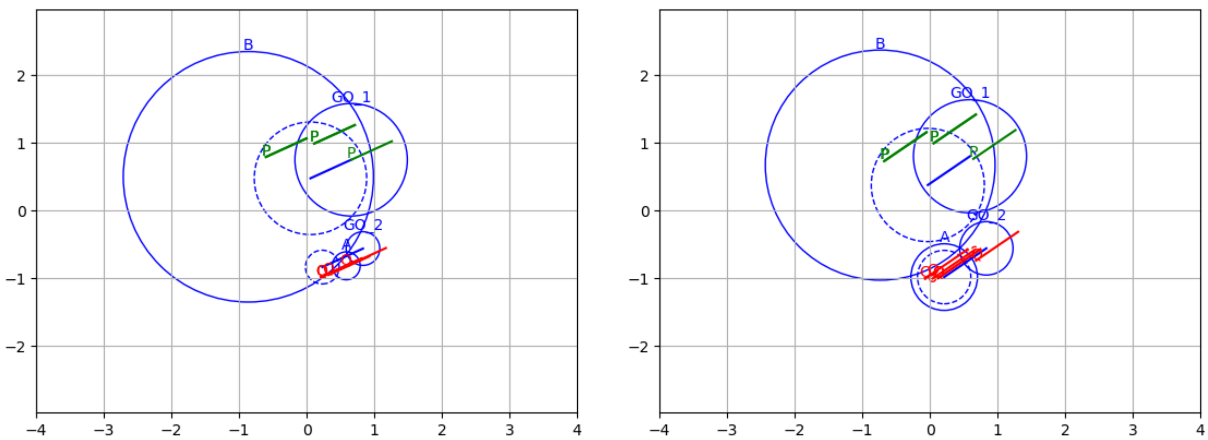
Fig. 1. *ELEmbeddings* example. Dashed circles represent translated classes by relational vector corresponding to *has_function* relation. The normalized theory $\{\{GO_1\} \sqcap \{GO_2\} \sqsubseteq \bot, A \sqcap B \sqsubseteq \bot, \exists has\_function.\{GO_1\} \sqsubseteq B, \exists has\_function.\{GO_2\} \sqsubseteq A\}$ is better preserved when negative losses are incorporated to all normal forms (Figure b) rather than only to GCI2 normal form (Figure a).

First, we investigate a simple example corresponding to the task of protein function prediction using the *ELEmbeddings* model. Let us consider an ontology consisting of two axioms stating that there are two disjoint functions $\{GO_1\}$ and $\{GO_2\}$, and proteins having these functions are also disjoint: $\{GO_1\} \sqcap \{GO_2\} \sqsubseteq \bot$, $\exists has\_function.\{GO_1\} \sqcap \exists has\_function.\{GO_2\} \sqsubseteq \bot$. After normalization, the last axiom is substituted by the following three axioms: $A \sqcap B \sqsubseteq \bot$, $\exists has\_function.\{GO_1\} \sqsubseteq B$, $\exists has\_function.\{GO_2\} \sqsubseteq A$ where A, B are new concept names. To visualize the results, we embed these axioms in 2D space. Figure 1(a) shows the embedding generated with the original *ELEmbeddings* model. Since there are no axioms of type GCI2 represented within the knowledge base, the model learns without any negative examples and demonstrates poor performance compared to the model with incorporated negative losses for all normal forms as demonstrated in Figure 1(b).

Since we are interested in predicting not only axioms of type $C \sqsubseteq \exists R.D$ for which negative sampling is used in the original *ELEmbeddings*, *ELBE* and *Box*^{2}*EL*, we also examine the effect of all negative losses utilization during training on Food Ontology and GALEN ontology for subsumption prediction (see Tables 5 and 6, respectively). We find that the *ELEmbeddings* model does not improve on the Food Ontology subsumption prediction task, but *ELBE* with additional losses improves over the original model; *Box*^{2}*EL* with additional losses surpasses its version with just GCI2 negative loss in Hits@n metrics. As for the results on GALEN ontology, we find that in case of all three models Hits@n metrics are improved when all negative losses are applied (except Hits@100 metric for *ELEmbeddings* model) indicating that in this particular case negative losses encourage models to predict more new axioms. Mean rank results are similarly better for *ELEmbeddings* and *Box*^{2}*EL* models.

Additionally, we evaluate the performance on a standard benchmark set for protein–protein interaction (PPI) prediction (see Table 4). For this task, the test axioms are of the type GCI2. We observe that *ELEmbeddings* and *ELBE* with negative losses for all normal forms integrated demonstrate superior performance compared to their initial configurations in terms of Hits@n metrics; it also allows *Box*^{2}*EL* to lower ranks of test axioms. Generally, for the task of PPI prediction, additional negative sampling improves performance.

To summarize the above mentioned observations, we note that in some cases additional negative losses may dicrease the ability of models to predict new axioms and encourage models to predict entailed knowledge first (as, e.g., in protein function prediction case) thus leading to construction of a more accurate model of a theory. Since there is a tradeoff between prediction of novel and entailed knowledge, additional negative losses may demonstrate worse performance on novel knowledge prediction.



(a) With random negatives

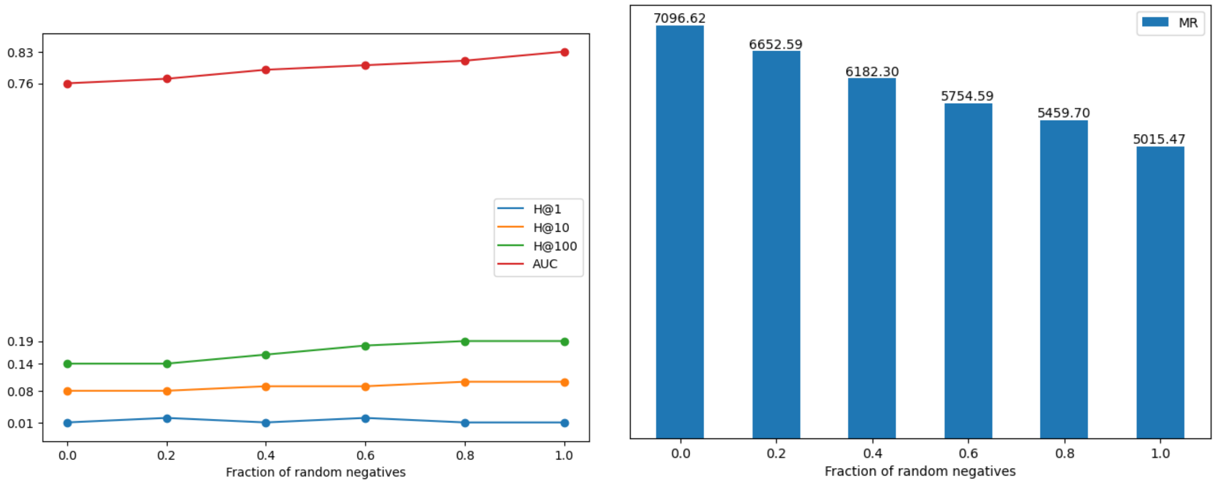
(b) With filtered negatives

Fig. 2. *ELEmbeddings* example. Dashed circles represent translated classes by relational vector corresponding to *has_function* relation. 'Red' classes represent proteins $\{Q_1\}, \dots, \{Q_5\}$, 'green' classes represent proteins $\{P_1\}, \dots, \{P_5\}$. Axioms $\{Q_i\} \sqsubseteq \exists has\_function.\{GO_2\}, i = 1, \dots, 5$ are better preserved when negatives are filtered based on precomputed deductive closure (Figure b) rather than when random negatives are sampled (Figure a). The same applies for the axiom $\exists has\_function.\{GO_2\} \sqsubseteq A$.

Using the example introduced above and the *ELEmbeddings* embedding model, we demonstrate that negatives filtration may be beneficial for constructing a model of a theory. Apart from axioms mentioned earlier, i.e., $\{GO_1\} \sqcap \{GO_2\} \sqsubseteq \bot$, $A \sqcap B \sqsubseteq \bot$, $\exists has\_function.\{GO_1\} \sqsubseteq B$ and $\exists has\_function.\{GO_2\} \sqsubseteq A$, we add 10 more axioms about 5 proteins $\{P_1\}, \dots, \{P_5\}$ having function $\{GO_1\}$ (i.e., $\{P_i\} \sqsubseteq \exists has\_function.\{GO_1\},\ i = 1, \dots, 5$), and 5 proteins $\{Q_1\}, \dots, \{Q_5\}$ having function $\{GO_2\}$ (i.e., $\{Q_i\} \sqsubseteq \exists has\_function.\{GO_2\},\ i = 1, \dots, 5$). Figure 2 shows the constructed models with and without negatives filtering. We observe that the model with filtered negatives provides faithful representation of GCI3 axiom $\exists has\_function.\{GO_2\} \sqsubseteq A$ and axioms introducing proteins having function $\{GO_2\}$ as opposed to its counterpart with random negatives: according to geometric interpretation, for

GCI3 axioms $\exists R.C \sqsubseteq D$ to be faithfully represented, the n-ball interpreting the concept C translated by $-r$ relation R vector should lie inside the n-ball interpreting the concept D. We see that this hold true on Figure 2(b) yet not on Figure 2(a).

Tables 4–6 show results in the tasks of protein–protein interaction and subsumption prediction. We find that excluding axioms in the deductive closure for negative selection slightly improves or yields similar results. One possible reason is that a randomly chosen axiom is very unlikely to be entailed since very few axioms are entailed compared to all possible axioms to choose from.



(a) H@1, H@10, H@100 and ROC AUC (b) macro_MR

Fig. 3. Metrics reported for biased fraction of random negatives combined with entailed axioms from the precomputed deductive closure.

Because the chance of selecting an entailed axiom as a negative depends on the knowledge base on which the embedding method is applied, we perform additional experiments on Food Ontology with *ELEmbeddings* model where we bias the selection of negatives; we chose between 100% negatives to 0% negatives from the entailed axioms. We find that reducing the number of entailed axioms from the negatives has an effect to improve performance and the effect increases the more axioms would be chosen from the entailed ones (see Figure 3).

We compute filtered metrics for the protein function and subsumption prediction tasks. Both of them account for entailed axioms prediction since if, e.g., $C \sqsubseteq D$ is being predicted then first models may predict axioms of type $C \sqsubseteq D'$ where D' is any superclass of D; the same is true for function prediction axioms $\{P\} \sqsubseteq \exists has\_function.\{GO\}$ and all superclasses $\{GO'\}$ of $\{GO\}$ class. Note that the protein–protein interaction prediction task is not tailored for evaluation using deductive closures of the train or test set: for each protein $\{P\}$ its subclasses include only $\bot$ and superclasses include only $\top$. As a result, the only inferred axioms will be of type $\bot \sqsubseteq \exists interacts\_with.\{P\}$, $\{P_1\} \sqsubseteq \exists interacts\_with.\{P_2\}$ or $\{P\} \sqsubseteq \exists interacts\_with.\top$, and filtered metrics may be computed only with respect to the train part of the ontology. For this reason we do not report filtered metrics for protein–protein interaction prediction task (Table 4).

For function prediction and subsumption prediction, we employ filtration of metrics based on the deductive closure of the train set and of the test set. Tables 5, 6 and 3 contain results for subsumption prediction on Food Ontology, subsumption prediction on GALEN ontology and function prediction on GO, respectively.

Our findings suggest that the baseline *ELEmbeddings* predicts primarily entailed axioms of GCI2 type, yet for GCI0 on Food Ontology the model predicts "novel" knowledge first whereas the model modifications with additional negative losses and negatives filtration derive entailed knowledge in the first place. For the GALEN ontology, however, the situation is similar to the protein function prediction case, i.e., novel knowledge is predicted in the first place for modifications with additional negative losses and negatives filtration. This may indicate model construction where many classes overlap or "collapse" for all negative losses and negatives filtering case since the GALEN ontology does not contain disjointness axioms and consequently no classes will be separated by the model. The same holds for *ELBE* and *Box*^{2}*EL* models. Losses for all normal forms and negatives filtering during training aid *ELBE* and *Box*^{2}*EL* to construct model-generated embeddings which first predict logically inferred knowledge and then non-entailed axioms of type GCI2 or GCI0 (on Food Ontology), respectively. The results indicate that models with all types of valid negatives in most cases explicitly construct models.

7. Discussion

We evaluated properties of *ELEmbeddings*, *ELBE* and *Box*^{2}*EL*, ontology embedding methods that aims to generate a model of an $\mathcal{EL}^{++}$theory; the properties we evaluate hold similarly for other ontology embedding methods that construct models of $\mathcal{EL}^{++}$theories. While we demonstrate several improvements over the original model, we can also draw some general conclusions about ontology embedding methods and their evaluation. Knowledge base completion is the task of predicting axioms that should be added to a knowledge base; this task is adapted from knowledge graph completion where triples are added to a knowledge graph. The way both tasks are evaluated is by removing some statements (axioms or triples) from the knowledge base, and evaluating whether these axioms or triples can be recovered by the embedding method. This evaluation approach is adequate for knowledge graphs which do not give rise to many entailments. However, knowledge bases give rise to potentially many non-trivial entailments that need to be considered in the evaluation. In particular, embedding methods that aim to generate a model of a knowledge base will first generate entailed axioms (because entailed axioms are true in all models); these methods perform knowledge base completion as a generalization of generating the model where either other statements may be true, or they may be approximately true in the generated structure. This has two consequences: the evaluation procedure needs to account for this; and the model needs to be sufficiently rich to allow useful predictions.

We have introduced a method to compute the deductive closure of $\mathcal{EL}^{++}$knowledge bases; this method relies on an automated reasoner and is sound. We use all the axioms in the deductive closure as positive axioms to be predicted when evaluating knowledge base completion, to account for methods that treat knowledge base completion as a generalization of constructing a model and testing for truth in this model. We find that some models (e.g., modified box-based models using valid negatives of all types) can predict entailed axioms well, some (e.g., the original *Box*^{2}*EL* model) preferentially predict "novel", non-entailed axioms; these methods solve subtly different problems (either generalizing construction of a model, or specifically predicting novel non-entailed axioms). We also modify the evaluation procedure to account for the inclusion of entailed axioms as positives; however, the evaluation measures are still based on ranking individual axioms and do not account for semantic similarity. For example, if during testing, the correct axiom to predict is $C \sqsubseteq \exists R.D$ but the predicted axiom is $C \sqsubseteq \exists R.E$, the prediction may be considered to be "more correct" if $D \sqsubseteq E$ was in the knowledge base than if $D \sqcap E \sqsubseteq \bot$ was in the knowledge base. Novel evaluation metrics need to be designed to account for this phenomenon, similarly to ontology-based evaluation measures used in life sciences [37]. It is also important to expand the set of benchmark sets for knowledge base completion.

Use of the deductive closure is not only useful in evaluation but also when selecting negatives. In formal knowledge bases, there are at least two ways in which negatives for axioms can be chosen: they are either non-entailed axioms, or they are axioms whose negation is entailed. However, in no case should entailed axioms be considered as negatives; we demonstrate that filtering entailed axioms from selected negatives during training improves the performance of the embedding method consistently in knowledge base completion (and, obviously, more so when entailed axioms are considered as positives during evaluation).

While we only report our experiments with *ELEmbeddings*, *ELBE* and *Box*^{2}*EL*, our findings, in particular about the evaluation and use of deductive closure, are applicable to other geometric ontology embedding methods. As ontology embedding methods are increasingly applied in knowledge-enhanced learning and other tasks that utilize some form of approximate computation of entailments, our results can also serve to improve the applications of ontology embeddings.

Table 3

Protein function prediction experiments on yeast proteins. ‘l’ corresponds to all negative losses, ‘l+n’ means a model was trained using all negative losses and negatives filtering. For each model we report non-filtered metrics (NF) and filtered metrics with respect to the deductive closure of the train and the test set combined together (F). For macro_MR and micro_MR we additionally report the difference between filtered and non-filtered metrics (NF-F) to check how much of entailed knowledge is predicted on average. Values in **bold** indicate best metrics.

Model	H@10		H@100		macro_MR			micro_MR			macro_AUC		micro_AUC	
	NF	F	NF	F	NF	F	NF-F	NF	F	NF-F	NF	F	NF	F
ELEm	**0.01**	**0.01**	**0.03**	**0.03**	21198	21150	**48**	21165	21118	**47**	0.62	0.62	0.63	0.63
ELEm+l	0.00	0.00	**0.03**	**0.03**	9603	9575	28	9449	9423	26	**0.83**	**0.83**	**0.84**	**0.84**
ELEm+l+n	0.00	0.00	**0.03**	**0.03**	**9488**	**9460**	28	**9334**	**9307**	27	**0.83**	**0.83**	**0.84**	**0.84**
ELBE	**0.03**	**0.03**	**0.24**	**0.24**	**4229**	**4209**	20	**4156**	**4137**	19	**0.92**	**0.92**	**0.93**	**0.93**
ELBE+l	0.00	0.00	0.01	0.01	12920	12865	**55**	12797	12745	52	0.77	0.77	0.78	0.78
ELBE+l+n	0.00	0.00	0.01	0.01	12900	12845	**55**	12772	12719	**53**	0.77	0.77	0.78	0.78
Box^2EL	**0.28**	**0.31**	**0.55**	**0.55**	**1988**	**1979**	**9**	**1988**	**1980**	**8**	**0.96**	**0.96**	**0.97**	**0.97**
Box^2EL+l	0.24	0.27	0.54	**0.55**	2129	2120	**9**	2099	2091	**8**	**0.96**	**0.96**	**0.97**	**0.97**
Box^2EL+l+n	0.24	0.27	0.54	**0.55**	2161	2152	**9**	2147	2139	**8**	**0.96**	**0.96**	0.96	0.96

Table 4

Protein–protein interaction prediction experiments on yeast proteins. ‘l’ corresponds to all negative losses, ‘l+n’ means a model was trained using all negative losses and negatives filtering. Non-filtered metrics are reported. Values in **bold** indicate best metrics.

Model	H@10	H@100	macro_MR	micro_MR	macro_AUC	micro_AUC
ELEm	0.05	0.31	599.21	701.57	0.90	0.90
ELEm+l	**0.06**	0.35	532.93	681.02	**0.91**	0.90
ELEm+l+n	**0.06**	**0.37**	**519.62**	**671.19**	**0.91**	**0.91**
ELBE	0.07	0.37	**829.86**	**1123.47**	**0.91**	**0.89**
ELBE+l	**0.08**	**0.40**	984.92	1259.54	0.84	0.82
ELBE+l+n	**0.08**	**0.40**	984.18	1281.20	0.84	0.82
Box^2EL	**0.05**	0.57	215.07	287.16	0.96	**0.96**
Box^2EL+l	**0.05**	0.57	200.85	**250.17**	**0.97**	**0.96**
Box^2EL+l+n	**0.05**	**0.58**	**197.73**	250.47	**0.97**	**0.96**

Table 5

Subsumption prediction experiments on Food Ontology. ‘l’ corresponds to all negative losses, ‘l+n’ means a model was trained using all negative losses and negatives filtering. For each model we report non-filtered metrics (NF) and filtered metrics with respect to the deductive closure of the train and the test set combined together (F). For macro_MR and micro_MR we additionally report the difference between filtered and non-filtered metrics (NF-F) to check how much of entailed knowledge is predicted on average. Values in **bold** indicate best metrics.

Model	H@10		H@100		macro_MR			micro_MR			macro_AUC		micro_AUC	
	NF	F	NF	F	NF	F	NF-F	NF	F	NF-F	NF	F	NF	F
ELEm	**0.12**	**0.12**	**0.21**	**0.21**	**4659**	**4656**	**3**	**4662**	**4659**	**3**	**0.84**	**0.84**	**0.84**	**0.84**
ELEm+l	0.10	0.11	0.19	0.19	5015	5013	2	5020	5017	**3**	0.83	0.83	0.83	0.83
ELEm+l+n	0.10	0.11	0.19	0.19	5022	5019	**3**	5027	5024	**3**	0.83	0.83	0.83	0.83
ELBE	0.01	0.01	0.09	0.09	6695	6692	**3**	6688	6686	2	0.77	0.77	0.77	0.77
ELBE+l	**0.04**	**0.04**	**0.14**	**0.14**	5428	5426	2	5412	5409	**3**	**0.81**	**0.81**	**0.82**	**0.82**
ELBE+l+n	**0.04**	**0.04**	**0.14**	**0.14**	**5427**	**5424**	**3**	**5410**	**5408**	2	**0.81**	**0.81**	**0.82**	**0.82**
Box^2EL	0.01	0.01	0.10	0.10	**3900**	**3898**	2	**3877**	**3874**	**3**	**0.87**	**0.87**	**0.87**	**0.87**
Box^2EL+l	0.04	0.04	0.13	0.13	7550	7547	**3**	7555	7553	2	0.74	0.74	0.74	0.74
Box^2EL+l+n	**0.05**	**0.05**	**0.14**	**0.14**	6865	6862	**3**	6869	6866	**3**	0.76	0.76	0.77	0.77

Acknowledgements

This work has been supported by funding from King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research (OSR) under Award No. URF/1/4355-01-01, URF/1/4675-01-01, URF/1/4697-01-01, URF/1/5041-01-01, and REI/1/5334-01-01. This work was supported by the SDAIA–KAUST Center of Excellence in Data Science and Artificial Intelligence (SDAIA–KAUST AI), by funding from King Abdullah University of Science and Technology (KAUST) – KAUST Center of Excellence for Smart Health (KCSH) under award number 5932, and by funding from King Abdullah University of Science and Technology (KAUST) – KAUST Center of Excellence for Generative AI under award number 5940. We acknowledge support from the KAUST Supercomputing Laboratory.

Table 6

Subsumption prediction experiments on GALEN Ontology. 'l' corresponds to all negative losses, 'l+n' means a model was trained using all negative losses and negatives filtering. For each model we report non-filtered metrics (NF) and filtered metrics with respect to the deductive closure of the train and the test set combined together (F). For macro_MR and micro_MR we additionally report the difference between filtered and non-filtered metrics (NF-F) to check how much of entailed knowledge is predicted on average. Values in **bold** indicate best metrics.

Model	H@10		H@100		macro_MR			micro_MR			macro_AUC		micro_AUC	
	NF	F	NF	F	NF	F	NF-F	NF	F	NF-F	NF	F	NF	F
ELEm	0.15	0.16	**0.35**	**0.35**	9106	9105	1	9106	9105	1	**0.82**	**0.82**	**0.82**	**0.82**
ELEm+l	**0.21**	**0.22**	0.34	0.34	**8977**	**8976**	1	**8977**	**8976**	1	**0.82**	**0.82**	**0.82**	**0.82**
ELEm+l+n	**0.21**	**0.22**	0.33	0.34	9005	9003	**2**	9005	9003	**2**	**0.82**	**0.82**	**0.82**	**0.82**
ELBE	0.08	0.08	0.22	0.22	**11236**	**11234**	2	**11236**	**11234**	2	**0.77**	**0.78**	**0.77**	**0.77**
ELBE+l	**0.13**	**0.13**	**0.34**	**0.34**	11884	11882	2	11884	11882	2	0.76	0.76	0.76	0.76
ELBE+l+n	0.12	0.12	**0.34**	**0.34**	11720	11717	**3**	11720	11717	**3**	**0.77**	0.77	0.76	0.76
Box^2*EL*	0.14	0.14	0.31	0.31	11724	11721	**3**	11724	11721	**3**	**0.77**	**0.77**	0.76	0.76
Box^2*EL*+l	**0.16**	**0.16**	**0.37**	**0.37**	**11371**	**11369**	2	**11371**	**11369**	2	**0.77**	**0.77**	**0.77**	**0.77**
Box^2*EL*+l+n	**0.16**	**0.16**	**0.37**	**0.37**	11378	11376	2	11378	11376	2	**0.77**	**0.77**	**0.77**	**0.77**

References

[1] A. Althagafi, F. Zhapa-Camacho and R. Hoehndorf, Prioritizing genomic variants through neuro-symbolic, knowledge-enhanced learning, *Bioinformatics* (2024), btae301. doi:10.1093/bioinformatics/btae301.

[2] F. Baader, S. Brandt and C. Lutz, Pushing the $\mathcal{EL}$ Envelope, in: *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05*, Morgan-Kaufmann Publishers, Edinburgh, UK, 2005.

[3] F. Baader, D. Calvanese, D. McGuinness, D. Nardi and P.F. Patel-Schneider (eds), *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press, 2003.

[4] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston and O. Yakhnenko, Translating Embeddings for Modeling Multi-relational Data, in: *Advances in Neural Information Processing Systems*, Vol. 26, Curran Associates, Inc., 2013.

[5] Z. Bouraoui, S. Jameel and S. Schockaert, Inductive Reasoning about Ontologies Using Conceptual Spaces, *Proceedings of the AAAI Conference on Artificial Intelligence* **31**(1) (2017). doi:10.1609/aaai.v31i1.11162.

[6] P. Cao, Y. Hao, Y. Chen, K. Liu, J. Xu, H. Li, X. Jiang and J. Zhao, Event Ontology Completion with Hierarchical Structure Evolution Networks, in: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino and K. Bali, eds, Association for Computational Linguistics, Singapore, 2023, pp. 306–320. doi:10.18653/v1/2023.emnlp-main.21. https://aclanthology.org/2023.emnlp-main.21.

[7] J. Chen, P. Hu, E. Jimenez-Ruiz, O.M. Holter, D. Antonyrajah and I. Horrocks, OWL2Vec*: embedding of OWL ontologies, *Machine Learning* (2021). doi:10.1007/s10994-021-05997-6.

[8] J. Chen, A. Althagafi and R. Hoehndorf, Predicting candidate genes from phenotypes, functions and anatomical site of expression, *Bioinformatics* **37**(6) (2020), 853–860. doi:10.1093/bioinformatics/btaa879.

[9] Z. Chen, Y. Wang, B. Zhao, J. Cheng, X. Zhao and Z. Duan, Knowledge graph completion: A review, *Ieee Access* **8** (2020), 192435–192456.

[10] J. Devlin, M. Chang, K. Lee and K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, J. Burstein, C. Doran and T. Solorio, eds, Association for Computational Linguistics, 2019, pp. 4171–4186. doi:10.18653/V1/N19-1423. https://doi.org/10.18653/v1/n19-1423.

[11] D.M. Dooley, E.J. Griffiths, G.S. Gosal, P.L. Buttigieg, R. Hoehndorf, M.C. Lange, L.M. Schriml, F.S. Brinkman and W.W. Hsiao, FoodOn: a harmonized food ontology to increase global food traceability, quality control and data integration, *npj Science of Food* **2**(1) (2018), 23.

[12] C. d'Amato, N. Fanizzi, B. Fazzinga, G. Gottlob and T. Lukasiewicz, Ontology-based semantic search on the Web and its combination with the power of inductive reasoning, *Annals of Mathematics and Artificial Intelligence* **65**(2–3) (2012), 83–121–. doi:10.1007/s10472-012-9309-7.

[13] S. Federhen, The NCBI taxonomy database, *Nucleic acids research* **40**(D1) (2012), D136–D143.

[14] C. Golbreich and I. Horrocks, The OBO to OWL Mapping, GO to OWL 1.1!, in: *Proceedings of the OWLED 2007 Workshop on OWL: Experiences and Directions, Innsbruck, Austria, June 6-7, 2007*, C. Golbreich, A. Kalyanpur and B. Parsia, eds, CEUR Workshop Proceedings, Vol. 258, CEUR-WS.org, 2007. https://ceur-ws.org/Vol-258/paper35.pdf.

[15] M. Jackermeier, J. Chen and I. Horrocks, Dual Box Embeddings for the Description Logic EL++, in: *Proceedings of the ACM Web Conference 2024*, WWW '24, 2024. doi:10.1145/3589334.3645648.

[16] P. Jaiswal, S. Avraham, K. Ilic, E.A. Kellogg, S. McCouch, A. Pujar, L. Reiser, S.Y. Rhee, M.M. Sachs, M. Schaeffer et al., Plant Ontology (PO): a controlled vocabulary of plant structures and growth stages, *Comparative and functional genomics* **6**(7–8) (2005), 388–397.

[17] Q. Ji, G. Qi, Y. Ye, J. Li, S. Li, J. Ren and S. Lu, Ontology Revision based on Pre-trained Language Models, 2023.

[18] X. Jiang, Y. Huang, M. Nickel and V. Tresp, *Combining Information Extraction, Deductive Reasoning and Machine Learning for Relation Prediction*, in: *The Semantic Web: Research and Applications*, Springer Berlin Heidelberg, 2012, pp. 164–178–. ISSN 1611-3349. ISBN 9783642302848. doi:10.1007/978-3-642-30284-8_18.

[19] Y. Kazakov, M. Krötzsch and F. Simančík, The Incredible ELK, *Journal of Automated Reasoning* **53**(1) (2013), 1–61–. doi:10.1007/s10817-013-9296-3.

[20] D.P. Kingma and J. Ba, Adam: A Method for Stochastic Optimization, in: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, eds, 2015.

[21] M. Kulmanov, W. Liu-Wei, Y. Yan and R. Hoehndorf, EL Embeddings: Geometric construction of models for the Description Logic EL ++, in: *International Joint Conference on Artificial Intelligence*, 2019.

[22] V. Lacerda, A. Ozaki and R. Guimarães, Faithel: Strongly tbox faithful knowledge base embeddings for EL, in: *International Joint Conference on Rules and Reasoning*, Springer, 2024, pp. 191–199.

[23] V. Lacerda, A. Ozaki and R. Guimarães, Strong Faithfulness for ELH Ontology Embeddings, *Transactions on Graph Data and Knowledge* **2**(3) (2024), 2–1.

[24] N. Li, T. Bailleux, Z. Bouraoui and S. Schockaert, Ontology Completion with Natural Language Inference and Concept Embeddings: An Analysis, 2024.

[25] O. Mashkova, F. Zhapa-Camacho and R. Hoehndorf, Enhancing geometric ontology embeddings for EL++ with negative sampling and deductive closure filtering, in: *International Conference on Neural-Symbolic Learning and Reasoning*, Springer, 2024, pp. 331–354.

[26] J. Mendez, A Classification Algorithm for ELHIfR+, *Dresden University of Technology* (2011).

[27] J. Mendez, jcel: A Modular Rule-based Reasoner, in: *Proceedings of the 1st International Workshop on OWL Reasoner Evaluation (ORE-2012), Manchester, UK, July 1st, 2012*, I. Horrocks, M. Yatskevich and E. Jiménez-Ruiz, eds, CEUR Workshop Proceedings, Vol. 858, CEUR-WS.org, 2012. https://ceur-ws.org/Vol-858/ore2012_paper12.pdf.

[28] C.v. Mering, STRING: a database of predicted functional associations between proteins, *Nucleic Acids Research* **31**(1) (2003), 258–261–. doi:10.1093/nar/gkg034.

[29] S. Mežnar, M. Bevec, N. Lavrač and B. Škrlj, Ontology Completion with Graph-Based Machine Learning: A Comprehensive Evaluation, *Machine Learning and Knowledge Extraction* **4**(4) (2022), 1107–1123–. doi:10.3390/make4040056.

[30] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient Estimation of Word Representations in Vector Space, in: *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, Y. Bengio and Y. LeCun, eds, 2013.

[31] B. Mohapatra, S. Bhatia, R. Mutharaju and G. Srinivasaraghavan, EmELvar: A NeuroSymbolic Reasoner for the EL++ Description Logic., in: *SemREC@ ISWC*, 2021, pp. 44–51.

[32] S. Mondal, S. Bhatia and R. Mutharaju, EmEL++: Embeddings for EL++ Description Logic, in: *Proceedings of the AAAI 2021 Spring Symposium on Combining Machine Learning and Knowledge Engineering (AAAI-MAKE 2021), Stanford University, Palo Alto, California, USA, March 22-24, 2021*, A. Martin, K. Hinkelmann, H. Fill, A. Gerber, D. Lenat, R. Stolle and F. van Harmelen, eds, CEUR Workshop Proceedings, Vol. 2846, CEUR-WS.org, 2021.

[33] C.J. Mungall, C. Torniai, G.V. Gkoutos, S.E. Lewis and M.A. Haendel, Uberon, an integrative multi-species anatomy ontology, *Genome biology* **13** (2012), 1–20.

[34] O.L. Özcep, M. Leemhuis and D. Wolter, Embedding Ontologies in the Description Logic ALC by Axis-Aligned Cones, *Journal of Artificial Intelligence Research* **78** (2023), 217–267–. doi:10.1613/jair.1.13939.

[35] R. Peñaloza and A.-Y. Turhan, Instance-Based Non-standard Inferences in with Subjective Probabilities, in: *International Workshop on Uncertainty Reasoning for the Semantic Web*, Springer, 2008, pp. 80–98.

[36] X. Peng, Z. Tang, M. Kulmanov, K. Niu and R. Hoehndorf, Description Logic EL++ Embeddings with Intersectional Closure, 2022.

[37] P. Radivojac and W.T. Clark, Information-theoretic evaluation of predicted ontological annotations, *Bioinformatics* **29**(13) (2013), i53–i61. doi:10.1093/bioinformatics/btt228.
[38] A. Rector and J. Rogers, Patterns, properties and minimizing commitment: Reconstruction of the galen upper ontology in owl, in: *Proceedings of the EKAW*, Vol. 4, 2004.
[39] A.L. Rector, J.E. Rogers and P. Pole, The GALEN high level ontology, in: *Medical Informatics Europe'96*, IOS Press, 1996, pp. 174–178.
[40] Y. Sato, G. Stapleton, M. Jamnik and Z. Shams, Deductive reasoning about expressive statements using external graphical representations, in: *Proceedings of the 40th Annual Conference of the Cognitive Science Society*, Cognitive Science Society, 2018, pp. 0–0, CogSci 2018 ; Conference date: 25-07-2018 Through 28-07-2018.
[41] Y. Shiraishi and K. Kaneiwa, A Self-matching Training Method with Annotation Embedding Models for Ontology Subsumption Prediction, 2024.
[42] D.-I.B. Suntisrivaraporn, Polynomial-time reasoning support for design and maintenance of large-scale biomedical ontologies (2009).
[43] A. Tarski et al., On the concept of logical consequence, *Logic, semantics, metamathematics* **52** (1936), 409–420.
[44] Q. Wang, Z. Mao, B. Wang and L. Guo, Knowledge graph embedding: A survey of approaches and applications, *IEEE transactions on knowledge and data engineering* **29**(12) (2017), 2724–2743.
[45] Z. Wang, J. Zhang, J. Feng and Z. Chen, Knowledge graph embedding by translating on hyperplanes, in: *Proceedings of the AAAI conference on artificial intelligence*, Vol. 28, 2014.
[46] B. Xiong, N. Potyka, T.-K. Tran, M. Nayyeri and S. Staab, Faithful Embeddings for EL++ Knowledge Bases, in: *Proceedings of the 21st International Semantic Web Conference (ISWC2022)*, 2022, pp. 1–18.
[47] H. Yang, J. Chen and U. Sattler, TransBox: EL++-closed Ontology Embedding, in: *Proceedings of the ACM on Web Conference 2025*, 2025, pp. 22–34.
[48] F. Zhapa-Camacho and R. Hoehndorf, From Axioms over Graphs to Vectors, and Back Again: Evaluating the Properties of Graph-based Ontology Embeddings, in: *Proceedings of the 17th International Workshop on Neural-Symbolic Learning and Reasoning, La Certosa di Pontignano, Siena, Italy, July 3-5, 2023*, A.S. d'Avila Garcez, T.R. Besold, M. Gori and E. Jiménez-Ruiz, eds, CEUR Workshop Proceedings, Vol. 3432, CEUR-WS.org, 2023, pp. 85–102. https://ceur-ws.org/Vol-3432/paper7.pdf.
[49] F. Zhapa-Camacho, M. Kulmanov and R. Hoehndorf, mOWL: Python library for machine learning with biomedical ontologies, *Bioinformatics* (2022), btac811. doi:10.1093/bioinformatics/btac811.
[50] Gene Ontology Consortium: going forward, *Nucleic Acids Research* **43**(D1) (2014), D1049–D1056–. doi:10.1093/nar/gku1179.

Appendix A. GO & STRING data Statistics, Train Part

Dataset	GCI0	GCI1	GCI2	GCI3	GCI0_BOT	GCI1_BOT	GCI3_BOT	Classes	Relations	Test axioms
Yeast iw	81,068	11,825	269,567	11,823	0	31	0	61,846	16	12,040
Yeast hf	81,068	11,825	290,433	11,823	0	31	0	61,850	16	1,530

Appendix B. Food Ontology Statistics, Train Part

GCI0	GCI1	GCI2	GCI3	GCI0_BOT	GCI1_BOT	GCI3_BOT	Classes	Relations	Test axioms
21,795	1,267	10,719	897	0	495	0	24,969	43	5,752

Appendix C. GALEN Ontology Statistics, Train Part

GCI0	GCI1	GCI2	GCI3	GCI0_BOT	GCI1_BOT	GCI3_BOT	Classes	Relations	Test axioms
27,339	15,613	29,618	15,615	0	0	0	49,223	888	667

Appendix D. Hyperparameters

Dataset	Model	dim	lr	γ	ϵ	δ	λ
Yeast iw	ELEm	100	0.0001	-0.10			
	ELEm+l	50	0.0001	0.00			
	ELBE	200	0.0001	0.00			
	ELBE+l	200	0.0100	0.00	0.001		
	Box^2EL	200	0.0010	0.01		1	0.05
	Box^2EL+l	200	0.0010	0.01	0.010	2	0.05
Yeast hf	ELEm	200	0.0001	0.01			
	ELEm+l	50	0.0001	-0.10			
	ELBE	200	0.0001	0.10			
	ELBE+l	200	0.0001	0.10	0.010		
	Box^2EL	200	0.0100	0.10		4	0.20
	Box^2EL+l	200	0.0100	0.10	0.010	4	0.05
FoodOn	ELEm	400	0.0010	-0.10			
	ELEm+l	400	0.0010	-0.10			
	ELBE	200	0.0100	0.10			
	ELBE+l	200	0.0100	-0.01	0.001		
	Box^2EL	100	0.0100	0.10		1	0.20
	Box^2EL+l	200	0.0010	0.10	0.010	4	0.10
GALEN	ELEm	400	0.0010	-0.10			
	ELEm+l	400	0.0010	-0.01			
	ELBE	100	0.0010	0.10			
	ELBE+l	200	0.0010	0.01	0.010		
	Box^2EL	200	0.0010	0.00		4	0.05
	Box^2EL+l	200	0.0100	0.00	0.100	1	0.05

Appendix E. Deductive Closure Computation Example

Let us add two more axioms to the simple ontology example from Section 6.4 about proteins $\{P\}$ and $\{Q\}$ having functions $\{GO_1\}$ and $\{GO_2\}$, respectively. ELK will infer the following class hierarchy:

C	Concepts D where $C \sqsubseteq D$
$\bot$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
$\{P\}$	$\{P\}$, B, $\top$
$\{Q\}$	$\{Q\}$, A, $\top$
A	A, $\top$
B	B, $\top$
$\{GO_1\}$	$\{GO_1\}$, $\top$
$\{GO_2\}$	$\{GO_2\}$, $\top$
$\top$	$\top$

For GCI2 axioms $\{P\} \sqsubseteq \exists has\_function.\{GO_1\}$ and $\{Q\} \sqsubseteq \exists has\_function.\{GO_2\}$ the algorithm will output

C	Concepts $D \neq \bot$ where $C \sqsubseteq \exists has\_function.D$
$\bot$	$\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
$\{P\}$	$\{GO_1\}$, $\top$
$\{Q\}$	$\{GO_2\}$, $\top$

For GCI3 axioms $\exists has\_function.\{GO_1\} \sqsubseteq B$ and $\exists has\_function.\{GO_2\} \sqsubseteq A$ the algorithm will infer

$C \neq \bot$	Concepts D where $\exists has\_function.C \sqsubseteq D$
$\{P\}$	$\top$
$\{Q\}$	$\top$
A	$\top$
B	$\top$
$\{GO_1\}$	B, $\top$
$\{GO_2\}$	A, $\top$
$\top$	$\top$

In this small protein function prediction example there are two disjointness axioms: $A \sqcap B \sqsubseteq \bot$ and $\{GO_1\} \sqcap \{GO_2\} \sqsubseteq \bot$. Taking into consideration the concept hierarchy and inference rules from part 2 the algorithm will infer the following GCI1 and GCI1_BOT axioms:

C	D	Subsumptions E where $C \sqcap D \sqsubseteq E$
$\bot$	$\bot$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\{P\}$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\{Q\}$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	A	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	B	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\{GO_1\}$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\{GO_2\}$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\top$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
$\{P\}$	$\{P\}$	$\{P\}$, B, $\top$
	$\{Q\}$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	A	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	B	$\{P\}$, B, $\top$
	$\{GO_1\}$	$\{P\}$, $\{GO_1\}$, B, $\top$
	$\{GO_2\}$	$\{P\}$, $\{GO_2\}$, B, $\top$
	$\top$	$\{P\}$, B, $\top$
$\{Q\}$	$\{Q\}$	$\{Q\}$, A, $\top$
	A	$\{Q\}$, A, $\top$
	B	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\{GO_1\}$	$\{Q\}$, $\{GO_1\}$, A, $\top$
	$\{GO_2\}$	$\{Q\}$, $\{GO_2\}$, A, $\top$
	$\top$	$\{Q\}$, A, $\top$
A	A	A, $\top$
	B	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\{GO_1\}$	A, $\{GO_1\}$, $\top$
	$\{GO_2\}$	A, $\{GO_2\}$, $\top$
	$\top$	A, $\top$
B	B	B, $\top$
	$\{GO_1\}$	B, $\{GO_1\}$, $\top$
	$\{GO_2\}$	B, $\{GO_2\}$, $\top$
	$\top$	B, $\top$
$\{GO_1\}$	$\{GO_1\}$	$\{GO_1\}$, $\top$
	$\{GO_2\}$	$\bot$, $\{P\}$, $\{Q\}$, A, B, $\{GO_1\}$, $\{GO_2\}$, $\top$
	$\top$	$\{GO_1\}$, $\top$
$\{GO_2\}$	$\{GO_2\}$	$\{GO_2\}$, $\top$
	$\top$	$\{GO_2\}$, $\top$
$\top$	$\top$	$\top$

Appendix F. Deductive Closure Computation Soundness

Let us show that each inference rule provides truth statements:

1. $\dfrac{C \sqcap D \sqsubseteq E \quad C' \sqsubseteq C \quad D' \sqsubseteq D \quad E \sqsubseteq E'}{C' \sqcap D' \sqsubseteq E'}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqcap D \sqsubseteq E$ then $C^{\mathcal{I}} \cap D^{\mathcal{I}} \subseteq E^{\mathcal{I}}$. Also, $C'^{\mathcal{I}} \subseteq C^{\mathcal{I}}$, $D'^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ and $E^{\mathcal{I}} \subseteq E'^{\mathcal{I}}$. From this we derive $C'^{\mathcal{I}} \cap D'^{\mathcal{I}} \subseteq E'^{\mathcal{I}}$, i.e., $\mathcal{I} \models C' \sqcap D' \sqsubseteq E'$.

2. $\dfrac{C \sqsubseteq \exists R.D \quad C' \sqsubseteq C \quad D \sqsubseteq D' \quad R \sqsubseteq R'}{C' \sqsubseteq \exists R'.D'}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqsubseteq \exists R.D$ then $C^{\mathcal{I}} \subseteq \{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R^{\mathcal{I}} \wedge b \in D^{\mathcal{I}}\}$. Additionally, $C'^{\mathcal{I}} \subseteq C^{\mathcal{I}}$, $D^{\mathcal{I}} \subseteq D'^{\mathcal{I}}$ and for arbitrary $a, b \in \Delta^{\mathcal{I}}$ $(a,b) \in R^{\mathcal{I}} \Rightarrow (a,b) \in R'^{\mathcal{I}}$. Then $C'^{\mathcal{I}} \subseteq \{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R'^{\mathcal{I}} \wedge b \in D'^{\mathcal{I}}\}$, i.e., $\mathcal{I} \models C' \sqsubseteq \exists R'.D'$.

3. $\dfrac{C \sqsubseteq \exists R.D \quad D \sqsubseteq \exists R'.E \quad R \circ R' \sqsubseteq S}{C \sqsubseteq \exists S.E}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqsubseteq \exists R.D$ and $\mathcal{I} \models D \sqsubseteq \exists R'.E$ then $C^{\mathcal{I}} \subseteq \{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R^{\mathcal{I}} \wedge b \in D^{\mathcal{I}}\}$ and $D^{\mathcal{I}} \subseteq \{b \in \Delta^{\mathcal{I}} | \exists c \in \Delta^{\mathcal{I}} : (b,c) \in R'^{\mathcal{I}} \wedge c \in E^{\mathcal{I}}\}$. For arbitrary $a, b, c \in \Delta^{\mathcal{I}}$ $(a,b) \in R^{\mathcal{I}} \wedge (b,c) \in R'^{\mathcal{I}} \Rightarrow (a,c) \in S^{\mathcal{I}}$. Then $C^{\mathcal{I}} \subseteq \{a \in \Delta^{\mathcal{I}} | \exists c \in \Delta^{\mathcal{I}} : (a,c) \in S^{\mathcal{I}} \wedge c \in E^{\mathcal{I}}\}$, i.e., $\mathcal{I} \models C \sqsubseteq \exists S.E$.

4. $\dfrac{\exists R.C \sqsubseteq D \quad C' \sqsubseteq C \quad D \sqsubseteq D' \quad R' \sqsubseteq R}{\exists R'.C' \sqsubseteq D'}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models \exists R.C \sqsubseteq D$ then $\{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\} \subseteq D^{\mathcal{I}}$. Also, $C'^{\mathcal{I}} \subseteq C^{\mathcal{I}}$, $D^{\mathcal{I}} \subseteq D'^{\mathcal{I}}$ and for arbitrary $a, b \in \Delta^{\mathcal{I}}$ $(a,b) \in R'^{\mathcal{I}} \Rightarrow (a,b) \in R^{\mathcal{I}}$. From this follows $\{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R'^{\mathcal{I}} \wedge b \in C'^{\mathcal{I}}\} \subseteq D'^{\mathcal{I}}$, i.e., $\mathcal{I} \models \exists R'.C' \sqsubseteq D'$.

5. $\dfrac{C \sqcap D \sqsubseteq \bot \quad C' \sqsubseteq C \quad D' \sqsubseteq D}{C' \sqcap D' \sqsubseteq \bot}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqcap D \sqsubseteq \bot$ then $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$. Additionally, Also, $C'^{\mathcal{I}} \subseteq C^{\mathcal{I}}$, $D'^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, from where we can derive $C'^{\mathcal{I}} \cap D'^{\mathcal{I}} = \emptyset$, i.e., $\mathcal{I} \models C' \sqcap D' \sqsubseteq \bot$.

6. $\dfrac{C \sqcap D \sqsubseteq \bot}{C \sqcap D \sqsubseteq E}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqcap D \sqsubseteq \bot$ then $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$. Since $\emptyset \subseteq E^{\mathcal{I}}$ for any concept E then $C^{\mathcal{I}} \cap D^{\mathcal{I}} \subseteq E^{\mathcal{I}}$, i.e., $\mathcal{I} \models C \sqcap D \sqsubseteq E$.

7. $\dfrac{\exists R.C \sqsubseteq \bot \quad C' \sqsubseteq C \quad R' \sqsubseteq R}{\exists R'.C' \sqsubseteq \bot}$
Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models \exists R.C \sqsubseteq \bot$ then $\{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R^{\mathcal{I}} \wedge b \in C^{\mathcal{I}}\} = \emptyset$. Additionally, $C'^{\mathcal{I}} \subseteq C^{\mathcal{I}}$ and for arbitrary $a, b \in \Delta^{\mathcal{I}}$ $(a,b) \in R'^{\mathcal{I}} \Rightarrow (a,b) \in R^{\mathcal{I}}$. From this we get that $\{a \in \Delta^{\mathcal{I}} | \exists b \in \Delta^{\mathcal{I}} : (a,b) \in R'^{\mathcal{I}} \wedge b \in C'^{\mathcal{I}}\} = \emptyset$, i.e., $\mathcal{I} \models \exists R'.C' \sqsubseteq \bot$.

8. $\dfrac{}{C \sqcap \bot \sqsubseteq E}$
Let $\mathcal{I}$ be an arbitrary interpretation. For an arbitrary concept C $(C \sqcap \bot)^{\mathcal{I}} = C^{\mathcal{I}} \cap \emptyset = \emptyset$ and for every concept E we have $\emptyset \subseteq E^{\mathcal{I}}$. Hence $\mathcal{I} \models C \sqcap \bot \sqsubseteq E$.

9. $\dfrac{D \sqsubseteq \bot}{C \sqcap D \sqsubseteq E}$

Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models D \sqsubseteq \bot$ then $D^{\mathcal{I}} = \emptyset$. On analogy with the previous case we get $\mathcal{I} \models C \sqcap D \sqsubseteq E$ for arbitrary concepts C and E.

10. $\dfrac{E \sqsubseteq E'}{C \sqcap E \sqsubseteq E'}$

Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models E \sqsubseteq E'$ then $E^{\mathcal{I}} \subseteq E'^{\mathcal{I}}$. For every concept C $(C \sqcap E)^{\mathcal{I}} = C^{\mathcal{I}} \cap E^{\mathcal{I}} \subseteq E^{\mathcal{I}} \subseteq E'^{\mathcal{I}}$, hence $\mathcal{I} \models C \sqcap E \sqsubseteq E'$.

11. $\dfrac{C \sqcap D \sqsubseteq \bot}{C \sqcap D \sqsubseteq E}$

Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqcap D \sqsubseteq \bot$ then $C^{\mathcal{I}} \cap D^{\mathcal{I}} = \emptyset$. Since $\emptyset \subseteq E^{\mathcal{I}}$ for every concept E then $\mathcal{I} \models C \sqcap D \sqsubseteq E$.

12. $\dfrac{C \sqsubseteq E \quad D \sqsubseteq E \quad C' \sqsubseteq C \quad D' \sqsubseteq D \quad E \sqsubseteq E'}{C' \sqcap D' \sqsubseteq E'}$

Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqsubseteq E$ and $\mathcal{I} \models D \sqsubseteq E$ then $C^{\mathcal{I}} \subseteq E^{\mathcal{I}}$ and $D^{\mathcal{I}} \subseteq E^{\mathcal{I}}$. Note that $C^{\mathcal{I}} \cap D^{\mathcal{I}} \subseteq C^{\mathcal{I}} \subseteq E^{\mathcal{I}}$, from this we get $\mathcal{I} \models C \sqcap D \sqsubseteq E$. On analogy with case 1 we derive that $\mathcal{I} \models C' \sqcap D' \sqsubseteq E'$.

13. $\dfrac{C \sqsubseteq C'}{C \sqcap \top \sqsubseteq C'}$

Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqsubseteq C'$ then $C^{\mathcal{I}} \subseteq C'^{\mathcal{I}}$. By definition of $\top$, $(C \sqcap \top)^{\mathcal{I}} = C^{\mathcal{I}} \cap T^{\mathcal{I}} = C^{\mathcal{I}}$, hence $\mathcal{I} \models C \sqcap \top \sqsubseteq C'$.

14. $\dfrac{}{\bot \sqsubseteq \exists R.D}$

Follows immediately from the fact that $\emptyset$ is a subset of any concept interpretation.

15. $\dfrac{C \sqsubseteq \bot}{C \sqsubseteq \exists R.D}$

Let $\mathcal{I}$ be an arbitrary interpretation. Since $\mathcal{I} \models C \sqsubseteq \bot$ then $C^{\mathcal{I}} = \emptyset$. On analogy with case 14 we get $\mathcal{I} \models C \sqsubseteq \exists R.D$.

16. $\dfrac{}{\exists R.C \sqsubseteq \top}$

Follows immediately from the fact that $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$.