FALCON: Sound and Complete Neural Semantic Entailment over ALC Ontologies

Zhenwei Tang¹,², Tilman Hinnerichs¹, Xi Peng¹, Xiangliang Zhang³, Robert Hoehndorf¹

¹ Computational Abdullah BioScience Research Center, Computer, Electrical and Mathematical Sciences & Engineering Division, King Abdullah University of Science and Technology, ² University of Toronto, ³ University of Notre Dame
{zhenwei.tang, tilman.hinnerichs, xi.peng, robert.hoehndorf}@kaust.edu.sa, xzhang33@nd.edu

Abstract
Many ontologies, i.e., Description Logic (DL) knowledge bases, have been developed to provide rich knowledge about various domains, and a lot of them are based on ALC, i.e., a prototypical and expressive DL, or its extensions. The main task that explores ALC ontologies is to compute semantic entailment. Symbolic approaches can guarantee sound and complete semantic entailment but are sensitive to inconsistency and missing information. To this end, we propose FALCON, a Fuzzy ALC Ontology Neural reasoner. FALCON uses fuzzy logic operators to generate single model structures for arbitrary ALC ontologies, and uses multiple model structures to compute semantic entailments. Theoretical results demonstrate that FALCON is guaranteed to be a sound and complete algorithm for computing semantic entailments over ALC ontologies. Experimental results show that FALCON enables not only approximate reasoning (reasoning over incomplete ontologies) and paraconsistent reasoning (reasoning over inconsistent ontologies), but also improves machine learning in the biomedical domain by incorporating background knowledge from ALC ontologies. FALCON is available at https://github.com/bio-ontology-research-group/FALCON.

1 Introduction
The prototypical description logic Attributive Concept Language with Complements (ALC) is the basis of many description logics (Schmidt-Schaub and Smolka 1991, Baader et al. 2003). A large number of description logic knowledge bases (ontologies) have been developed to provide rich and accurate knowledge about various domains, in particular in the biomedical field where hundreds of ontologies have been created (Smith et al. 2007). Semantic entailment is the core task to explore ALC ontologies and it is defined upon model structures (models) (Baader et al. 2003), i.e., algebraic structures in which a set of statements is true. A statement \( \phi \) is semantically entailed by the ontology \( O \) if \( \phi \) is true in all models of \( O \). An algorithm to compute semantic entailments is called sound if all statements it computes are correct; it is called complete if all entailed statements can be computed.

Sound and complete symbolic approaches have been developed to compute semantic entailments over ALC ontologies (Schild 1991, Donini and Massacci 2000). These algorithms 1) will no longer compute useful entailments when applied to inconsistent ontologies (because all statements are entailed), and 2) do not allow for approximate entailments when applied to ontologies with missing information. Recently, neural semantic entailment methods (Kulmanov et al. 2019, Ren, Hu, and Leskovec 2020, Badreddine et al. 2022) have been developed to deal with inconsistency and missing information, and make semantic entailment accessible to neural networks. However, they either cope with only some aspects of semantic entailment such as logical query answering (Hamilton et al. 2018), limit the scope to simpler DLs such as EL (Kulmanov et al. 2019), or are not sound and complete. Therefore, previous neural approaches do not guarantee a sound and complete algorithm for semantic entailment over ALC ontologies.

We developed FALCON: Fuzzy ALC Ontology Neural Reasoner for sound and complete semantic entailment over ALC ontologies. Since semantic entailment relies on all models of an ALC ontology \( T \), FALCON needs to generate potentially any model of \( T \). To achieve this goal, FALCON uses fuzzy logic operators, an intermediate embedding space, and a model-generating function. The fuzzy logic operators provide a differentiable means to constrain optimization while preserving logical semantics (van Krieken, Acar, and van Harmelen 2020), and the separation of the embedding space from the models allows us to generate models of arbitrary cardinality. FALCON generates multiple such models and uses them to compute semantic entailment. The ability to generate any model combined with the generation of multiple models enables sound and complete semantic entailment when infinitely many models are generated, and also enables reasoning under the open world assumption (Reiter 1981); they also allow approximate entailments and computing entailments in the presence of inconsistency. We summarize the main contributions of this work as:

- To the best of our knowledge, we are the first to propose a neural approach for computing semantic entailment over ALC ontologies; we use differentiable fuzzy logic operators to generate model structures for ALC ontologies, and then generate multiple such models to implement an algorithm for semantic entailment;
- We prove that our algorithm, FALCON, is sound and complete for semantic entailment over ALC ontologies;
• Experimental results show that FALCON enables not only approximate and paraconsistent reasoning, but also ALC-enhanced scientific facts discovery.

2 Preliminaries

2.1 Description Logic ALC

Description logics (DLs) are fragments of first-order logic (Baader et al., 2003). A signature \( \Sigma \) of a DL theory consists of a set of concept symbols \( C \), a set of relation symbols \( R \), and a set of individual symbols \( I \). In the DL ALC (Baader et al., 2003), every concept symbol is a concept description; if \( C \) and \( D \) are concept descriptions and \( r \) a relation symbol, then \( C \cap D, C \cup D, \neg C, r.C, \text{and } \exists r.C \) are concept descriptions. Axioms in ALC are divided in TBox and ABox axioms. Let \( C \) and \( D \) be concept descriptions, \( a \) and \( b \) individual symbols, \( r \) a relation symbol. A TBox axiom has the form \( C \sqsubseteq D \), and an ABox axiom has the form \( C(a) \) or \( R(a,b) \). A TBox is a set of TBox axioms, and an ABox a set of ABox axioms. An interpretation \( I = (\Delta^I, \cdot^I) \) consists of a non-empty domain \( \Delta^I \) and an interpretation function \( \cdot^I \) such that \( C^I \subseteq \Delta^I \) for all \( C \in \mathcal{C} \), \( R^I \subseteq \Delta^I \times \Delta^I \) for all \( R \in \mathcal{R} \), and \( a^I \in \Delta^I \) for all \( a \in \mathcal{I} \). The interpretation function is extended to concept descriptions as:

\[
\begin{align*}
(C \cap D)^I & := C^I \cap D^I, \\
(C \cup D)^I & := C^I \cup D^I, \\
(\forall R.C)^I & := \{d \in \Delta^I | \forall e \in \Delta^I : (d, e) \in R^I \implies e \in C^I\}, \\
(\exists R.C)^I & := \{d \in \Delta^I | \exists e \in \Delta^I : (d, e) \in R^I \text{ and } e \in C^I\}, \\
(\neg C)^I & := \Delta^I - C^I.
\end{align*}
\]

2.2 Semantic Entailment

Semantic entailment is the core task to explore ALC ontologies and it is defined upon model structures (models) (Baader et al., 2003). An interpretation \( I \) is a model of TBox \( T \) if \( C^I \subseteq \Delta^I \) for all axioms \( C \sqsubseteq D \) in \( T \); it is a model of ABox \( A \) if \( a^I \in C^I \) for all \( C(a) \in A \) and \( (a^I, b^I) \in R^I \) for all \( R(a,b) \in A \); it is a model of an ALC theory \( O = \{A, T\} \) if it is a model of \( T \) and \( A \). A statement \( \phi \) is semantically entailed by the ontology \( O \) if \( \phi \) is true in all models of \( O \). An algorithm to compute semantic entailments is called sound if all statements it computes are correct; it is called complete if all entailed statements can be computed. Semantic entailment over ALC ontologies is defined under open world assumption (Reiter, 1981), i.e., the assumption that our knowledge of the world is incomplete and non-provable statements are not necessarily false.

3 Related Work

3.1 Symbolic ALC Reasoners

Several automated reasoners (Shearer, Motik, and Horrocks 2008; Tsarkov and Horrocks 2006) implement sound and complete algorithms for semantic entailment over ALC ontologies. However, they will no longer compute useful entailments when applied to inconsistent ontologies because all statements are entailed. Several paraconsistent reasoning approaches (Schlobach, Cornet et al. 2003; Flouris et al. 2008; Kaminski, Knorr, and Leite 2015) were proposed to partially solve this issue, but discard potentially crucial knowledge (Bienvenu 2020). Sound and complete symbolic approaches do not allow for approximate semantic entailment, which may be useful when applied to incomplete knowledge bases. For example, symbolic approaches will not entail Father \( \sqsubseteq \text{Parent} \sqcap \text{Male} \) from Father \( \sqsubseteq \text{Parent} \) in the absence of Father \( \sqsubseteq \text{Male} \), even if all named instances of Father are also instances of Male. FALCON allows for both paraconsistent and approximate semantic entailment.

3.2 Neural Logical Reasoners

Logical query answering methods (Hamilton et al., 2018; Ren, Hu, and Leskovec 2020; Tang et al., 2022) reason over knowledge graphs, which are subsets of ontologies that only include ABox axioms of the type \( R(a,b) \) and potentially \( C(a) \). Knowledge graph completion methods (Bordes et al., 2013; Yang et al., 2015; Tang et al., 2022) can be regarded as a special case where queries are restricted to the form of 1-projection (Ren and Leskovec, 2020). However, these methods are limited to subsets of ontologies and thus only deal with a subproblem of semantic entailment. Another set of methods reason over ontologies by generating a single model based on geometric shapes for \( \mathcal{EL}^\leftrightarrow \) (Kulmanov et al., 2019; Sun et al., 2020; Mondal, Bhattacharya, and Mutharaju, 2021; Peng et al., 2022) or ALC (Ozcep, Leemhuis, and Wolter, 2021). These approaches rely on random samples and cannot represent all models of an ontology, and are therefore not suitable for semantic entailment. More recently, Logic Tensor Network (LTN) (Badreddine et al., 2022) and its extensions (Luus et al., 2021; Wagner and d’Avila Garcez, 2022) enable neural first-order logic (FOL) reasoning. Since DLs are fragments of FOL, ALC can be handled by LTNs with some modifications. However, LTNs are either used under closed-world assumptions or limited to finite universes. Therefore, they cannot generate all models of an ontology and therefore cannot be used for sound and complete semantic entailment algorithms for ALC ontologies.

4 Methods

We first show how to construct single models in Section 4.1 and then how to perform semantic entailment with multiple generated models in Section 4.2.

4.1 Generating model structures

Overview We rely on fuzzy sets to define interpretations; a fuzzy set \((U, m)\) is an ordered pair of a non-empty set of entities \(U\) and membership function \(m : U \to [0, 1] ; m\) specifying a fuzzy subset of \(U\), i.e., assigning a degree of membership \(m(x)\) to each \(x \in U\). Our aim is to identify a function \(f_{mod}\) that, given an ontology \(O\) with signature \(\Sigma\), generates a model of \(O\). We split the function \(f_{mod}\) in two parts: The first part, \(f_e : \Sigma \to \mathbb{R}^n\), generates an embedding of symbols in \(\Sigma \) in \(\mathbb{R}^n\). The dimension \(n\) is a parameter and may be chosen differently for different types of symbols (concept, relation, or individual names); we chose the same value \(n\) for all types of symbols for convenience. We then fix
a degree of membership in concepts, that maps representations in \( \mathbb{R}^n \) to fuzzy subsets of \( \Delta \) for embeddings of concept symbols, to fuzzy subsets of \( \Delta \times \Delta \) for embeddings of relation symbols, and to elements of \( \Delta \) for individual symbols. Our goal is that \( f_{\text{mod}} = f_{\text{e}} \circ f_{\text{x}} \) is the interpretation function that generates a fuzzy model \( I \) of \( O \), i.e., ensures that, for all TBox axioms \( C \sqsubseteq D \) in \( O \) (with \( C, D \) arbitrary concept descriptions), \( C^2 \subseteq D^2 \) holds, for all axioms \( R(a,b) \), \((a^2, b^2) \in R^2 \) holds, and for all axioms \( C(a) \), \( a^2 \in C^2 \) holds. We therefore assign a degree of membership in \( C^2 \) to all elements of \( \Delta \).

Given the signature \( \Sigma = (C, R, I) \) of ontology \( O \), we add to \( \Sigma \) a set of individual symbols \( I_{R^n} \), where \( I_{R^n} \) contains a new individual symbol for every member of the set \( \mathbb{R}^n \); \( \mathbb{R}^n \) is the same space that we use for generating our embeddings, and \( f_e(x) = x \) for every \( x \in I_{R^n} \). In other words, we extend our signature by introducing a new individual symbol for every element of the set \( \mathbb{R}^n \); through the requirement that \( f_e(x) = x \), this individual symbol is both a symbol (i.e., a name) and a distributed representation (an embedding in \( \mathbb{R}^n \)) at the same time. This expansion makes the signature uncountable, and interpretations may also become uncountable (i.e., have an uncountable universe).

We now have the challenge that, for concept and relation interpretations, we must provide a degree of membership for each element of an uncountable set. We address this problem by sampling a (finite) set of individual symbols from \( I_{R^n} \), in addition to all the individual symbols from \( I \), and assigning \( m(\cdot) \) (defined in Eqs. 13, 14) accordingly. During training, we will repeatedly sample finite subsets of \( I_{R^n} \) (and therefore of \( \mathbb{R}^n \)), whereas during the prediction phase, the degree of membership of any element of \( I_{R^n} \) in any concept description \( C \) can be computed through Eqn. 8 because the embedding of an individual \( x \) is either the result of the embedding function \( f_e \) (when \( x \) is a named individual in \( \Sigma \)) or simply \( x \) (when \( x \) is not named, because \( f_e(x) = x \) in this case). The sampling strategy can be either uniformly or constrained in various ways (e.g., by sampling in the proximity of embeddings of named individuals). \( \Sigma \) is finite and therefore \( f_e \) generates a finite representation, whereas \( \Delta \) may be of arbitrary cardinality.

**Concept Symbols** We chose \( \Delta \) to be the universe consisting of all individual names plus the sampled individuals. We express interpretations of concept descriptions by assigning a degree of membership in \( C^2 \) to all elements of \( \Delta \). Specifically, we interpret a concept name \( C \) as

\[
m(x, C^2) = \sigma(\text{MLP}(f_e(C), f_e(x)))
\]

where \( f_e(C) \) is the embedding (in \( \mathbb{R}^n \)) of \( C \), \( f_e(x) \) the embedding of the individual \( x \) (either an individual symbol or an individual sampled from \( \mathbb{R}^n \)), \( \sigma \) the sigmoid function.

In other words, \( \text{FALCON} \) generates the degree of membership of \( x^2 = x \) in \( C^2 \) based on the concept embedding \( f_e(C) \), the embedding \( f_e(x) \) of individual \( x \), and a multilayer perceptron (MLP).

**Relation Symbols** The interpretation of relation symbols in \( \Sigma \) is a set of tuples of elements of \( \Delta \) combined with a degree of memberships and can be written as the pair \((m, \Delta \times \Delta)\), which is defined as:

\[
m((x, y), R^2) = \sigma(\text{MLP}(f_e(x) + f_e(R), f_e(y)))
\]

**Concept Descriptions** Interpretations of a concept description \( C \), where \( C \) is not a concept name, are defined recursively. For the operators \( \sqcap, \sqcup, \text{and } \neg \), we define the membership function \( m \) as

\[
m(x, (C_1 \sqcap C_2)^2) = \theta(m(x, C_1^2), m(x, C_2^2))
\]

\[
m(x, (C_1 \sqcup C_2)^2) = \kappa(m(x, C_1^2), m(x, C_2^2))
\]

\[
m(x, (\neg C)^2) = \nu(m(x, C^2))
\]

Here, \( \theta \) stands for a \( t \)-norm, \( \kappa \) for the corresponding \( t \)-conorm, and \( \nu \) for fuzzy negation. A \( t \)-norm (triangular norm) is a binary operator \( \theta : [0, 1] \times [0, 1] \rightarrow [0, 1] \), which is associative, commutative, has 1 as its identity element and is monotone in both arguments, i.e., whenever \( x \leq x' \) and \( y \leq y' \), then \( \theta(x, y) \leq \theta(x', y') \). We assume \( \theta \) to be continuous, and \( \theta(x, y) = 0 \) if and only if \( x = 0 \) or \( y = 0 \) (and therefore equivalent to crisp semantics when \( x, y \in \{0, 1\} \)). The \( t \)-conorm is a binary operator, defined as \( \kappa(x, y) := 1 - \theta(1 - x, 1 - y) \). The fuzzy negation \( \nu(x) \) is strong, i.e., if \( x < x' \), then \( \nu(x) > \nu(x') \), and involutive, i.e., \( \forall x \in [0, 1] : \nu(\nu(x)) = x \).

For the membership of \( x \) in \((\exists R, D)^2\), we iterate through the set of individuals in \( \Delta \) to find the maximum of the intersection of membership in \( D^2 \) and standing in relation \( R \) to \( x \) (i.e., membership of \((x, y)\) in \( R^2 \)) as:

\[
m(x, (\exists R, D)^2) = \max_{y \in \Delta} \theta(m(y, D^2), m((x, y), R^2))
\]

For the universal quantifier, we follow a similar approach and assign the degree of membership of \( x \) in \((\forall R, D)^2\) as:

\[
m(x, (\forall R, D)^2) = \min_{y \in \Delta} \kappa(m(y, D^2), m((x, y), R^2))
\]

In these formulations, the choice of \( t \)-norm, \( t \)-conorm, and negation (\( \nu \)) are parameters of our method.

**Optimization** We formulate the problem of finding a model of TBox axioms \( C \subseteq D \) as maximizing the degree of membership of any individual in \((\neg C \sqcup D)^2\), or, alternatively, minimizing the degree of membership in \((C \cap \neg D)^2\); this is justified because \( \mathcal{I} \) is a model of \( O \) if for all \( C \subseteq D, C^2 \subseteq D^2 \), or \((\neg C \sqcup D)^2 = \Delta, \) or \((C \cap \neg D)^2 = \emptyset \). W.l.o.g., we chose the second formulation and solve a minimization problem. First, we normalize the TBox and rewrite all axioms of the type \( C \subseteq D \) as \( C \sqsubseteq \neg D \sqsubseteq \bot \); the entire TBox can then be represented as a disjunction of conjunctions, which is subsumed by \( \bot \), i.e., \( \mathcal{I} \models C \sqsubseteq D \iff C^2 \subseteq D^2 \iff (C \cap \neg D)^2 = \emptyset \). With this formulation, given a set of individual symbols \( E \) sampled from \( I_{R^n} \) and given the TBox \( \mathcal{T} \), the degree of membership of entities in each of the disjuncts can be minimized through this loss:

\[
\mathcal{L}_T = \frac{1}{|E|} \frac{1}{|\mathcal{T}|} \sum_{C \subseteq D \in \mathcal{T}} \sum_{e \in E} m(e, (C \cap \neg D)^2)
\]
This loss ensures that TBox axioms are satisfied in the interpretation generated.

A second loss is defined for ABox axioms of the type $C(e)$ (concept assertion); this loss aims to ensure that axioms of the type $C(e)$ are satisfied and will maximize the degree of membership of $e$ in $C^2$:

$$\mathcal{L}_{A_1} = \frac{1}{|A_1|} \sum_{C(e) \in A_1} (1 - (m(e, C^2)))$$  \hspace{1cm} (10)

A third loss maximizes membership of pairs of individuals $(e_1, e_2)$ in the interpretation of the relation $R^2$ if the axiom $R(e_1, e_2)$ is included in the ABox:

$$\mathcal{L}_{A_2} = \frac{1}{|A_2|} \sum_{R(e_1, e_2) \in A_2} (1 - m((e_1, e_2), R^2))$$  \hspace{1cm} (11)

The final loss function is:

$$\mathcal{L} = \alpha \mathcal{L}_T + \beta \mathcal{L}_{A_1} + (1 - \alpha - \beta) \mathcal{L}_{A_2}$$  \hspace{1cm} (12)

with $\alpha, \beta \in [0, 1]$ and $\alpha + \beta < 1$.

The algorithm randomly initializes $f_e$ and MLP (from Eqs. 5 and 6 which generates the interpretation of concept names and relation names) and minimizes $\mathcal{L}$ through gradient descent. The choice of $t$-norm and $t$-conorm as well as fuzzy negation are parameters of the algorithm, and any differentiable $t$-norm and $t$-conorm (van Krieken, Acar, and van Harmelen 2020) can be used. Furthermore, the number of individuals to sample from the embedding space $(I_{g,n})$, and the sampling strategy, are parameters of the algorithm. As a result of our algorithm, we generate (or approximate) one model of the input ontology $O$, by first creating distributed representations (through $f_e$) of the symbols in the signature $\Sigma$ of $O$, and then an interpretation $I$ through the interpretation-generating function MLP (as well as $\alpha^2 = a$ for individual symbols).

### 4.2 Semantic entailment

We use the degree of membership to define the degree of satisfiability of a concept description $C$:

**Definition 1** (Degree of satisfiability). A concept $C$ is called satisfiable with respect to TBox $T$ if there is a model $I$ of $T$ and an $a \in \Delta^2$ such that $m(a, C^2) = 1$. Let $\text{Mod}(T)$ be the class of all models of $T$. $C$ is satisfiable to degree $\alpha$ with respect to TBox $T$ if $I = \alpha = \max_{I \in \text{Mod}(T)} \max_{a \in \Delta^2} m(a, C^2)$; we write $\tau(T, C) = \alpha$.

We can use degrees of satisfiability to explain truth values of subsumptions between concept descriptions.

**Definition 2** (Truth value of subsumption). Let $C$ and $D$ be concept descriptions. The truth value of the subsumption statement $C \subseteq D$ with respect to TBox $T$ is the degree of satisfiability of $\neg C \sqcup D$ with respect to $T$, $\tau(T, \neg C \sqcup D)$.

Semantic entailment ($\models$) of a TBox axiom $C \subseteq D$ (with $C$ and $D$ being concept descriptions) is then defined as a relation between classes of models, i.e., $T \models C \subseteq D$ if $\text{Mod}(T) \subseteq \text{Mod}(C \subseteq D)$. In general, $\text{Mod}(T)$ will be a class, and it will be impossible to enumerate $\text{Mod}(T)$.

However, we can sample from $\text{Mod}(T)$ using our algorithm to construct models in order to implement an approximate form of semantic entailment. Let $\text{Mod}_k(T)$ be a set of $k$ models generated through our model-constructing algorithm (which will form a subset with cardinality $k$ of $\text{Mod}(T)$); we can then define that $T \models_k C \subseteq D$ if $\text{Mod}_k(T) \subseteq \text{Mod}(C \subseteq D)$. As $C \subseteq D$ may have a different truth value in each of these $k$ models, we also assign a degree of entailment $\alpha$ and define $T \models^\alpha_k C \subseteq D$ if $\alpha = \min_{I \in \text{Mod}_k(T)} \max_{a \in \Delta^2} m(a, (\neg C \sqcup D)^2)$. In other words, we generate $k$ models through our algorithm and identify the minimum degree at which $\neg C \sqcup D$ is satisfied in any of these $k$ models.

A similar approach can be used to formulate the instantiation reasoning task, i.e., the task of finding all (named) individuals that are the instance of a class description. In this case, we test for entailment of $T \cup A \models C(a)$ by $\text{Mod}_k(T \cup A) \subseteq \text{Mod}(C(a))$, with $a$ being an individual name and $C$ a concept description; we define $T \cup A \models^\alpha_k C(a)$ as $\min_{I \in \text{Mod}_k(T \cup A)} m(a, C^2) \geq \alpha$, i.e., the minimum degree of membership of $a$ in the interpretation of $C$ across all $k$ models. Entailment of relation assertions can be formalized similarly.

We can further test for consistency of an ABox; in our case, we can define a degree of consistency.

**Definition 3** (Degree of consistency). ABox $A$ is consistent with respect to TBox $T$ iff $A$ has a model $I$ that is also a model of $T$. ABox $A$ is consistent with respect to TBox $T$ to degree $\alpha$ if $\alpha = \max_{I \in \text{Mod}(T)} \min_{C(a) \in A} m(a, C^2) \geq \alpha$, i.e., the minimum degree of membership of $a$ in the interpretation of $C$ across all $k$ models.

In other words, there is at least a model $I^*$ that is not only a model of $A$ but also a model of $T$. The detailed computation steps are shown in Algorithm 1 in Supplementary Materials.

### 5 Theoretical Results

Here we prove soundness and completeness of F-ALC\textsc{CON}. We first prove that F-ALC\textsc{CON} describes a (classical) model of a DL theory if the loss is 0.

**Theorem 1.** Let $T$ be a TBox and $A$ an ABox in ALC over signature $\Sigma = \{\text{C, R, I \in I_{g,n} \cup I_{f,n}}\}$. If $L = 0$ and the $t$-norm $\theta$ satisfies $\theta(x, y) = 0$ if and only if $x = 0$ or $y = 0$, then the interpretation $I$ with $C^2 = f_2(f_1(C))$ for all $C \in C$, $R^2 = f_2(f_1(C))$ for all $R \in (R)$, and $\alpha^2 = a$ for all $a \in I$ is a classical model of $T$ and $A$ with domain $\Delta = \mathbb{R}^n$.

**Proof.** By definition, $f_e \circ f_2$ generates an interpretation. We need to show that such an interpretation is a model, i.e., all axioms in $T \cup A$ are true in the interpretation $I$ generated by $f_e \circ f_2$ if $L = 0$.

For concept descriptions $C, D$, we need to show that, if the loss is 0, in the interpretation $I$, the degree of membership for all individuals in $(C \sqcup \neg D)^2 = 0$ for every $C \subseteq D \subseteq T$. If $L = 0$ implies $L = 0$. Because membership is always positive, i.e., $m(\cdot, \cdot) \geq 0$, $m(x, (C \sqcup \neg D)^2) = 0$ is true for all $C \subseteq D \subseteq T$ and $x \in E$.

Consider $m(x, (C \sqcup \neg D)^2) = 0$ for some $x \in E$. Let $C$ and $D$ be concept names (induction start), then, by the
condition that \( \theta(x_1, x_2) = 0 \) iff \( x_1 = 0 \) or \( x_2 = 0 \), either \( m(x, C^2) = 0 \) or \( m(x, (\neg D)^2) = 0 \) for any \( x \); by monotonicity of fuzzy negation and Eqn. 6, either \( m(x, C^2) = 0 \) or \( m(x, D^2) = 1 \).

Let \( C \) and \( D \) be concept descriptions (induction step). If \( C = \neg A \) then we minimize \( \nu(m(x, A^2)) \); by monotonicity of \( \nu \) this maximizes \( m(x, A^2) \), i.e., \( m(x, A^2) = 1 \) for all \( x \); analogously for \( D \). If \( C = A \cap B \) then we minimize \( \theta(m(x, A^2), m(x, B^2)) \); \( \theta \) is commutative and monotone, and minimizing the \( t \)-norm will minimize at least one conjunct; the value is minimal (0) only when one conjunct is 0 (and maximal when both are 1 for maximizing \( D^2 \)). Let \( C = A \cup B \); the \( \cap \)-norm is a strict t-norm when membership in both \( A^2 \) and \( B^2 \) is 0 (either \( m(x, A^2) = 1 \) or \( m(x, B^2) = 1 \) when maximizing \( D^2 \)). For \( C = \exists R.A \), we rely on Eqn. 7 for generating the interpretation of \( R \), Eqn. 8 relies on an MLP and combines with \( A^2 \) through a \( t \)-norm; \( C^2 \) is minimized if \( m(x, A^2) = 0 \) (induction hypothesis), or the \( \max_{y \in \Delta} \) (which runs over finite subsets of \( \Delta \) sampled from \( I \) is 0 when no individual \( y \) stands in relation \( R \) to some member of \( A^2 \); because we use \( \theta \) in Eqn. 6 this entails \( m((x, y), R^2) = 0 \). The argument runs analogously for \( C = \forall R.A \), and similarly for maximizing \( D^2 \).

If \( L = 0 \) then \( a^2 \in C^2 \) for all \( C(a) \in A \). \( L = 0 \) implies \( \mathcal{L}_A = \{ \chi \} \sum_{i \in A}(1 - (m(e, C^2))) = 0 \). If \( C \) is a concept name, the degree of membership is 1 for all \( C^2 \) with \( C(a) \in A \). If \( C \) is a concept description, the statement follows inductively as for TBox axioms and \( \mathcal{L}_R \).

If \( L = 0 \) then \( (a^2, b^2) \in R^2 \) for all \( R(a, b) \in A \). In the absence of axioms for relations, this follows directly from the definition of \( \mathcal{L}_A \).

We also need to show that every (classical) model can be represented by our algorithm, i.e., that we can reach a loss of 0 for any model. \( \mathcal{A} \mathcal{C} \mathcal{L} \) has the “finite model property”, i.e., every statement that may be false in a model is already false for any model.

Corollary 1 (Soundness). Let \( \text{Mod}_k (\mathcal{T} \cup A) \) be the \( k \) models generated by \( \text{FALCON} \) from \( \mathcal{T} \cup A \). If \( \phi \) is true in all \( I \in \text{Mod}_k (\mathcal{T} \cup A) \) and \( k = \omega \) (the smallest infinite ordinal), \( \phi \) is semantically entailed by \( \mathcal{T} \cup A \).

Sound entailment requires generating all models of a theory, and while \( \text{FALCON} \) can represent all such models (and find the representation through exhaustive search of parameters of \( f_q \) and \( f_t \)), the class of all models cannot be generated in finite time.

Corollary 2 (Completeness). If \( \phi \) is semantically entailed by \( \mathcal{T} \cup A \), it can be inferred by \( \text{FALCON} \).

Note: we may only perform sound and complete semantic entailment if \( \mathcal{T} = 0 \) for all models, which will not hold when the ontology is inconsistent.

6 Experiments

We conduct extensive experiments to answer the following research questions: RQ1: How well can \( \text{FALCON} \) perform approximate semantic entailment? RQ2: Is \( \text{FALCON} \) robust to inconsistency? RQ3: Can \( \text{FALCON} \) improve machine learning in the biomedical domain by incorporating background knowledge from \( \mathcal{A} \mathcal{C} \mathcal{L} \) ontologies?

6.1 Experimental Settings

Datasets We use three ontologies for evaluation with special, respective properties for different use cases. (1) The Family Ontology (Eqn. 14) is sufficiently small to manually evaluate learned representations and models as we can inspect the generated degrees of memberships; this also allows us to identify “unprovable” axioms, i.e., axioms that are neither entailed nor disproved (false in all possible models). Such axioms are crucial for open-world reasoning, and we use the Family Ontology to test the open-world semantic entailment capability of \( \text{FALCON} \) by inspecting the degrees of memberships in multiple models. We use the Family Ontology with one named individual for each concept name, and individuals sampled from \( \mathbb{R}^n \). (2) The Pizza Ontology is an ontology that is widely used in teaching about the Semantic Web. It is of moderate size and complexity and gives rise to several non-trivial inferences. We can add axioms that make the Pizza Ontology inconsistent, e.g., by generating

Note: if it is not possible to reduce \( \mathcal{L} \) to 0, then \( \mathcal{T} \cup A \) has no (classical) model, but there will still be an interpretation that optimizes the objective function; such interpretations will form the basis for paraconsistent reasoning. Furthermore, we have not shown how to find the functions \( f_q \) and \( f_t \), but have (non-constructively) shown that every (finite) interpretation is representable using \( \text{FALCON} \) due to the use of MLPs as universal approximators and the choice of a strict \( t \)-norm. The following two statements follow directly from the two theorems above and the finite model property of \( \mathcal{A} \mathcal{C} \mathcal{L} \).

\[ \text{Corollary 1 (Soundness). Let } M_{\text{Mod}}(\mathcal{T} \cup A) = \text{the } k \text{ models generated by } \text{FALCON} \text{ from } \mathcal{T} \cup A. \text{ If } \phi \text{ is true in all } I \in M_{\text{Mod}}(\mathcal{T} \cup A) \text{ and } k = \omega (\text{the smallest infinite ordinal}), \phi \text{ is semantically entailed by } \mathcal{T} \cup A. \]

\[ \text{Corollary 2 (Completeness). If } \phi \text{ is semantically entailed by } \mathcal{T} \cup A, \text{ it can be inferred by } \text{FALCON}. \]
an individual that is both a Meaty Pizza and a Vegetarian Pizza (which are disjoint in the Pizza Ontology). In this way, we can evaluate the robustness of \textsc{falcon} to inconsistency using a complex ontology. (3) The Human Phenotype Ontology (HPO) (Köhler et al. 2020) is a large real-world ontology used widely in clinical applications and biomedical research. We include gene-to-phenotype annotations in the form of $\exists\text{anno}$.\text{Phenotype} (gene) to the ABox, and we use gene–gene interactions in BioGRID in the form of $\text{interacts}$(gene$_1$, gene$_2$) in the ABox. In this way, we can test whether \textsc{falcon} can help scientific fact discovery in the biomedical domain, e.g., gene–gene interaction prediction, by incorporating \textsc{alc}-based background knowledge.

Female $\sqcup$ Person, Female $\sqcap$ Female $\sqsubseteq \bot$, Parent $\sqsubseteq$ Person, Child $\sqsubseteq$ Person, Parent $\sqcap$ Child $\sqsubseteq \bot$, Father $\sqcap$ Male, Boy $\sqsubseteq$ Male, Father $\sqcap$ Boy $\sqsubseteq \bot$, Mother $\sqsubseteq$ Female, Girl $\sqsubseteq$ Female, Mother $\sqcap$ Girl $\sqsubseteq \bot$, Father $\sqcap$ Parent, Mother $\sqcap$ Parent, Father $\sqcap$ Mother $\sqsubseteq \bot$, Boy $\sqsubseteq$ Child, Girl $\sqcup$ Child, Boy $\sqsubseteq$ Girl $\sqsubseteq \bot$, Female $\sqcap$ Parent, Mother $\sqcap$ Parent, Female $\sqcap$ Father, Female $\sqcap$ Child $\sqcap$ Girl, Male $\sqcap$ Child $\sqsubseteq$ Boy, $\exists$hasChild.Person $\sqsubseteq$ Parent, $\exists$hasParent.Person $\sqsubseteq$ Child, Grandma $\sqsubseteq$ Mother

Note that we further add ABox axioms and create one named individual for each concept name, e.g., creating the individual \textit{A child} for concept name \textit{Child}.

Although these three datasets have unique use cases, they are all used to evaluate the semantic entailment performance of \textsc{falcon}. We evaluate Family with specially prepared axioms as Eqn. $14$,$17$. We apply HermiT (Shearer, Motik, and Horrocks 2008) on Pizza and apply ELK (Kazakov, Krötzsch, and Simančík 2014) on HPO to generate axioms that should be entailed, i.e., \textit{entailments}. Since automated reasoners do not directly output disproved axioms, we regard the axioms that are neither included in the original ontology nor entailed as unprovable axioms. Statistics of datasets can be found in Table 1 in Supplementary Materials.

### Implementation Details

We implement \textsc{falcon} using PyTorch and conduct all the experiments on a GNU/Linux server with Nvidia RTX 3090 GPU and Intel Xeon CPU. Since \textsc{alc} axioms with variable complexity causes different depths of neural networks and a resulting large memory consumption, we apply the gradient checkpointing in the training process. For semantic entailment performance evaluation, we report the Mean Absolute Error (MAE) between 1 and scores of entailed axioms predicted by \textsc{falcon}. We also report AUC, AUPR, and Fmax to evaluate how well \textsc{falcon} can distinguish entailed and non-entailed axioms.

For gene–gene interaction performance evaluation, we apply the filtered setting (Bordes et al. 2013) and report the Mean Reciprocal Rank (MRR), Hit Rate at 3 (H@3), and

\begin{align*}
\text{MRR} & = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{r_i} \\
\text{H@3} & = \frac{1}{n} \sum_{i=1}^{n} I(r_i \leq 3) \\
\text{H@10} & = \frac{1}{n} \sum_{i=1}^{n} I(r_i \leq 10)
\end{align*}

where $n$ denotes the number of negative samples for each positive sample in ABox, are tuned by grid searching within $\{e^{-2}, e^{-3}, e^{-4}\}$, $\{32, 64, 128\}$, and $\{4, 8, 16\}$, respectively. We set the number of models $k$ as a hyperparameter and report its value in the experiments. The choice of $t$-norms is a hyperparameter and includes the Gödel $t$-norm $\theta(x,y) = \min\{x,y\}$, the Product $t$-norm $\theta(x,y) = x\cdot y$, and the Łukasiewicz $t$-norm $\theta(x,y) = \max\{x+y-1,0\}$; while the Gödel and Product $t$-norms satisfy the conditions on $t$-norms stated in Theorem 1, the Łukasiewicz is not a strict $t$-norm and does not satisfy these conditions. We use the Product $t$-norm in all experiments. We sample anonymous individuals before each optimization step; consequently, we can potentially cover large parts of the embedding space as the number of optimization steps grows, even with a small number of anonymous individuals being sampled each time. In our experiments, we choose to sample two anonymous individuals based on named individuals with 0-mean 0.1-$\text{std}$ Gaussian noise, and another two anonymous individuals which are uniformly sampled from $\mathbb{R}^n$. The tuned hyperparameter settings are listed in Table 2 in Supplementary Materials.

### 6.2 Experimental Results

#### Semantic Entailment (RQ1)

We come up with representative axioms for evaluations on the \textbf{Family} Domain:

\begin{align*}
\text{Female } \sqsubseteq \text{ Child } \sqcup \text{ Girl,} \quad (14) \\
\exists\text{hasChild}\text{.Person } \sqcap \text{ Female } \sqsubseteq \text{ Mother,} \quad (15) \\
\text{Person } \sqcap \text{ Parent,} \quad (16) \\
\text{Mother } \sqsubseteq \text{ Grandma,} \quad (17)
\end{align*}

where Eqn. $14$ is an axiom that should be entailed by \textsc{falcon}; we expect the membership within the fuzzy sets generated from Eqn. $14$ to be 0 in all models. More intuitively, the element-wise $t$-norm of the fuzzy set (rows in Fig. $1$) of \textit{Female} and \textit{Child} is a subset of the fuzzy

![Figure 1: Learned degree of memberships of individuals in concepts in the Family Ontology in 2 models. Dark and light grids denote the learned degrees of membership are exactly 1 and 0, respectively. Anon $i$ denotes the randomly sampled anonymous individuals.](https://hpo.jax.org/app/download/annotation)

$\text{https://pytorch.org/docs/stable/checkpoint.html}^4$

$\text{https://downloads.thebiogrid.org/BioGRID}$

$\text{https://hpo.jax.org/app/download/annotation}$

$\text{https://downloads.thebiogrid.org/BioGRID}$
Table 1: Experimental results of paraconsistent reasoning with $k=1$ on Pizza. $N_{inc}$ denotes the number of inconsistent axioms we added in to the ontology.

| $N_{inc}$ | ALC | FALCON |
|-----------|-----|--------|
|           | Reasons | MAE | AUC | AUPR | Fmax |
| 0         | ✓   | 0.0181 | 0.8365 | 0.8804 | 0.8240 |
| 1         | ×   | 0.0186 | 0.8434 | 0.8669 | 0.8033 |
| 5         | ×   | 0.0161 | 0.8380 | 0.8849 | 0.8318 |
| 10        | ×   | 0.0158 | 0.8507 | 0.8777 | 0.8246 |
| 50        | ×   | 0.0914 | 0.7811 | 0.8121 | 0.7967 |
| 100       | ×   | 0.1016 | 0.7661 | 0.8075 | 0.7642 |
| 500       | ×   | 0.1516 | 0.7142 | 0.6614 | 0.7545 |
| 1000      | ×   | 0.2033 | 0.5524 | 0.5227 | 0.6667 |

Table 2: Experimental results of Gene-Gene Interaction and TBox Entailment on HPO.

| G-G Interaction | Semantic Entailment |
|-----------------|---------------------|
| MRR H@3 H@10 | MAE | AUC | AUPR | Fmax |
| DistMult | 9.6% | 8.1% | 14.7% | × | × | × | × | |
| TransE | 8.2% | 8.3% | 17.2% | × | × | × | × | |
| ConvKB | 8.6% | 7.9% | 17.3% | × | × | × | × | |
| FALCON | 10.1% | 9.7% | 20.0% | 0.024 | 0.805 | 0.847 | 0.752 | |

set of Girl in both models. Furthermore, FALCON is also capable of entailing more complex axioms like Eqn. 15. As shown in Fig. 1, Person is always a superclass of all other concepts, so the degree of memberships of Person$^{\sim}$–Parent will always contain 1-valued elements in all models. Thus Eqn. 16 is disproved by FALCON. Eqn. 17 is false in model 0 while true in model 1. Therefore, we can conclude that Eqn. 17 is unprovable under open world assumption. Therefore, FALCON is capable of handling entailed, disproved, and unprovable axioms under open world assumption, which empowers semantic entailment on the Family Ontology; as more models are generated (in particular when approaching infinity), the entailment will become (more) accurate.

As the results on Pizza show in the first row of Table 1, the MAE of computing axioms that should be entailed is very close to 0. Also, the capability of distinguishing entailments and unprovable axioms is ‘Excellent Discrimination’ according to the criterion summarized by (Hosmer Jr., Lemeshow, and Sturdivant 2013), demonstrating the effectiveness of FALCON in computing semantic entailments.

Well-established neural knowledge representation learning methods (Bordes et al. 2013, Yang et al. 2015, Dai Quoc Nguyen, Nguyen, and Phung 2018) are not able to perform semantic entailment because they are limited to triple-wise predictions without considering logical axioms. However, as shown in the last row of Table 2, FALCON is not only capable of computing entailments on large real-world ontologies such as HPO, but also able to distinguish between entailments and unprovable axioms under the open world assumption.

Parasistent Reasoning (RQ2) We add explicit contradictions to the Pizza Ontology to test entailment under inconsistency, i.e., paraconsistent reasoning. As shown in Table 1 with the introduction of a single or few inconsistent statements, symbolic ACC reasoners such as FaCT++ (Tsarkov and Horrocks 2006) and HermiT (Shearer, Motik, and Horrocks 2008) fail to compute useful entailments. However, FALCON can consistently compute entailments with low error and distinguish between entailments and unprovable axioms. Entailed axioms have additional support within the loss function and the parts of the models related to entailed axioms will degenerate more slowly than other parts; this difference allows FALCON to compute semantic entailments even in the presence of inconsistency. However, FALCON will eventually fail to distinguish between entailed and non-entailed statements when many contradictory statements are present (shown in the last a few rows of Table 1). This is also a direct consequence of the soundness and completeness of FALCON which mandates that every statement is entailed by an inconsistent ontology. We also evaluate the improvement provided by multi-model semantic entailment under inconsistency. The results are shown in Table 3 in the Supplementary Materials.

ACC Enhanced Prediction (RQ3) We use HPO to predict gene–gene interactions (GGIs) based on phenotypic relatedness of the genes. These can be expressed by ABox facts (triples) and thus knowledge graph completion (KGC) methods are naturally applicable. Representative methods include TransE (Bordes et al. 2013), DistMult (Yang et al. 2015), and ConvKB (Dai Quoc Nguyen, Nguyen, and Phung 2018). However, background knowledge on GGIs are expressed by ACC axioms and KGC methods can only deal with triples. To ensure fair comparison, we use as much data as baseline methods can use. That is, we use a subset of TBox axioms, i.e., $C \sqsubseteq D$ ($C$ and $D$ are limited to concept names), along with the ABox as the training data for KGC methods. FALCON use the same ABox and the whole TBox with all forms of ACC axioms ($C$ and $D$ can be concept descriptions), and prediction of GGIs is formulated as semantic entailment with number of models $k$ set to 1.

As the results shown in Table 2, FALCON can outperform all the baselines on all metrics, which clearly demonstrates that the added ACC axioms are informative and helpful for GGI prediction. Moreover, our results demonstrate that FALCON can effectively exploit and incorporate the information in complex ACC axioms to help machine learning in the biomedical domain.

7 Discussion

FALCON gives rise to two distinct forms of approximate entailment. Consider the example where we aim to entail $\text{Father} \sqsubseteq \text{Parent} \sqcap \text{Male}$ from $\text{Father} \sqsubseteq \text{Parent}$ in the absence of $\text{Father} \sqsubseteq \text{Male}$. This statement is not classically entailed, even if all named instances of $\text{Father}$ are also instances of $\text{Male}$ due to the open world assumption (implemented in FALCON by sampling random individuals from the embedding space). However, FALCON may still entail this statement.
The first type of approximate entailment is based on statements that are approximately true in models generated by FALCON. If we do not sample any anonymous individuals, the semantics of FALCON follows a closed-world semantics. In the example above, if all instances of Father are also instances of Male and no anonymous individuals are sampled, Father \sqsubseteq Male would be true in all models generated by FALCON. If anonymous individuals are sampled, the entailment may no longer be true in the classical sense (i.e., where the truth value is 1 within the generated models) but hold only approximately, based on the proportion of individuals that are instances of Male and those that are not. When generating multiple models and computing semantic entailment, we will use the min truth over all generated models as value of the approximate entailment of Father \sqsubseteq Male (or, more precisely, we will use the max of the degree of satisfiability of Father \sqcap \neg Male).

The second type of approximate entailment is more subtle and we have not explicitly explored it here. We compute semantic entailment of TBox axioms based on the maximum degree of satisfiability across all models generated by FALCON. However, we can also relax this condition and use the mean degree of satisfiability across multiple models, or use some other function that combines the degree of satisfiability in multiple models. In the above example, we can count the number of models in which Father \sqsubseteq Male is true, and determine the degree of entailment based on both the degree of truth of the statement within the models and the number of models. Such an approach would give rise to a different form of approximate entailment, and the interactions that can occur here are subject to future research.

8 Conclusion
To the best of our knowledge, we are the first to propose a neural ALC reasoner FALCON that is sound and complete for semantic entailment over ALC ontologies. Experimental results demonstrate that FALCON enables neural networks to perform semantic entailment and can be used to enhance scientific prediction tasks with background knowledge. Future works will include extensions to other DLs and potentially FOL with n-ary relations and applications in fields like bioinformatics where many expressive ontologies exist.

Acknowledgements
We would like to thank Stefan Borgwardt and Till Mossakowski for their comments on an earlier draft of this manuscript.

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, and FCC/1/1976-34-01. We acknowledge support from the KAUST Supercomputing Laboratory.
Advances in neural information processing systems.

Baader, F.; Calvanese, D.; McGuinness, D.; Patel-Schneider, P.; Nardi, D.; et al. 2003. *The description logic handbook: Theory, implementation and applications*. Cambridge university press.

Badreddine, S.; Garcez, A. d.; Serafini, L.; and Spranger, M. 2022. Logic tensor networks. *Artificial Intelligence*, 303: 103649.

Bienvenu, M. 2020. A short survey on inconsistency handling in ontology-mediated query answering. *KI-Künstliche Intelligenz*, 34(4): 443–451.

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Antoniou, G. 2008. Ontology change: classification and survey. *The Knowledge Engineering Review*, 23(2): 117–152.

Dai Quoc Nguyen, T. D. N.; Nguyen, D. Q.; and Phung, D. 2018. A Novel Embedding Model for Knowledge Base Completion Based on Convolutional Neural Network. In *Proceedings of NAACL-HLT*, 327–333.

Donini, F. M.; and Massacci, F. 2000. ExpTime tableaux for *ABox*. *Artificial Intelligence*, 124(1): 87–138.

Flouris, G.; Manakanatas, D.; Kondylakis, H.; Plexousakis, D.; and Antoniou, G. 2008. Ontology change: classification and survey. *The Knowledge Engineering Review*, 23(2): 117–152.

Hamilton, W.; Bajaj, P.; Zitnik, M.; Jurafsky, D.; and Leskovec, J. 2018. Embedding logical queries on knowledge graphs. *Advances in neural information processing systems*, 31.

Hornik, K.; Stinchcombe, M.; and White, H. 1989. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5): 359–366.

Hosmer Jr, D. W.; Lemeshow, S.; and Sturdivant, R. X. 2013. *Applied logistic regression*. John Wiley & Sons.

Kaminski, T.; Knorr, M.; and Leite, J. 2015. Efficient paraconsistent reasoning with ontologies and rules. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Kazakov, Y.; Krötzsch, M.; and Simančík, F. 2014. The incredible elk. *Journal of automated reasoning*, 53(1): 1–61.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kulmanov, M.; Liu-Wei, W.; Yan, Y.; and Hoehndorf, R. 2019. El embeddings: geometric construction of models for the description logic el++. *arXiv preprint arXiv:1902.10499*.

Köhler, S.; Gargano, M.; Matentzoglu, N.; Carmody, L. C.; Lewis-Smith, D.; Vasilevsky, N. A.; Danis, D.; Balagura, G.; Baynam, G.; Brower, A. M.; Callahan, T. J.; Chute, C. G.; Est, J. L.; Galer, P. D.; Ganesan, S.; Griese, M.; Haimel, M.; Pazmandi, J.; Hanauer, M.; Harris, N. L.; Hartnett, M.; Hastreiter, M.; Hauck, F.; He, Y.; Jeske, T.; Kearney, H.; Kindle, G.; Klein, C.; Knofflach, K.; Krause, R.; Lagorce, D.; McMurry, J. A.; Miller, J. A.; Munoz-Torres, M.; Peters, R. L.; Rapp, C. K.; Rath, A. M.; Rind, S. A.; Rosenberg, A.; Segal, M. M.; Seidel, M. G.; Smedley, D.; Talmy, T.; Thomas, Y.; Wiafe, S. A.; Xian, J.; Yüksel, Z.; Helbig, I.; Mungall, C. J.; Haendel, M. A.; and Robinson, P. N. 2020. The Human Phenotype Ontology in 2021. *Nucleic Acids Research*, 49(D1): D1207–D1217.

Luus, F.; Sen, P.; Kapanipathi, P.; Riegel, R.; Makondo, N.; Lebese, T.; and Gray, A. 2021. Logic embeddings for complex query answering. *arXiv preprint arXiv:2103.00418*.

Mondal, S.; Bhatia, S.; and Mutharaju, R. 2021. EmEL++: Embeddings for elL++ description logic. In *AAAI Spring Symposium on Combining Machine Learning and Knowledge Engineering*, CEUR-WS.

Özçep, Ö. L. t.; Leemhuis, M.; and Wolter, D. 2021. Core semantics for logics with negation. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, 1820–1826.

Peng, X.; Tang, Z.; Kulmanov, M.; Niu, K.; and Hoehndorf, R. 2022. Description Logic EL++ Embeddings with Intersectional Closure. *arXiv preprint arXiv:2202.14018*.

Reiter, R. 1981. On closed world data bases. In *Readings in artificial intelligence*, 119–140. Elsevier.

Ren, H.; Hu, W.; and Leskovec, J. 2020. Query2box: Reasoning Over Knowledge Graphs In Vector Space Using Box Embeddings. In *International Conference on Learning Representations (ICLR)*.

Ren, H.; and Leskovec, J. 2020. Beta embeddings for multiply logical reasoning in knowledge graphs. *Advances in Neural Information Processing Systems*, 33: 19716–19726.

Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*.

Schild, K. 1991. A correspondence theory for terminological logics: Preliminary report. In Mylopoulos, J.; and Reiter, R., eds., *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI 1991)*, 466–471. Morgan Kaufmann.

Schlobach, S.; Cornet, R.; et al. 2003. Non-standard reasoning services for the debugging of description logic terminologies. In *Ijcai*, volume 3, 355–362.

Schmidt-Schauß, M.; and Smolka, G. 1991. Attributive concept descriptions with complements. *Artificial intelligence*, 48(1): 1–26.

Shearer, R. D.; Motik, B.; and Horrocks, I. 2008. HermiT: A highly-efficient OWL reasoner. In *Owled*, volume 432, 91.

Smith, B.; Ashburner, M.; Rosse, C.; Bard, J.; Bug, W.; Ceusters, W.; Goldberg, L. J.; Eilbeck, K.; Ireland, A.; Mungall, C. J.; Leontis, N.; Serra, P. R.; Ruttenberg, A.; Sane, S. A.; Scheuermann, R. H.; Shah, N.; Whetzel, P. L.; and Lewis, S. 2007. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotech*, 25(11): 1251–1255.

Sun, H.; Arnold, A.; Bedrax Weiss, T.; Pereira, F.; and Cohen, W. W. 2020. Faithful embeddings for knowledge base queries. *Advances in Neural Information Processing Systems*, 33: 22505–22516.
Tang, Z.; Pei, S.; Peng, X.; Zhuang, F.; Zhang, X.; and Hoehndorf, R. 2022a. Joint Abductive and Inductive Neural Logical Reasoning. arXiv preprint arXiv:2205.14591.

Tang, Z.; Pei, S.; Zhang, Z.; Zhu, Y.; Zhuang, F.; Hoehndorf, R.; and Zhang, X. 2022b. Positive-Unlabeled Learning with Adversarial Data Augmentation for Knowledge Graph Completion. arXiv preprint arXiv:2205.00904.

Tsarkov, D.; and Horrocks, I. 2006. FaCT++ description logic reasoner: System description. In International joint conference on automated reasoning, 292–297. Springer.

van Krieken, E.; Acar, E.; and van Harmelen, F. 2020. Analyzing Differentiable Fuzzy Logic Operators. CoRR, abs/2002.06100.

van Krieken, E.; Acar, E.; and van Harmelen, F. 2022. Analyzing differentiable fuzzy logic operators. Artificial Intelligence, 302: 103602.

Wagner, B.; and d’Avila Garcez, A. 2022. Neural-Symbolic Reasoning Under Open-World and Closed-World Assumptions. In CEUR Workshop Proceedings, volume 3121. CEUR.

Yang, B.; Yih, S. W.-t.; He, X.; Gao, J.; and Deng, L. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In Proceedings of the International Conference on Learning Representations (ICLR) 2015.
Reproducibility

We provide the statistics of the datasets in Table 1 and list the tuned hyperparameter settings in Table 2. The detailed computation steps are shown in Algorithm 1 and our code.

Table 1: Statistics of datasets.

|                  | Family | Pizza | HPO |
|------------------|--------|-------|-----|
| Individual Symbols | 0      | 5     | 2504|
| Concept Symbols   | 10     | 99    | 30893|
| Relation Symbols  | 2      | 3     | 186 |
| Abox C(a) Train   | 0      | 5     | 93270|
| Abox R(a, b) Train| 0      | 0     | 9554 |
| Abox R(a, b) Test | N/A    | N/A   | 2400|
| Tbox Train        | 25     | 676   | 64416|
| Tbox Test         | N/A    | 115   | 2000|

Table 3: Table 2: Tuned hyperparameter settings.

|                  | Family | Pizza | HPO |
|------------------|--------|-------|-----|
| Number of models k | 100    | 1-20  | 1   |
| Contradictory axioms Ninc | 0      | 0-1000| N/A |
| Learning rate     | 1e-2   | 5e-3  | 1e-4|
| Embedding dimension | 50    | 50    | 128 |
| Negatives         | N/A    | N/A   | 8   |
| Batchsize R(a, b) | N/A    | N/A   | 64  |
| Batchsize C(a)    | N/A    | N/A   | 64  |
| Batchsize TBox    | 25     | 256   | 64  |
| Sampled individuals | 4     | 4     | 4   |
| Created ABox axioms | 10    | 99    | 1000|
| t-norm            | Product| Product| Product|

Additional Experiments

Table 4: Table 3: Reasoning performance (AUC) with 10 models and improvement of the multi-model semantic entailment described in Section 4.2 across increasing inconsistency on Pizza, where Avg denotes the average results (AUC) of independent single models and Multi denotes multi-model semantic entailment. Ninc denotes the number of explicitly added contradiction.

| Average inconsistency Ninc | 0 | 1 | 10 | 50 |
|---------------------------|---|---|----|----|
| Avg                       | 0.8418 | 0.8375 | 0.8500 | 0.7884 |
| Multi                     | 0.8543 | 0.8564 | 0.8706 | 0.8032 |
| Impr. (%)                 | 3.66% | 5.04% | 5.89% | 5.13% |

As shown in Table 3, with the introduction of inconsistency, the improvement of Multi over Avg gets more significant. It is because, as contradictory statements tend to create degenerate models (with empty concepts and relations), aggregating over multiple models generally performs better under inconsistency since some models still tend to preserve the memberships required for entailment.

We also evaluate the effectiveness of semantic entailment under different number of models. As the results show in Figure 2, we observe that the semantic entailment performance improves with a larger number of models and will eventually converge. Although the performance will fluctuate with fewer models, multi-model semantic entailment can consistently improve over the single-model case, i.e., Avg in Figure 2. Such results demonstrate the effectiveness of reasoning with multiple models as well as the rational of setting the number of models as a hyperparameter in practice.

Algorithm 1: Generating C^2 for a Concept Description C.

Function: Calculate m(·, C^2)

Require: Embedding function f_c;
Multilayer Perceptron MLP;
Activation function σ;
Sampling size k;
Fuzzy operators θ, κ, ν;
Individuals I = I_a ∪ I_b.

1: Sample X with |X| = k from I
2: Compute m(X, C^2) := \{m(x, C^2)| x ∈ X\}:
3: if C is a concept name then
4: m(X, C^2) = σ(MLP(f_c(x), f_e(x)))
5: else if C = C_1 ∩ C_2 then
6: m(X, (C_1 ∩ C_2)^2) = θ(m(X, C_1^2), m(x, C_2^2))
7: else if C = C_1 ∪ C_2 then
8: m(X, (C_1 ∪ C_2)^2) = κ(m(X, C_1^2), m(x, C_2^2))
9: else if C = ¬D then
10: m(X, (¬D)^2) = ν(m(X, D^2))
11: else if C = ∃R.D then
12: Sample Y with |Y| = k from I
13: m(X, (∃R.D)^2) = max_{y ∈ Y} θ(m(y, D^2), m((X, y), R^2))
14: with m((x, y), R^2) = σ(MLP(f_c(x) + f_e(R), f_e(y)))
15: else if C = ∀R.D then
16: Sample Y with |Y| = k from I
17: m(X, (∀R.D)^2) = min_{y ∈ Y} κ(ν(m(y, D^2), m((X, y), R^2))
18: with m((x, y), R^2) = σ(MLP(f_c(x) + f_e(R), f_e(y)))
19: end if
20: return m(X, C^2)

https://anonymous.4open.science/r/FALCON-7AD6