Language Model Analysis for Ontology Subsumption Inference

Yuan He[1], **Jiaoyan Chen**[2], **Ernesto Jiménez-Ruiz**[3,4],
Hang Dong[1], **Ian Horrocks**[1]
[1] University of Oxford, [2] The University of Manchester,
[3] City, University of London, [4] University of Oslo
{yuan.he,hang.dong,ian.horrocks}@cs.ox.ac.uk
jiaoyan.chen@manchester.ac.uk
ernesto.jimenez-ruiz@city.ac.uk

Abstract

Investigating whether pre-trained language models (LMs) can function as knowledge bases (KBs) has raised wide research interests recently. However, existing works focus on simple, triple-based, relational KBs, but omit more sophisticated, logic-based, conceptualised KBs such as OWL ontologies. To investigate an LM's knowledge of ontologies, we propose OntoLAMA, a set of inference-based probing tasks and datasets from ontology subsumption axioms involving both atomic and complex concepts[1]. We conduct extensive experiments on ontologies of different domains and scales, and our results demonstrate that LMs encode relatively less background knowledge of Subsumption Inference (SI) than traditional Natural Language Inference (NLI) but can improve on SI significantly when a small number of samples are given. We will open-source our code and datasets.[2]

1 Introduction

The advancements of large pre-trained language models (LMs) have sparked research interests in investigating how much explicit semantics LMs can learn or infer from knowledge bases (KBs) (AlKhamissi et al., 2022). The LAMA (LAnguage Model Analysis) probe (Petroni et al., 2019) is among the first works that adopt prompt-based methods to simulate the process of querying relational knowledge from various KBs such as ConceptNet (Speer and Havasi, 2012) and GoogleRE[3]. Some subsequent studies focus on probing specific types of knowledge from sources like commonsense KBs (Da et al., 2021), biomedical KBs (Sung et al., 2021), temporal KBs (Dhingra et al., 2022), and cross-lingual KBs (Liu et al., 2021a).

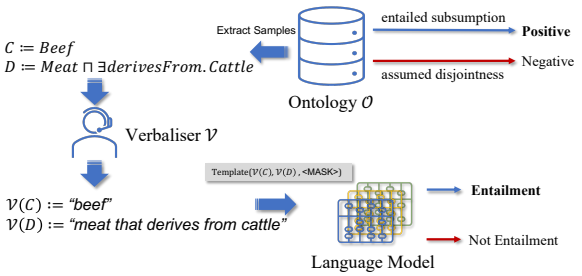

Figure 1: OntoLAMA framework.

However, existing "LMs-as-KBs" works focus on simple, triple-based, relational KBs, but neglect more formalised, logic-based, conceptualised KBs. For example, a statement like *"London is the capital of the UK"* can be expressed in the triple (London, capitalOf, UK); but a sentence like *"arthritis is a kind of arthropathy with an inflammatory morphology"*, which describes the concept *"arthritis"*, cannot be easily expressed using just triples. Conceptual knowledge like this requires a formal and expressive representation to be defined precisely. A well-known model for conceptual knowledge is the OWL[4] ontology (Bechhofer et al., 2004; Grau et al., 2008), which can be seen as a description logic (DL) KB with rich built-in vocabularies for knowledge representation and various reasoning tools supported. Taking the example of *"arthritis"*, in DL the concept can be described as $Arthritis \sqsubseteq Arthropathy \sqcap \exists hasMorphology.Inflammatory$.

In this work, we take a further step along the "LMs-as-KBs" research line towards more formalised semantics by targeting DL KBs and in particular the OWL ontologies. Current works on LMs concerning ontologies are mostly driven by a target application. Liu et al. (2020), He et al. (2022),

[1] An ontology *concept* is also known as a *class*. To avoid confusion with *class* in machine learning classification, we stick to use the term *concept*.

[2] Code and Instructions: https://krr-oxford.github.io/DeepOnto/ontolama; Dataset at HuggingFace: https://huggingface.co/datasets/krr-oxford/OntoLAMA/ or at Zenodo: https://doi.org/10.5281/zenodo.6480540

[3] https://code.google.com/archive/p/relation-extraction-corpus/

[4] For simplicity, we refer to the second edition OWL 2 as OWL: https://www.w3.org/TR/owl2-overview/

Findings of the Association for Computational Linguistics: ACL 2023, pages 3439–3453
July 9-14, 2023

and Chen et al. (2022) apply language model fine-tuning to address ontology curation tasks such as concept insertion and matching, while Ye et al. (2022) transform ontologies into graphs for data augmentation in few-shot learning. In contrast to these application-driven approaches, we investigate a more fundamental question: *To what extent can LMs infer conceptual knowledge modelled by an ontology?* Particularly, we focus on the subsumption relationships between ontology concepts. As shown in Figure 1, we first extract concept pairs (C, D) that are deemed as positive (C and D are in a subsumption relationship) and negative (C and D are assumed to be disjoint) samples from an ontology. Note that the sampling procedure is fully automatic with the syntax and semantics of OWL ontology carefully considered. To translate the concepts and especially the ones with complex logical expressions into natural language texts, we develop a recursive concept verbaliser. We formulate the Subsumption Inference (SI) task similarly to the Natural Language Inference (NLI) task and treat the concept pairs as premise-hypothesis pairs (Padó and Dagan, 2022), which will then be wrapped into a template for generating inputs of LMs.

We have created SI datasets from ontologies of various domains and scales, and conducted extensive experiments. Our results demonstrate that LMs perform better on a typical NLI task than the constructed SI tasks under the zero-shot setting, indicating that LMs encode relatively less background knowledge of ontology subsumptions. However, by providing a small number of samples (K-shot settings), the performance on SI is significantly improved. This observation is consistent with the three LMs that are studied in this work.

2 Background

2.1 OWL Ontology

An OWL ontology is a description logic (DL) knowledge base that consists of the TBox (terminological), ABox (assertional), and RBox (relational) axioms (Krötzsch et al., 2012). In this work, we focus on the TBox axioms which specify the subsumption relationships between concepts of a domain. A subsumption axiom has the form of $C \sqsubseteq D$ where C and D are concept expressions involving atomic concept, negation ($\neg$), conjunction ($\sqcap$), disjunction ($\sqcup$), existential restriction ($\exists r.C$), universal restriction ($\forall r.C$), and so on (see complete definition in Appendix A). An **atomic concept** is a named concept, a top concept $\top$ (a concept with every individual as an instance), or a bottom concept $\bot$ (an empty concept); while a **complex concept** consists of at least one of the available logical operators. An equivalence axiom $C \equiv D$ is equivalent to $C \sqsubseteq D$ and $D \sqsubseteq C$.

Regarding the semantics, in DL we define an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ that consists of an non-empty set $\Delta^{\mathcal{I}}$ and a function $\cdot^{\mathcal{I}}$ that maps each concept C to $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each *property* r to $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. We say $\mathcal{I}$ is a model of $C \sqsubseteq D$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds, and $\mathcal{I}$ is a model of an ontology $\mathcal{O}$ if $\mathcal{I}$ is a model of all axioms in $\mathcal{O}$. If $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds for every model $\mathcal{I}$ of $\mathcal{O}$, then we can say $\mathcal{O} \models C \sqsubseteq D$. This defines logical entailment w.r.t. an ontology and it is more strictly defined than textual entailment based on human beliefs.

An individual a is an instance of a concept C in $\mathcal{O}$ if $\mathcal{O} \models C(a)$ ($a^{\mathcal{I}} \in C^{\mathcal{I}}$ for every model $\mathcal{I}$ of $\mathcal{O}$). C and D are *disjoint* in $\mathcal{O}$ if $O \models C \sqcap D \sqsubseteq \bot$ (or equivalently $O \models C \sqsubseteq \neg D$) which means there can be no common instance a of C and D.

The Open World Assumption (OWA) underpins OWL ontologies, according to which we cannot say what is not entailed by the ontology is necessarily false. For example, if we have an ontology that contains just one axiom $Paella \sqsubseteq \exists hasIngredient.Chicken$, in OWA we cannot determine if paella can have chorizo as an ingredient or not. To allow reuse and extension, ontologies are often (intentionally) underspecified (Cimiano and Reyle, 2003); this characteristic motivates how we define the negative samples in Section 3.1.

2.2 Related Work

Recently, the rise of the prompt learning paradigm has shed light on better usage of pre-trained LMs without, or with minor, supervision (Liu et al., 2022). However, LMs are typically pre-trained in a stochastic manner, making it challenging to study what knowledge LMs have implicitly encoded (Petroni et al., 2019) and how to access LMs in an optimal or controllable way (Gao et al., 2021; Li et al., 2022).

Our work is informed by the "LMs-as-KBs" literature (AlKhamissi et al., 2022), where different probes have been designed to test LMs' knowledge of relational data. In Petroni et al. (2019), the probing task of world knowledge has been formulated as a cloze-style answering task where LMs are required to fill in the `<MASK>` token given in-

put texts wrapped into a manually designed template. Sung et al. (2021) did a similar work but shift the focus to (biomedical) domain knowledge of domain-specific LMs. Liu et al. (2021a) pretrained LMs with multi-lingual knowledge graphs (KGs) and test on the cross-lingual tasks. Dhingra et al. (2022) proposed datasets with temporal signals and probed LMs on them with templates generated by the text-to-text transformer T5 (Raffel et al., 2022).

However, existing "LMs-as-KBs" works mostly focus on relational facts, but omit logical semantics and conceptual knowledge. In contrast, our work focuses on OWL ontologies which represent conceptual knowledge with an underlying logical formalism. Although there are some recent works concerning both LMs and ontologies, they do not compare them at the semantic level but rather emphasise on downstream applications. For example, He et al. (2022) adopted LMs as synonym classifiers to predict mappings between ontologies; whereas Ye et al. (2022) used ontologies to provide extra contexts to help LMs to make predictions.

3 Subsumption Inference

3.1 Task Definition

Recall the definitions in Section 2.1, a subsumption axiom $C \sqsubseteq D$ can be interpreted as: *"every instance of C is an instance of D"*. We can accordingly form a premise-hypothesis pair where the *premise* is *"x is a C"* and the *hypothesis* is *"x is a D"* for some individual x. Note that there are different ways to express the premise and hypothesis, and we adopt a simple but effective one (see Section 5.1). Next, an *ontology verbaliser* is required for transforming the concept expressions C and D into natural language texts. Analogous to Natural Language Inference (NLI) or Recognising Textual Entailment (RTE) (Poliak, 2020; Padó and Dagan, 2022), the task of Subsumption Inference (SI) is thus defined as *classifying if the premise entails or does not entail the hypothesis*. Note that SI is similar to a two-way RTE task[5] where we do not consider the *neutral*[6] class.

Given an ontology $\mathcal{O}$, we extract positive and negative subsumptions to probe LMs. The positive samples are concept pairs (C, D) with $\mathcal{O} \models C \sqsubseteq D$. Due to OWA, we cannot determine if (C, D) with $\mathcal{O} \not\models C \sqsubseteq D$ really forms a negative subsumption (see Appendix F for more explanation); to generate plausible negative samples, we propose the assumed disjointness[7] defined as follows:

Definition (Assumed Disjointness). *If two concepts C and D are satisfiable in $\mathcal{O} \cup \{C \sqcap D \sqsubseteq \bot\}$ and there is no named atomic concept A in $\mathcal{O}$ such that $\mathcal{O} \models A \sqsubseteq C$ and $\mathcal{O} \models A \sqsubseteq D$, then C and D are assumed to be disjoint.*

The first condition ensures that C and D are still **satisfiable** after adding the disjointness axiom for them into $\mathcal{O}$ whereas the second condition ensures that C and D have **no common descendants** because otherwise the disjointness axiom will make any common descendant unsatisfiable. If two concepts C and D satisfy these two conditions, we treat (C, D) as a valid negative subsumption.

However, in practice validating the satisfiability for each concept pair (C, D) would be inefficient especially when the ontology is large and complex. Thus, we propose a pragmatical alternative to the satisfiability check in Appendix E.

To conduct reasoning to extract entailed positive subsumptions and validate sampled negative subsumptions, we need to adopt a proven sound and complete OWL reasoner, e.g., HermiT (Glimm et al., 2014).

In the following sub-sections, we propose two specific SI tasks and their respective subsumption sampling methods.

3.2 Atomic Subsumption Inference

The first task aims at subsumption axioms that involve just *named atomic concepts*. Such axioms are usually the most prevalent in an ontology and can be easily verbalised by using the concept names. In this work, we use labels (in English) defined by the built-in annotation property `rdfs:label` as concept names.

The positive samples are extracted from all entailed subsumption axioms of the target ontology. We consider two types of negative samples: *(i)* **soft negative** composed of two random concepts, and *(ii)* **hard negative** composed of two random *sibling* concepts. Two sibling concepts lead to a "hard" negative sample because they share a common parent (thus having closer semantics) but are

[5] RTE guidelines: https://tac.nist.gov/2008/rte/rte.08.guidelines.html.

[6] *Neutral* essentially means two terms are unrelated. Ontologies are invariably underspecified, so even if two concepts have not been entailed as a subsumption or non-subsumption, they may still be implicitly related in the real world.

[7] Schlobach (2005) and Solimando et al. (2017) defined a similar assumption but in different contexts.

Pattern	Verbalisation ($\mathcal{V}$)
A (atomic)	the name (`rdfs:label`) of A
r (property)	the name (`rdfs:label`) of r, subject to rules in Appendix C
$\neg C$	*"not $\mathcal{V}(C)$"*
$\exists r.C$	*"something that $\mathcal{V}(r)$ some $\mathcal{V}(C)$"*
$\forall r.C$	*"something that $\mathcal{V}(r)$ only $\mathcal{V}(C)$"*
$C_1 \sqcap ... \sqcap C_n$	if $C_i = \exists/\forall r.D_i$ and $C_j = \exists/\forall r.D_j$, they will be re-written into $\exists/\forall r.(D_i \sqcap D_j)$ before verbalisation; suppose after re-writing the new expression is $C_1 \sqcap ... \sqcap C_{n'}$ **(a)** if **all** C_is (for $i = 1, ..., n'$) are restrictions, in the form of $\exists/\forall r_i.D_i$: *"something that $\mathcal{V}(r_1)$ some/only $V(D_1)$ and ... and $\mathcal{V}(r_{n'})$ some/only $V(D_{n'})$"* **(b)** if **some** C_is (for $i = m+1, ..., n'$) are restrictions, in the form of $\exists/\forall r_i.D_i$: *"$\mathcal{V}(C_1)$ and ... and $\mathcal{V}(C_m)$ that $\mathcal{V}(r_{m+1})$ some/only $V(D_{m+1})$ and ... and $\mathcal{V}(r_{n'})$ some/only $V(D_{n'})$"* **(c)** if **no** C_i is a restriction: *"$\mathcal{V}(C_1)$ and ... and $\mathcal{V}(C_{n'})$"*
$C_1 \sqcup ... \sqcup C_n$	similar to verbalising $C_1 \sqcap ... \sqcap C_n$ except that *"and"* is replaced by *"or"* and case **(b)** uses the same verbalisation as case **(c)**

Table 1: Recursive rules for verbalising a complex concept expression C in OWL ontologies. Note that C_i in the conjunction/disjunction pattern is also an arbitrary complex concept.

often disjoint. The sampled pairs need to meet the assumed disjointness defined in Section 3.1 to be accepted as valid negatives. We first sample equal numbers of soft and hard negatives and then randomly truncate the resulting set into the size of the positive sample set to keep class balance.

3.3 Complex Subsumption Inference

In the second SI task, we consider subsumption axioms that involve *complex concepts*. Particularly, we choose equivalence axioms of the form $A \equiv C$[8] (where A and C are atomic and complex concepts, respectively) as anchors, and equivalently transform them into subsumption axioms of the forms $A \sqsubseteq C$ and $C \sqsubseteq A$, through which complex concepts can appear on both the premise and hypothesis sides.

Recursive Concept Verbaliser To transform a complex C into a natural language text, we develop the *recursive concept verbaliser* consisting of a *syntax tree parser* and a set of *recursive rules* (see Table 1). A concrete example is shown in

[8]Equivalence axioms of this form are referred to as the *definition* of the named concept, and are common in OWL.

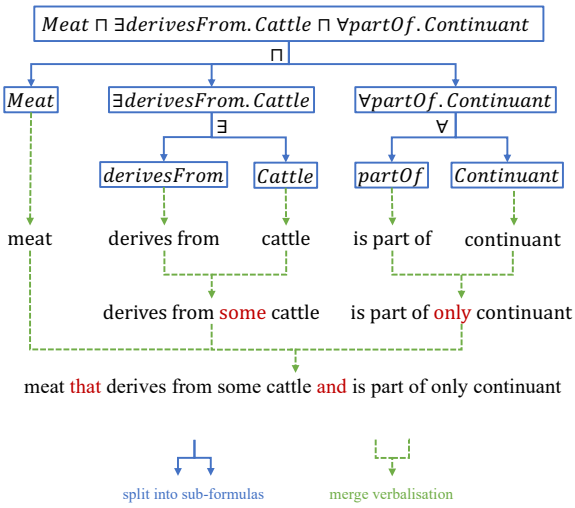

Figure 2: Illustration of how the recursive concept verbaliser is applied to an example complex concept expression. The algorithm first splits the complex concept into a sub-formula tree, verbalising the leaf nodes, and then merging the verbalised sub-formulas recursively. The key word associated with the logical operator at each merging step is marked in red. See Appendix D for more examples.

Figure 2, where the complex concept $Meat \sqcap \exists derivesFrom.Cattle \sqcap \forall partOf.Continuant$ is first split into a sub-formula tree by the syntax parser, then verbalised according to the recursive rules in Table 1. The leaf nodes are either atomic concepts or properties and they are verbalised by their names. At each recursive step, verbalised child nodes are merged according to the logical pattern in their parent node. Note that we enforce some extent of redundancy removal for the conjunction ($\sqcap$) and the disjunction ($\sqcup$) patterns. Taking the example in Figure 2, the verbalised atomic concept *"meat"* is placed before *"that"* as an *antecedent*, and the verbalised conjunction of two restrictions is placed after *"that"* as a *relative clause*. *"meat"* can be replaced by *"something"* if the concept $Meat$ is not involved. Moreover, if two restrictions with the same quantifier and property are connected by $\sqcap$ or $\sqcup$, they will be merged into one restriction. For example, $\exists derivesFrom.Cattle \sqcap \exists derivesFrom.Sheep$ will be transformed into $\exists derivesFrom.(Cattle \sqcap Sheep)$.

We extract equivalence axioms in the form of $A \equiv C$ from the target ontology. Taking each such axiom as an anchor, we can obtain positive complex subsumption axioms of the form $A_{sub} \sqsubseteq C$ or $C \sqsubseteq A_{super}$ where A_{sub} and A_{super} are a sub-class and a super-class of A, respectively. To derive challenging negative samples, we first randomly re-

Source	#Concepts	#EquivAxioms	#Dataset (Train/Dev/Test)
Schema.org	894	-	Atomic SI: $808/404/2,830$
DOID	$11,157$	-	Atomic SI: $90,500/11,312/11,314$
FoodOn	$30,995$	$2,383$	Atomic SI: $768,486/96,060/96,062$ Complex SI: $3,754/1,850/13,080$
GO	$43,303$	$11,456$	Atomic SI: $772,870/96,608/96,610$ Complex SI: $72,318/9,040/9,040$
MNLI	-	-	biMNLI: $235,622/26,180/12,906$

Table 2: Statistics for ontologies, SI datasets, and the biMNLI dataset.

place a named concept or a property in $A \equiv C$ to generate either *(i)* $A' \equiv C$ (if A is replaced by A') or *(ii)* $A \equiv C'$ (if C is corrupted). Without loss of generality, we assume the random replacement leads to case *(ii)*. We then check if A and C' satisfy the assumed disjointness as described in Section 3.1. In the affirmative case, we can have either $A \sqsubseteq C'$ or $C' \sqsubseteq A$ as the final negative subsumption; otherwise, we skip this sample. For example, given $SunflowerSeed \equiv Seed \sqcap \exists DerivesFrom.HelianthusAnnuus$, a possible negative subsumption is $SunflowerSeed \sqsubseteq Fruit \sqcap \exists DerivesFrom.HelianthusAnnuus$ if $Seed$ in C is replaced by $Fruit$ to create C'.

4 Datasets

In this work, we consider ontologies of different domains and scales including:

- Schema.org[9] (released on 2022-03-17): a general-purpose ontology that maintains a basic schema for structured data on the Web;
- DOID[10] (released on 2022-09-29): an ontology for human diseases (Schriml et al., 2012);
- FoodOn[11] (released on 2022-08-12): an ontology specialised in food-related knowledge including food products, food sources, food nutrition, and so on (Dooley et al., 2018).
- GO[12] (released on 2022-11-03): a very fine-grained and widely used biomedical ontology specialised in genes and gene functions (Ashburner et al., 2000).

We used the most updated versions at the time of experiment. The details for pre-processing the ontologies are illustrated in Appendix B.

[9] https://schema.org/
[10] https://disease-ontology.org/
[11] https://foodon.org/
[12] http://geneontology.org/

We construct an Atomic SI dataset for each ontology, but Complex SI datasets are created for FoodOn and GO only, due to their abundance of equivalence axioms. To avoid too many repetitive concept expressions brought by a particular equivalence axiom, we sample at most 4 positive and 4 negative samples for each equivalence axiom in the Complex SI setting. To attain class balance, we purposely keep the number of negative samples the same as the positive samples in each data split. For most of the resulting datasets, we divide each into $8:1:1$ for training, development, and testing; for the Schema.org's Atomic SI and the FoodOn's Complex SI datasets, which are relatively smaller, we apply a $2:1:7$ division instead. Note that we mainly focus on K-shot settings in the probing study, thus the required training and development sample sets are small.

To compare with how LMs perform on traditional NLI, we additionally create biMNLI, a subset of the Multi-Genre Natural Language Inference (MNLI) corpus (Williams et al., 2018) where *(i)* the neutral class and its samples are removed, *(ii)* the Matched and Mismatched testing sets are merged into one testing set, *(iii)* 10% of the training data is used as the development set, and *(iv)* the entailment-contradiction ratio is set to $1:1$ (by discarding extra samples from the dominant class) for a balanced prior. The numbers of named concepts and equivalence axioms in ontologies, and the numbers of samples in (each split of) SI datasets and the biMNLI dataset are reported in Table 2.

5 Experiments

5.1 Prompt-based Inference

To conduct the inference task under the prompt-based settings, we wrap the verbalised subsumption axioms and the <MASK> token into a template to

serve as inputs of LMs. We opt to use different combinations of manually designed templates[13] (T_1 and T_2) and label words (L_1 to L_3) that have achieved promising results on the NLI tasks (Schick and Schütze, 2021; Gao et al., 2021) as follows:

$$T_1 := \underbrace{\text{It is <A> } \mathcal{V}(C)}_{\text{premise}} \text{ ? <MASK>, } \underbrace{\text{it is <A> } \mathcal{V}(D)}_{\text{hypothesis}} .$$

$$T_2 := \text{“}\underbrace{\text{It is <A> } \mathcal{V}(C)}_{\text{premise}}\text{”? <MASK>, “}\underbrace{\text{it is <A> } \mathcal{V}(D)}_{\text{hypothesis}}\text{”.}$$

$L_1 :=$ {“positive”: [“Yes”], “negative”: [“No”]}
$L_2 :=$ {“positive”: [“Right”], “negative”: [“Wrong”]}
$L_3 :=$ {“positive”: [“Yes”, “Right”],
“negative”: [“No”, “Wrong”]}

where <A> is *“a”*, *“an”*, or just *blank* depending on the next word[14], $\mathcal{V}(\cdot)$ is the concept verbalisation function defined in Section 3, and <MASK> is the token that LMs need to predict. The probability of predicting class y (“positive” or “negative”) for an input sample $x = (C, D)$ is defined as:

$$\begin{aligned} P(y \mid x) &= P(\texttt{<MASK>} \in L_j[y]) \mid T_i(C, D)) \\ &= \frac{\sum_{v \in L_j[y]} \exp(\mathbf{w}_v \cdot \mathbf{h}_{\texttt{<MASK>}})}{\sum_{w \in L_j[\cdot]} \exp(\mathbf{w}_w \cdot \mathbf{h}_{\texttt{<MASK>}})} \end{aligned}$$

where $L_j[\cdot]$ and $L_j[y]$ denote all the label words defined in L_j and the label words of class y defined in L_j, respectively; $T_i(C, D)$ denotes the transformed texts of concepts C and D through the template T_i; $\mathbf{w}_v$ and $\mathbf{w}_w$ are vectors for the label words v and w, respectively; and $\mathbf{h}_{\texttt{<MASK>}}$ denotes the hidden vector of the masked token. The prediction can be trained by minimising the cross-entropy loss.

For the biMNLI dataset, the premise and hypothesis are replaced by what were originally given in the dataset – except that we have removed trailing punctuations.

In the main experiments concerning language models, we consider all the combinations of T_i and L_j and additionally consider 3 random seeds (thus 18 experiments each) for K-shot settings where $K > 0$. The value of K refers to the number of samples per classification label (positive or negative) we randomly extract from training and development sets, respectively. For $K = 0$ (zero-shot), different random seeds do not affect the results. For the fully supervised setting, we consider only one random seed and one combination (T_1 and L_1) because our pilot experiments demonstrate that fine-tuning on large samples results in low variance brought by different random seeds and different combinations of templates and label words.

Our code implementations mainly rely on `The OWL API`[15] for ontology processing and reasoning, and `OpenPrompt`[16] for prompt learning (Ding et al., 2022). Training of each K-shot (where $K > 0$) experiment takes 10 epochs, while for the fully supervised setting involving very large training samples, we only train for 1 epoch.[17] The best-performing model on the development set (at each epoch) is selected for testing set inference. We use the AdamW optimiser (Loshchilov and Hutter, 2019) with the initial learning rate, weight decay, and the number of warm-up steps set to 10^{-5}, 10^{-2}, and 50, respectively. All our experiments are conducted on two `Quadro RTX 8000` GPUs.

5.2 Results and Analysis

LMs and Settings We choose LMs from the RoBERTa family (Liu et al., 2019) as they are frequently introduced in cloze-style probing tasks (Liu et al., 2021b; Sung et al., 2021; Kavumba et al., 2022). In Table 3, we present key experiment results for roberta-large and roberta-base; we have a further ablation study for a biomedical variant of roberta-large in the latter paragraph.

For both LMs in Table 3, we report results of K-shot settings with $K \in \{0, 4, 32, 128\}$. We additionally present the results of the fully supervised setting for roberta-large as the oracle. For each setting, we report the averaged accuracy and standard deviation (where applicable). To clearly observe how the performance varies as K increases, we present Figure 3 which visualises the K-shot results for roberta-large with additional values of K ($\{8, 16, 64\}$). The complete result table for both language models and the figure that visualises the performance of roberta-base are available in Appendix G.

Baselines As aforementioned, we purposely keep class balance in each data split, thus the accuracy scores for majority vote are all 50.0%. Besides,

[13] We make slight modifications by adding the prefix “It/it is <A>” to make premise and hypothesis sentences complete.

[14] “an” is used when the next word starts with a vowel; leaving it blank when the next word is *“something”*.

[15] https://owlapi.sourceforge.net/

[16] https://thunlp.github.io/OpenPrompt/

[17] Since `Schema.org`'s Atomic SI and `FoodOn`'s Complex SI datasets have a small training set, their fully supervised settings still take 10 epochs.

		Atomic SI				Complex SI	
Setting	**biMNLI**	**Schema.org**	**DOID**	**FoodOn**	**GO**	**FoodOn**	**GO**
majority	50.0	50.0	50.0	50.0	50.0	50.0	50.0
word2vec							
K=4	51.5 (0.2)	54.9 (2.9)	64.6 (2.6)	63.5 (1.0)	60.1 (4.1)	56.8 (4.2)	56.9 (5.4)
K=128	52.1 (0.4)	73.0 (0.4)	70.8 (1.7)	71.4 (1.0)	66.3 (0.9)	63.8 (0.6)	66.4 (1.3)
roberta-base							
K=0	62.5 (6.5)	56.4 (3.6)	53.3 (4.0)	54.6 (4.4)	49.0 (2.4)	55.9 (3.6)	48.7 (3.1)
K=4	67.6 (5.2)	62.9 (5.2)	61.8 (6.7)	62.1 (4.2)	65.2 (5.0)	62.4 (3.2)	52.2 (7.1)
K=32	78.8 (1.1)	84.3 (2.0)	89.0 (1.4)	85.0 (1.1)	84.6 (2.5)	77.0 (1.5)	76.4 (2.5)
K=128	85.1 (1.0)	91.1 (0.7)	92.4 (0.7)	90.0 (0.7)	89.0 (0.8)	85.5 (1.3)	86.9 (1.5)
roberta-large							
K=0	68.7 (6.2)	61.7 (7.2)	59.8 (5.4)	60.1 (8.8)	54.6 (1.9)	56.1 (1.9)	50.4 (0.6)
K=4	78.1 (6.6)	69.4 (5.4)	74.0 (5.5)	71.6 (4.4)	67.6 (3.4)	64.1 (5.1)	56.9 (5.7)
K=32	89.9 (1.2)	87.3 (1.9)	92.3 (0.7)	88.9 (1.6)	87.7 (1.6)	80.8 (3.8)	81.6 (2.2)
K=128	93.0 (0.8)	92.9 (0.8)	93.4 (0.5)	92.2 (0.5)	91.0 (0.7)	88.4 (1.1)	90.2 (1.0)
full	97.5	95.4	97.8	98.7	98.1	95.8	98.8

Table 3: Results for the biMNLI, Atomic SI, and Complex SI tasks with each cell stating *"mean accuracy (standard deviation)"* except for majority vote and the fully supervised settings where standard deviation is not available.

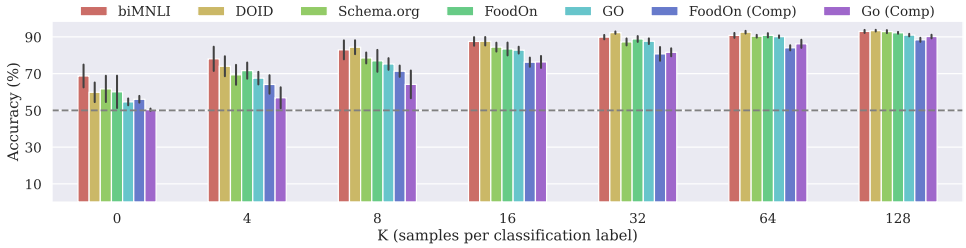

Figure 3: Visualisation of K-shot results (for roberta-large) on the biMNLI, Atomic SI, and Complex SI tasks, where the dotted horizontal line indicates majority vote. The order of the bars is the same as in the legend.

we consider word2vec (Mikolov et al., 2013) pre-trained on GoogleNews[18] with a logistic regression classifer as a baseline model, which demonstrates how a classic non-contextual word embedding model performs on the SI tasks. For this baseline, we only report results for $K \in \{4, 128\}$ as the increase of K does not bring significant change and results of $K = 128$ are roughly comparable to results of $K = 4$ for roberta-large. This suggests that the SI sample patterns are not easily captured with word2vec.

SI vs biMNLI From the results, we first observe that both roberta-large and roberta-base achieve better zero-shot results on biMNLI than on all the SI datasets by at least 7.0% and 6.1% respectively, showing that under our prompt settings, both LMs encode better background knowledge on biMNLI than SI. However, as K grows, the performances on both biMNLI and SI improve consistently and significantly (while the standard deviation generally reduces), and we can see at $K = 32$, the mean accuracy scores on the Atomic SI tasks have surpassed biMNLI for roberta-base. At $K = 64$ (see Figure 3), the mean accuracy scores on biMNLI and all the Atomic SI tasks converge to around 90.0%; the scores on the two Complex SI tasks are also above 80.0% for both LMs. Moreover, roberta-large consistently attains a better score than roberta-base for every setting.

Comparison Among SI Tasks We observe that Complex SI is generally harder than Atomic SI. For example, at $K = 0$, roberta-large attains 50.4% almost as majority vote on the Complex SI dataset of GO; at $K = 128$, roberta-large attains 88.4% on the Complex SI dataset of FoodOn while it attains more than 90% for the others. We can also observe

[18] https://code.google.com/archive/p/word2vec/

K	DOID	GO	GO (Comp)
0	49.7 (0.4)	50.1 (0.2)	50.0 (0.0)
4	64.8 (7.9)	66.2 (6.5)	50.0 (0.7)
32	94.7 (1.3)	93.5 (1.1)	73.5 (3.6)
128	96.3 (0.4)	95.2 (0.5)	90.5 (1.8)

Table 4: Results for roberta-large-pm-m3-voc on SI tasks of biomedical ontologies DOID and GO.

from Figure 3 that the scores on Complex SI tasks are generally lower than those on the Atomic SI tasks. Among the Atomic SI tasks, we find that GO is the most challenging which is as expected because GO is a fine-grained expert-level ontology. However, it surprises us that at $K = 32$ the score (92.3%) on DOID is better than all other tasks, considering that DOID is a domain-specific ontology.

Domain-specific SI We conduct further experiments for domain-specific LMs on domain-specific SI tasks. Specifically, we consider the variant roberta-large-pm-m3-voc which has been pre-trained on biomedical corpora PubMed abstracts, PMC full-text, and MIMIC-III clinical notes with an updated sub-word vocabulary learnt from PubMed (Lewis et al., 2020). In Table 4, we present the K-shot results of roberta-large-pm-m3-voc on three SI tasks related to biomedical ontologies DOID and GO. The zero-shot scores are almost equivalent to majority vote but the performance improves more prominently than roberta-large on the Atomic SI tasks of DOID and GO as K increases. Surprisingly, the Complex SI setting of GO seems to be quite challenging to this biomedical variant of RoBERTa. For example, at $K = 4$, the score is not improved compared to $K = 0$.

Template and Label Words The access to LMs is an influential factor of performance especially when there are no or fewer training samples. For example, roberta-large attains a standard deviation of 8.8% for $K = 0$ on FoodOn's Atomic SI task, suggesting that there is a significant performance fluctuation brought by different combinations of templates and label words. Although the standard deviation on GO's Complex SI is just 0.6%, the corresponding accuracy score (50.4%) indicates that none of these combinations work. Furthermore, effective template or label words are not transferable from one LM to another, as we can observe from the bad performance of roberta-large-pm-m3-voc for $K = 0$ on the SI tasks of biomedical ontologies. These observations suggest that either we did not find a generalised template and label words combination, or LMs require customised access for different types of knowledge.

6 Conclusion and Discussion

As a work that introduces ontologies to the "LMs-as-KBs" collection, this paper emphasises on how to establish a meaningful adaptation from logical expressions to natural language expressions, following their formal semantics. To this end, we leverage the Natural Language Inference (NLI) setting to define the Subsumption Inference (SI) task with careful considerations to address the differences between textual entailment and logical entailment. We also develop the recursive concept verbaliser for OWL ontologies as an auxiliary tool. Our results demonstrate that with our SI set-ups, LMs can successfully learn to infer both atomic and complex subsumptions when a small number of annotated samples are provided. This paves the way for investigating more complex reasoning tasks with LMs or guiding LMs using ontology semantics with limited training.

In fact, the current SI setting is not the only way for probing subsumption knowledge of an ontology; for example we can directly verbalise $C \sqsubseteq D$ as *"$\mathcal{V}(C)$ is a kind of $\mathcal{V}(D)$"* and formulate the probing task similar to fact-checking or equivalently, an inference task with empty premises. However, our pilot experiments demonstrate that such setting is not as effective as the current SI setting.

The presented work brings opportunities for future work as *(i)* the proposed ontology verbalisation method has not covered all possible patterns of complex concepts (e.g., with cardinality restrictions and nominals); *(ii)* we have not fully considered textual information such as synonyms, definitions, and comments, that are potentially available in an ontology; *(iii)* we have considered only TBox (terminological) axioms, but ABox (assertional) axioms can be involved in, e.g., the membership prediction task, where the objective is to classify which concept an individual belongs to. Therefore, developing a robust tool for verbalising logical expressions and extending the ontology inference settings are potential next tasks. Another interesting line for the near future is to train an LM using ontologies with their logical semantics considered. The resulting LM is expected to be applicable to different downstream ontology curation tasks such as ontology matching and entity linking, with fewer samples necessary for fine-tuning.

Limitations

As we mainly focus on conceptual knowledge captured in so-called TBox (terminological) axioms, the ABox (assertional) axioms are not considered. ABox axioms can capture situations for specific individuals (e.g., health status of a person) which could cause privacy issue and we would not expect LMs to capture such knowledge. Hence, dealing with ABox axioms could require additional engineering for data preprocessing.

Ethical Considerations

In this work, we construct new datasets for the proposed Subsumption Inference (SI) task from publicly available ontologies: `Schema.org`, `DOID`, `FoodOn`, and `GO`, with their download links specified in Section 4. The biMNLI dataset is constructed from the existing open-source MNLI dataset. We have confirmed that there is no privacy or license issue in all these datasets.

Acknowledgements

This work was supported by Samsung Research UK (SRUK), and the EPSRC projects OASIS (EP/S032347/1), UK FIRES (EP/S019111/1) and ConCur (EP/V050869/1).

References

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona T. Diab, and Marjan Ghazvininejad. 2022. A Review on Language Models as Knowledge Bases. *ArXiv*, abs/2204.06031.

Michael Ashburner, Catherine A. Ball, Judith A. Blake, David Botstein, Heather L. Butler, J. Michael Cherry, Allan Peter Davis, Kara Dolinski, Selina S. Dwight, Janan T. Eppig, Midori A. Harris, David P. Hill, Laurie Issel-Tarver, Andrew Kasarskis, Suzanna E. Lewis, John C. Matese, Joel E. Richardson, Martin Ringwald, Gerald M. Rubin, and Gavin Sherlock. 2000. Gene Ontology: tool for the unification of biology. *Nature Genetics*, 25:25–29.

Sean Bechhofer, Frank Van Harmelen, Jim Hendler, Ian Horrocks, Deborah L McGuinness, Peter F Patel-Schneider, Lynn Andrea Stein, et al. 2004. OWL web ontology language reference. *W3C recommendation*.

Jiaoyan Chen, Yuan He, Ernesto Jiménez-Ruiz, Hang Dong, and Ian Horrocks. 2022. Contextual Semantic Embeddings for Ontology Subsumption Prediction. *ArXiv*, abs/2202.09791.

Philipp Cimiano and Uwe Reyle. 2003. Ontology-based semantic construction underspecification and disambiguation. In *Proceedings of the Lorraine/Saarland Workshop on Prospects and Recent Advances in the Syntax-Semantics Interface, October 20-21, 2003, Nancy, France*, pages 33–38.

Jeff Da, Ronan Le Bras, Ximing Lu, Yejin Choi, and Antoine Bosselut. 2021. Analyzing Commonsense Emergence in Few-shot Knowledge Models. In *Conference on Automated Knowledge Base Construction*.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Ning Ding, Shengding Hu, Weilin Zhao, Yulin Chen, Zhiyuan Liu, Haitao Zheng, and Maosong Sun. 2022. OpenPrompt: An open-source framework for prompt-learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 105–113, Dublin, Ireland. Association for Computational Linguistics.

Damion M. Dooley, Emma J. Griffiths, Gurinder Pal Singh Gosal, Pier Luigi Buttigieg, R. Hoehndorf, Matthew Lange, Lynn M. Schriml, Fiona S. L. Brinkman, and William W. L. Hsiao. 2018. FoodOn: a harmonized food ontology to increase global food traceability, quality control and data integration. *NPJ Science of Food*, 2.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. 2014. HermiT: an OWL 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269.

Bernardo Cuenca Grau, Ian Horrocks, Boris Motik, Bijan Parsia, Peter Patel-Schneider, and Ulrike Sattler. 2008. OWL 2: The next step for OWL. *Web Semantics*, 6(4):309–322.

Yuan He, Jiaoyan Chen, Denvar Antonyrajah, and Ian Horrocks. 2022. BERTMap: A BERT-Based Ontology Alignment System. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(5):5684–5691.

Pride Kavumba, Ryo Takahashi, and Yusuke Oda. 2022. Are prompt-based models clueless? In *Proceedings of the 60th Annual Meeting of the Association for*

Computational Linguistics (Volume 1: Long Papers), pages 2333–2352, Dublin, Ireland. Association for Computational Linguistics.

Markus Krötzsch, František Simančík, and Ian Horrocks. 2012. A Description Logic Primer. *ArXiv*, abs/1201.4089.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori Hashimoto. 2022. Diffusion-LM Improves Controllable Text Generation. *ArXiv*, abs/2205.14217.

Hao Liu, Yehoshua Perl, and James Geller. 2020. Concept placement using BERT trained by transforming and summarizing biomedical ontology structure. *Journal of Biomedical Informatics*, 112:103607.

Linlin Liu, Xin Li, Ruidan He, Lidong Bing, Shafiq Joty, and Luo Si. 2021a. Knowledge Based Multilingual Language Model. *arXiv preprint arXiv:2111.10962*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2022. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys (CSUR)*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zeyu Liu, Yizhong Wang, Jungo Kasai, Hannaneh Hajishirzi, and Noah A. Smith. 2021b. Probing across time: What does RoBERTa know and when? In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 820–842, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*.

Sebastian Padó and Ido Dagan. 2022. 679Textual Entailment. In *The Oxford Handbook of Computational Linguistics*. Oxford University Press.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Adam Poliak. 2020. A survey on recognizing textual entailment as an NLP evaluation. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 92–109, Online. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2022. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.*, 21(1).

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Stefan Schlobach. 2005. Debugging and Semantic Clarification by Pinpointing. In *The Semantic Web: Research and Applications, Second European Semantic Web Conference (ESWC)*, volume 3532 of *Lecture Notes in Computer Science*, pages 226–240. Springer.

Lynn M. Schriml, Cesar Arze, Suvarna Nadendla, Yu-Wei Wayne Chang, Mark Mazaitis, Victor Felix, Gang Feng, and W. Kibbe. 2012. Disease Ontology: a backbone for disease semantic integration. *Nucleic Acids Research*, 40:D940 – D946.

Alessandro Solimando, Ernesto Jiménez-Ruiz, and Giovanna Guerrini. 2017. Minimizing conservativity violations in ontology alignments: algorithms and evaluation. *Knowl. Inf. Syst.*, 51(3):775–819.

Robyn Speer and Catherine Havasi. 2012. Representing General Relational Knowledge in ConceptNet 5. In *LREC*.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4723–4734, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume*

1 (Long Papers), pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Hongbin Ye, Ningyu Zhang, Shumin Deng, Xiang Chen, Hui Chen, Feiyu Xiong, Xi Chen, and Huajun Chen. 2022. Ontology-enhanced Prompt-tuning for Few-shot Learning. *Proceedings of the ACM Web Conference 2022*.

A OWL Ontology Concept Expression

The Description Logic $\mathcal{SROIQ}$ underlies the semantics of OWL 2 ontologies. Given the top concept $\top$, the bottom concept $\bot$, the named concept A, an individual a, a role (or property) r and a non-negative integer n, $\mathcal{SROIQ}$ concept expressions are constructed as:

$$C, D ::= \top | \bot | A | (C \sqcap D) | (C \sqcup D) | \neg C | \exists r.C | \forall r.C | \geq n\ r.C | \leq n\ r.C | \exists r.Self | \{a\}$$

Recall the definition of *interpretation* $I = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, where $\Delta^{\mathcal{I}}$ is a non-empty set (the domain) and $\cdot^{\mathcal{I}}$ maps each concept C to $C^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, a each property r to $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ and each individual a to an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$. We present the semantics of the concept constructors in Table 5.

Constructor	Semantics
A	$A^{\mathcal{I}}$
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
$\top$	$\Delta^{\mathcal{I}}$
$\bot$	$\emptyset$
$\exists r.C$	$\{x \mid$ some $r^{\mathcal{I}}$-successor of x is in $C^{\mathcal{I}}\}$
$\forall r.C$	$\{x \mid$ all $r^{\mathcal{I}}$-successors of x are in $C^{\mathcal{I}}\}$
$\geq n\ r.C$	$\{x \mid$ at least n $r^{\mathcal{I}}$-successors of x are in $C^{\mathcal{I}}\}$
$\leq n\ r.C$	$\{x \mid$ at most n $r^{\mathcal{I}}$-successors of x are in $C^{\mathcal{I}}\}$
$\exists r.Self$	$\{x \mid \langle x, x\rangle \in r^{\mathcal{I}}\}$
$\{a\}$	$\{a^{\mathcal{I}}\}$

Table 5: Semantics of the OWL Ontology concept constructors.

B Ontology Preprocessing

In case that some of the ontologies we use in this work contain meaningless (e.g., obsolete) concepts regarding subsumption sampling and/or contain concept names (or aliases) that are apparently unnatural, we apply a **general** preprocessing procedure to all the ontologies, and then conduct **individual** preprocessing for each ontology.

General Preprocessing

- Remove obsolete concepts (which are indicated by the built-in annotation property `owl:deprecated`) and apparently redundant concepts such as `foodOn:stupidType`.
- Use `rdfs:label` as the main annotation property to extract concept names except when its literal value is not available. The extracted concept names are lower-cased and any underscores "_" in them are removed.

Individual Preprocessing

- `Schema.org`: concept names (defined in this ontology are in the Java-identifier style; thus, they are parsed into natural expressions, e.g., *"APIReference"* to *"API Reference"*.
- `DOID`: remove the concept `doid:Disease` because it is a general concept just below the root concept `owl:Thing` which will lead to too many simple subsumptions in the form of $C \sqsubseteq Disease$.
- `FoodOn`: reconstruct label strings containing non-natural-language texts of three regular expression patterns (note that `(.*)` captures what to be preserved):

  (a) `[0-9]+ - (.*) \(.+\)`

  (b) `\('(.*)\(gs1', 'gpc\)'\)`

  (c) `\('(.*)\(efsa', 'foodex2\)'\)`

  followed by removal of leading and trailing whitespaces. Note that concepts in this ontology sometimes have an empty literal given by `rdf:label`; in these cases, the annotation properties `obo:hasSynonym` and `obo:hasExactSynonym` are used instead.
- `GO`: no individual processing.

C Object Property Verbalisation

Different from verbalising an atomic concept where we simply use its name (or alias), we enforce some simple rules to verbalise an object property for a basic grammar fix. If the property name starts with a *passive verb*, *adjective*, or *noun*, we append *"is"* to the head. For example, *"characteristic of"* is changed to *"is characteristic of"*; *"realised in"* is changed to *"is realised in"*. Note that the word's part-of-speech tag is automatically determined using the Python library `Spacy`[19].

[19] https://spacy.io/

Complex Concept C	**Verbalisation** $\mathcal{V}(C)$
$BioRegulation \sqcap \exists negRegulate.ProlineBiosynProc$	*"biological regulation that negatively regulates some proline biosynthetic process"*
$ApoptoticProc \sqcap \exists partOf.Luteolysis$	*"apoptotic process that is part of some lutelysis"*
$ConcnOf \sqcap \exists charOf.(fucose \sqcap \exists partOf.MaterialEnt)$	*"concentration of something that is characteristic of some fucose that is part of some material entity"*
$\exists derivesFrom.(TimothyPlant \sqcup TrifoliumPratense) \sqcap PlantFoodProd \sqcap Silage$	*"silage and plant food product that derives from some timothy plant or trifolium pratense"*
$Apple \sqcap \neg\exists hasPart.ApplePeel$	*"apple (whole or parts) and not something that has part some apple peel"*

Table 6: Examples of verbalised complex concepts from GO's and FoodOn's Complex SI datasets. Note that in the real datasets, the named concepts and object properties are represented by their IRIs (unique identifiers) instead of the abbreviated names shown in the table.

D Complex Concept Verbalisation Examples

For clearer understanding of how our verbalisation approach works, we present some typical examples of verbalised concepts from the constructed Complex SI datasets in Table 6.

E Implementation Choices for Assumed Disjointness

As mentioned in Section 3.1, validating the disjointness axiom for each concept pair (C, D) we have sampled as a potential negative subsumption would be time-consuming because we need to iteratively add the disjointness axiom into the ontology $\mathcal{O}$, conduct reasoning, and remove the axiom afterwards. Therefore, in practice we can use the following conditions to replace the satisfiability check:

(i) **No subsumption relationship**: $\mathcal{O} \not\models C \sqsubseteq D$ and $\mathcal{O} \not\models D \sqsubseteq C$;

(ii) **No common instance**: there is no *named* instance a in $\mathcal{O}$ such that $\mathcal{O} \models C(a)$ and $\mathcal{O} \models D(a)$.

If C and D satisfy these two conditions, they are **likely to be satisfiable** after adding the disjointness axiom $C \sqcap D \sqsubseteq \bot$ into $\mathcal{O}$. Since these two conditions involve **no extra reasoning** for a new axiom, they are much more efficient than iteratively conducting satisfiability check for candidates.

It is important to notice that we still need the **no common descendant** check to prevent foreseeable unsatisfiability.

(iii) **No common descendant**: there is no named atomic concept A in $\mathcal{O}$ such that $\mathcal{O} \models A \sqsubseteq C$ and $\mathcal{O} \models A \sqsubseteq D$.

This is because if there is a named atomic concept A that is an inferred sub-class (i.e., descendant) of C and D, then it is possible that C and D are satisfiable in $\mathcal{O} \cup \{C \sqcap D \sqsubseteq \bot\}$, but A is certainly unsatisfiable (equivalent to $\bot$).

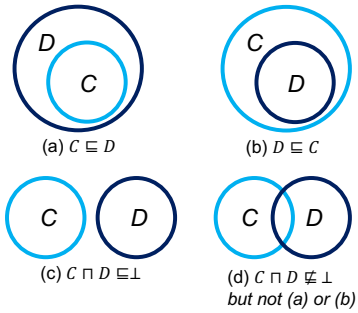

Figure 4: Set-based semantics for relationships between two ontology concepts.

F Set-based Interpretations of Subsumption Samples

In this section, we provide more explanation for how we define positive and negative samples in the Subsumption Inference (SI) task.

Recall the definitions in Section 2.1, an ontology $\mathcal{O}$ entails a subsumption axiom $C \sqsubseteq D$ if it holds for **every interpretation** $\mathcal{I}$ of $\mathcal{O}$. In terms of set-based semantics, this refers to case *(a)* in Figure 4. In the *(b)*, *(c)*, or *(d)* cases, there exists at least one interpretation $\mathcal{I}$, such that we can find an individual x that $x^{\mathcal{I}} \in C^{\mathcal{I}}$ and $x^{\mathcal{I}} \notin D^{\mathcal{I}}$; hence $\mathcal{O}$ **does not entail** the subsumption axiom $C \sqsubseteq D$. Non-subsumption is entailed only when *(a)* **does not hold for every interpretation** of $\mathcal{O}$.

Disjointness corresponds to *(c)* in Figure 4 where the set of C and the set of D have no overlap for every interpretation. Non-subsumptions an ontology **typically** entails come from the disjointness axioms (but disjointness $\forall x.C(x) \rightarrow \neg D(x)$ is

		Atomic SI				Complex SI	
Setting	**biMNLI**	**Schema.org**	**DOID**	**FoodOn**	**GO**	**FoodOn**	**GO**
roberta-base							
K=0	62.5 (6.5)	56.4 (3.6)	53.3 (4.0)	54.6 (4.4)	49.0 (2.4)	55.9 (3.6)	48.7 (3.1)
K=4	67.6 (5.2)	62.9 (5.2)	61.8 (6.7)	62.1 (4.2)	65.2 (5.0)	62.4 (3.2)	52.2 (7.1)
K=8	70.7 (4.5)	71.2 (4.5)	72.9 (5.7)	69.0 (5.2)	70.4 (5.1)	66.0 (4.4)	63.0 (5.0)
K=16	74.3 (3.3)	79.7 (4.2)	83.4 (2.5)	79.8 (3.0)	78.3 (3.0)	70.2 (5.5)	73.8 (4.0)
K=32	78.8 (1.1)	84.3 (2.0)	89.0 (1.4)	85.0 (1.1)	84.6 (2.5)	77.0 (1.5)	76.4 (2.5)
K=64	80.9 (1.5)	88.3 (1.5)	91.2 (0.7)	88.2 (0.7)	87.3 (0.8)	80.0 (2.0)	81.7 (1.4)
K=128	85.1 (1.0)	91.1 (0.7)	92.4 (0.7)	90.0 (0.7)	89.0 (0.8)	85.5 (1.3)	86.9 (1.5)
roberta-large							
K=0	68.7 (6.2)	61.7 (7.2)	59.8 (5.4)	60.1 (8.8)	54.6 (1.9)	56.1 (1.9)	50.4 (0.6)
K=4	78.1 (6.6)	69.4 (5.4)	74.0 (5.5)	71.6 (4.4)	67.6 (3.4)	64.1 (5.1)	56.9 (5.7)
K=8	83.0 (5.2)	78.5 (3.0)	84.4 (3.8)	77.0 (6.0)	75.3 (3.2)	71.3 (3.1)	64.2 (7.6)
K=16	87.5 (2.4)	84.4 (2.4)	87.6 (2.3)	83.4 (3.5)	82.8 (1.9)	76.2 (2.5)	76.4 (3.3)
K=32	89.9 (1.2)	87.3 (1.9)	92.3 (0.7)	88.9 (1.6)	87.7 (1.6)	80.8 (3.8)	81.6 (2.2)
K=64	90.8 (1.4)	90.4 (0.8)	92.6 (0.7)	90.9 (1.2)	90.1 (0.7)	84.1 (1.4)	86.2 (2.2)
K=128	93.0 (0.8)	92.9 (0.8)	93.4 (0.5)	92.2 (0.5)	91.0 (0.7)	88.4 (1.1)	90.2 (1.0)
full	97.5	95.4	97.8	98.7	98.1	95.8	98.8

Table 7: Full results of roberta-base and roberta-large on the biMNLI, Atomic SI, and Complex SI tasks with each cell stating *"mean accuracy (standard deviation)"* except for the majority vote and fully supervised settings where standard deviation is not available.

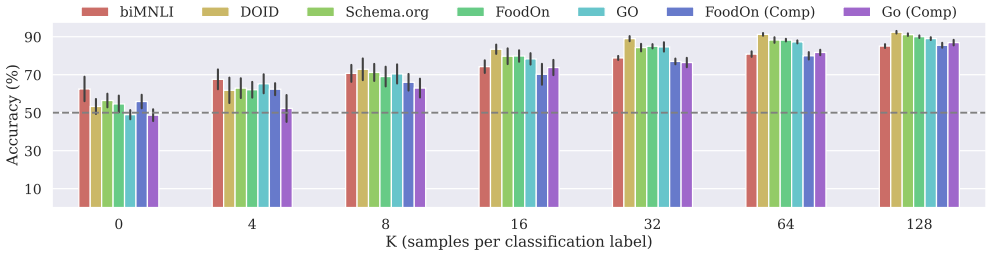

Figure 5: Visualisation of K-shot results (for roberta-base) on the biMNLI, Atomic SI, and Complex SI tasks where the dotted horizontal line indicates majority vote. The order of the bars is the same as in the legend.

stricter than non-subsumption $\exists x.C(x) \wedge \neg D(x)$). Nevertheless, ontologies are typically underspecified in terms of disjointness, and thus getting enough negative samples is unfeasible. To find a middle ground, it is reasonable to adopt heuristics. The assumed disjointness we follow in Section 3.1 in the main body of the paper serves this purpose. In the ideal setting where we check the **satisfiability** of C and D after adding the disjointness axiom and **no common descendant** of C and D, cases *(a)* and *(b)* in Figure 4 will be prevented and the chance of *(d)* reduced. Even in the practical alternative proposed in this Appendix E, the **no subsumption relationship** condition also ensures that *(a)* and *(b)* are not entailed and the **no common descendant** and **no common instance** conditions reduce the chance of *(d)*. Thus, the assumed disjointness is a reasonable approach to approximate non-subsumptions.

G Complementary Results and Figures

In the main body of the paper, we report partial results (accuracy scores and standard deviations) of roberta-large and roberta-base for $K \in \{0, 4, 32, 128\}$. In Table 7, we present full results of both LMs for $K \in \{0, 4, 8, 16, 32, 64, 128\}$.

Besides, we provide the visualisation of K-shot results for roberta-base in Figure 5. The observations are consistent with those for roberta-large in Figure 3.

ACL 2023 Responsible NLP Checklist

A For every submission:

☐ A1. Did you describe the limitations of your work?
Left blank.

☐ A2. Did you discuss any potential risks of your work?
Left blank.

☐ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☐ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☐ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
Left blank.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Left blank.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Left blank.

C ☐ Did you run computational experiments?

Left blank.

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Left blank.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☐ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Left blank.

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Left blank.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Left blank.

D ☐ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Left blank.