DeepOnto: A Python Package for Ontology Engineering with Deep Learning and Language Models

Yuan He

Department of Computer Science

University of Oxford

yuan.he@cs.ox.ac.uk

DEPARTMENT OF
COMPUTER
SCIENCE

UNIVERSITY OF
OXFORD

Ontology

- An ontology is an **explicit specification** of a **conceptualisation**.

  model of domain knowledge

- What are explicitly specified?
  - The set of entities: **concepts**, **properties**, and **instances**.
  - The (logical) **relationships** among them.

- Web Ontology Language (OWL):
  - (W3C) standardisation.
  - High expressiveness.
  - Reasoning capabilities.

Ontology Engineering

- Concerns different stages of ontology development.
  - E.g., ontology design, construction, curation, etc.

- Challenges:
  - **Scalability**: knowledge is ever-expanding.
  - **Quality assurance**: consistency, completeness.
  - **Integration**: merging data from heterogeneous sources.

Automation is Required

Solution:

Build ontology engineering systems with **Deep Learning**, and particularly, **Language Models**.
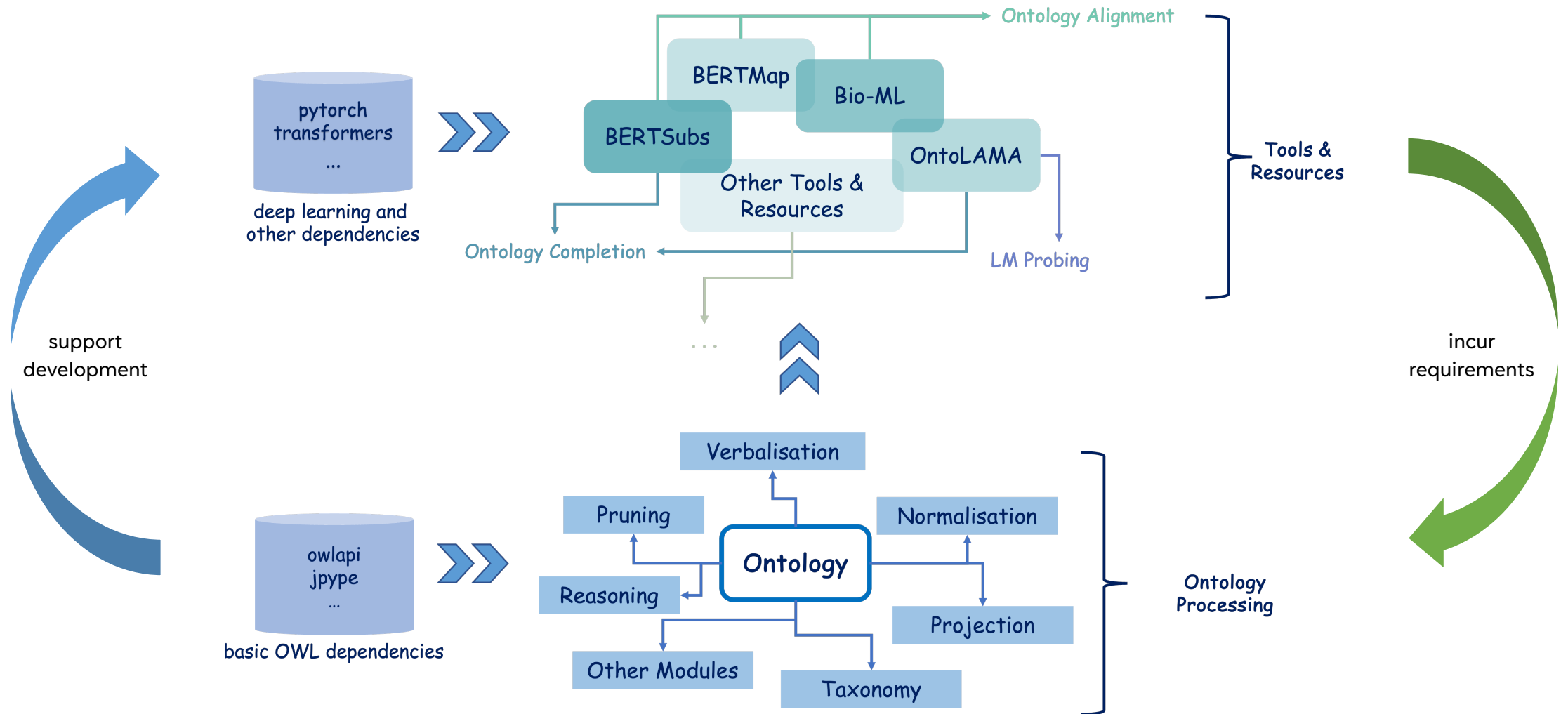
DeepOnto

- A Python package for ontology engineering with deep learning.

We will briefly introduce:

- Overview.

- Dependencies.

- Ontology processing modules.

- Tools and resources built upon the modules.

- Long-term stability.

- Impact and use.

Overview

Ontology Summit 2024
DeepOnto: A Python Package for Ontology Engineering with DL and LMs

Dependencies

- **OWL API** (Java-based) for basic ontology processing features.

- **PyTorch** for deep learning framework.

- **Huggingface Transformers** for language models.

Dependencies

Q: Why Java?

- Reliable and stable **ontology APIs** were mainly built with Java:
  - E.g., The OWL API, Jena, Jcel, etc.

- Previous attempt on solely Python build failed:
  - Owlready2 lacked essential features.
  - Requires considerable low-level engineering efforts if build from scratch.

Dependencies

Q: Why Python? 🐍

- Mainstream open-source **deep learning frameworks** are Python-based.
- E.g., load a pre-trained distilled BERT:

```
>>> from transformers import AutoModelForSequenceClassification

>>> model = AutoModelForSequenceClassification.from_pretrained("distilbert-base-uncased")
```

Figure: Code snippet taken from the Huggingface Transformers documentation.

To bridge Java and Python, we used **JPype**.

The Base Ontology Class

`deeponto.onto.Ontology`

- Encapsulate basic features from the OWL API, such as:
  - Access and manipulate entities and axioms.
  - Query for (asserted) annotations, parents and children.

Ontology Reasoning

`deeponto.onto.OntologyReasoner`

- Conduct reasoning on ontologies such as:
  - Various entailment and consistency checks.
  - More complex reasoning subject to task requirements.

- Different types of reasoners supported.
  - **HermiT**: sound and complete.
  - **ELK**: tailored to OWL EL.
  - **Struct**: simple structural reasoning.

Ontology Verbalisation

deeponto.onto.OntologyVerbaliser

- Verbalising complex concepts into natural language texts.

- A rule-based recursive algorithm is implemented.

- Useful for models (e.g., language model-based) that take textual inputs.

$FoodProduct \sqcap \exists. derivesFrom. (InvertebrateAnimal \sqcup VertebrateAnimal)$

$\sqcap$

$FoodProduct$ $\exists. derivesFrom. (InvertebrateAnimal \sqcup VertebrateAnimal)$

$\exists$

food product $derivesFrom$ $InvertebrateAnimal \sqcup VertebrateAnimal$

$\sqcup$

derives from $InvertebrateAnimal$ $VertebrateAnimal$

invertebrate animal   vertebrate animal

invertebrate animal or vertebrate animal

derives from invertebrate animal or vertebrate animal

food product that derives from invertebrate animal or vertebrate animal
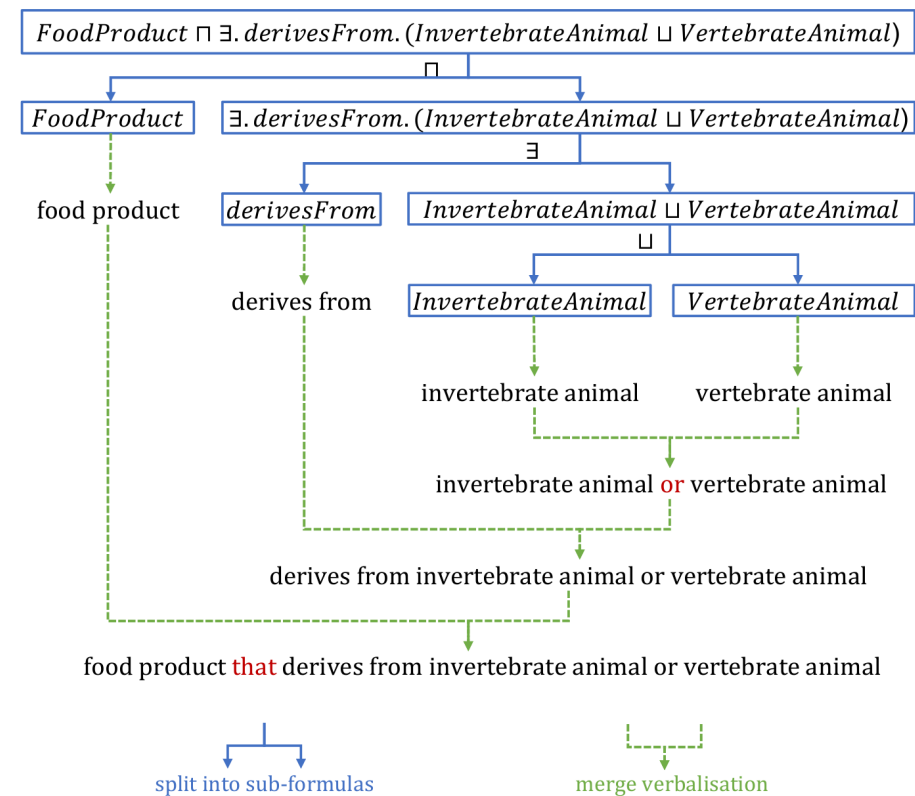
split into sub-formulas   merge verbalisation

Figure: Example taken from the OntoLAMA paper by He et al. (ACL'23: Findings).

Ontology Pruning

deeponto.onto.OntologyPruner

- Extract sub-ontologies.

- Current implementation based on reconstructing subsumptions.
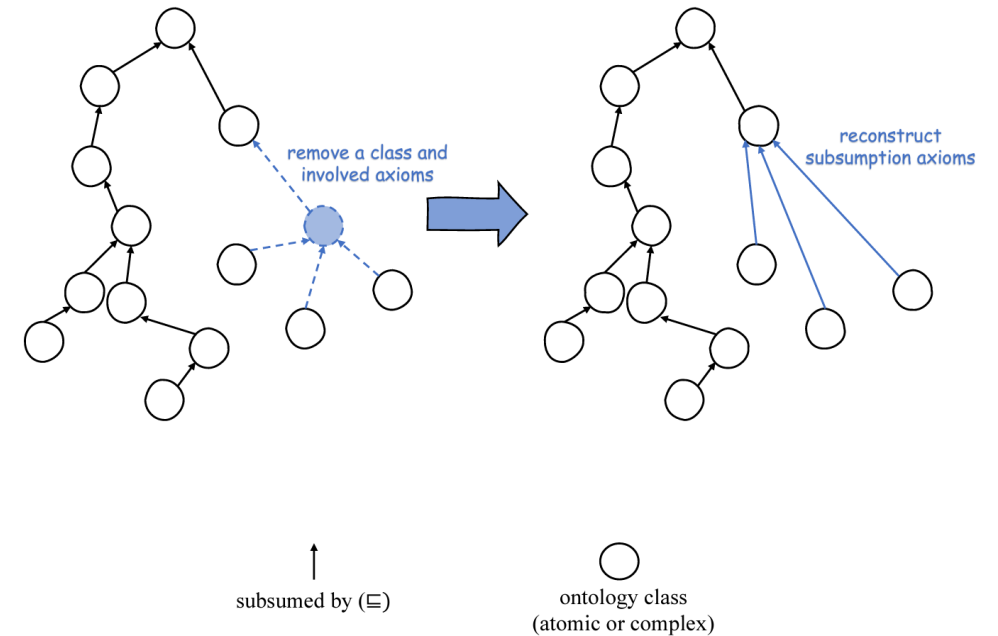
- Preserving logical modules as a future work.

Figure: Illustration of the pruning algorithm from the Bio-ML paper by He et al. (ISWC'22).

Ontology-to-Graph

deeponto.onto.OntologyProjector

- Project an ontology into triples.

deeponto.onto.OntologyTaxonomy

- Extract subsumption hierarchy (tree graph) from an ontology.

To support:

- Graph-based deep learning models, e.g., OWL2Vec*, TransE.

More Processing Modules

`deeponto.onto.OntologyNormaliser`

- Normalise axioms into basic forms to support, e.g., logical embedding.

`deeponto.onto.OntologyModulariser`   (*future work*)

- Extract logical modules for a given set of axioms.

Tools and Resources

- The ontology processing modules are **building blocks** for tools and resources.

- DeepOnto currently incorporates tools and resources mainly for ontology **alignment** and **completion**.

Task Definitions

- **Ontology Alignment** is the task of identifying the set of mappings (that indicate semantic relationships) between entities from different ontologies.

- **Ontology Completion** is the task of identifying missing semantic relationships between entities within the same ontology.

Ontology Alignment

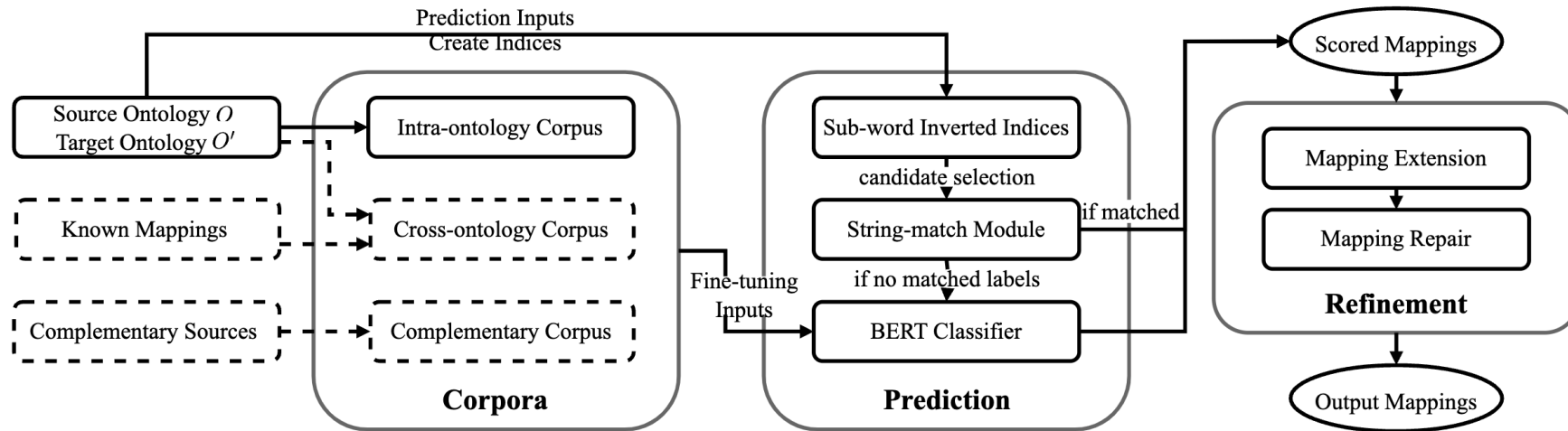- **BERTMap** for concept equivalence ontology alignment.

Figure: Illustration of the BERTMap system by He et al. (AAAI'22).

Ontology Alignment

- **Bio-ML**: a resource for concept equivalence and subsumption alignment (used in the OAEI track since 2022).

Ontology Summit 2024
DeepOnto: A Python Package for Ontology Engineering with DL and LMs

Ontology Completion

- **BERTSubs** for predicting missing subsumption relationships between ontology concepts.
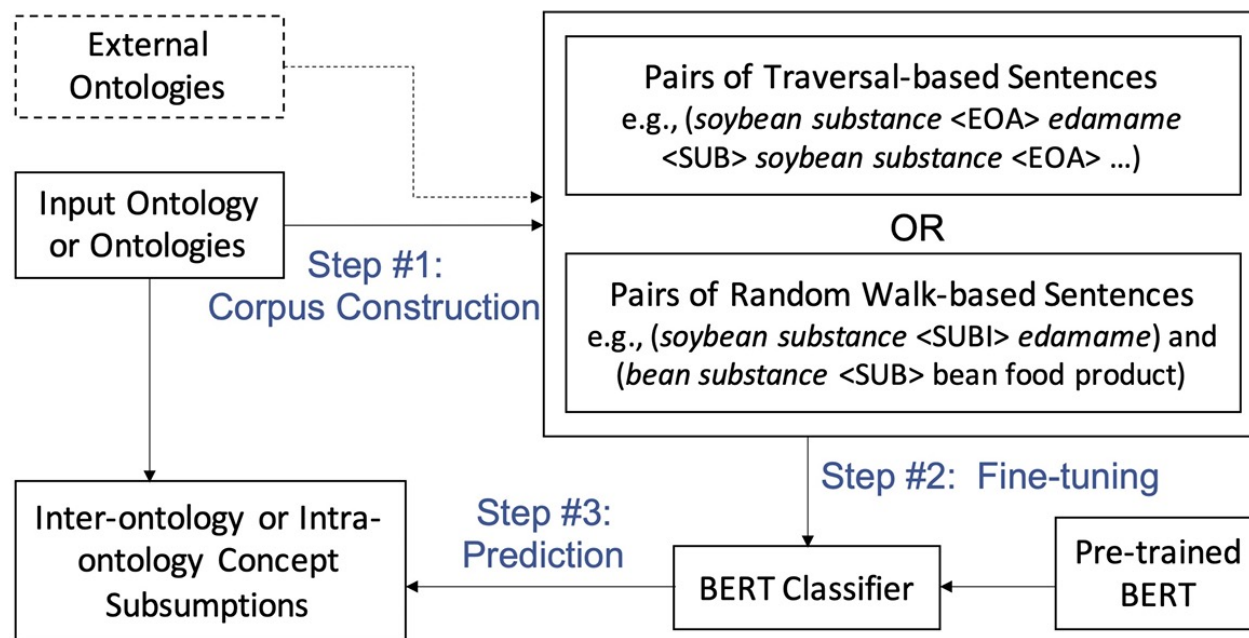

Figure: Illustration of the BERTSubs system by Chen et al. (WWWJ'23).

Ontology Completion

- **OntoLAMA** is originally for probing conceptual knowledge (from ontologies) encoded by language models. The probing method based on *prompt learning* can be adopted for ontology completion.

$$C := Beef$$
$$D := Meat \sqcap \exists derivesFrom.Cattle$$

Extract Samples

entailed subsumption → **Positive**

assumed disjointness → Negative

Ontology $\mathcal{O}$

Verbaliser $\mathcal{V}$

$$\mathcal{V}(C) := \text{"beef"}$$
$$\mathcal{V}(D) := \text{"meat that derives from cattle"}$$

Template($\mathcal{V}(C), \mathcal{V}(D)$, <MASK>)
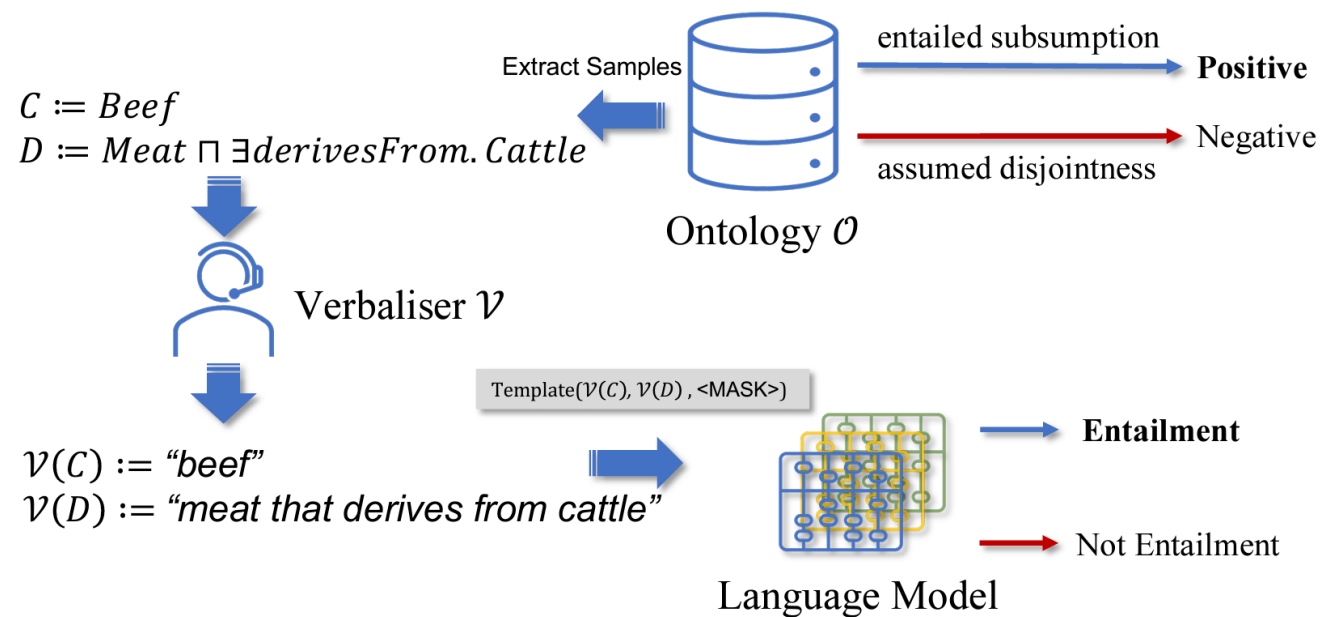
Entailment

Not Entailment

Language Model

Figure: Illustration of the OntoLAMA framework by He et al. (ACL'23: Findings).

More Tools and Resources

- Ontology construction and curation encompass **a diverse range of tasks** besides alignment and completion.

- Even settings within alignment and completion **vary**.

- DeepOnto will incorporate **more tools and resources** for tasks like ontology embedding, concept insertion, and more.

Long-term Stability

- The OWL API 4 is **statically deployed** – so the ontology processing features are relatively stable.

- Deep learning frameworks are rapidly developing but they are actively maintained for **backward compatibility**.

Impact and Use

- DeepOnto has been adopted in by **Samsung Research UK** (SRUK) for developing Digital Health Coaching.

- DeepOnto offers resources and evaluation framework for the **Bio-ML track of the OAEI**.

- Received help requests for research, internship, micro-services, etc.

Impact and Use

- Still a young package gaining attention from the community.

- Feedback and help/feature requests are extremely welcomed.

About

A package for ontology engineering with deep learning and language models.

🔗 **krr-oxford.github.io/DeepOnto/**

`deep-learning` `ontologies`
`language-model` `ontology-engineering`
`owlapi`

📖 Readme
⚖️ Apache-2.0 license
💗 Code of conduct
⚖️ Security policy
〰️ Activity
☆ **112** stars
👁 **4** watching
⑂ **8** forks

Figure: DeepOnto's GitHub statistics on 17th October 2023.

THANK YOU!