Self-learning ontological concept representation for searching and matching tasks

Duy-Hoa Ngo, Bevan Koopman

The Australian e-Health Research Centre, CSIRO

Abstract

Ontology searching and matching can be achieved by learning vector-based representation of ontology concepts. Choosing appropriate features is a key step to learn good concept representations. Lexical annotations and structural features are complementary to the concept representation, but how to combine them to work together effectively is still an open question. To handle this problem, we propose a self-generated data method from input ontologies and corresponding deep neural network models to encode ontological concepts as embedding vectors. Our experiments on biomedical ontologies (SNOMED CT vs. HPO) showed that using semantic embedding features can increase searching effectiveness by over 20% from baseline methods. Our approach provides a generic method for both ontology searching and matching.

Keywords

Machine learning, Label embedding, Transformers, Semantic similarity, Ontology matching

1. Introduction

Concept matching and searching are fundamental for tasks such as text data annotation, integration and analysis. Concept matching involves matching a concept in one ontology to its equivalent in another. In health domain, matching facilitates interoperability, thus allowing collaboration between various healthcare centres when their health data resources can be shared and integrated for intensive analysis. Whereas, concept searching involves finding the relevant concept in an ontology given some free-text input extracted from many sources like patient's medical record, publications, reports or news about health care. Searching allows unstructured text to be mapped to a standard coding system to reduce ambiguity and enable machine understanding through semantic expressions.

Ontology matching (OM) tools aim to automate the laborious human task of concept matching between ontologies. While OM tools continue to make improvements, there is still a gap between automated OM methods and human terminology experts [1]. Typically, the produced mappings are suggested as the best results that the OM tool can find, but nothing guarantees that they are all correct or complete. Therefore, terminology experts are still needed to verify the matching results by using different features of concepts such as lexical annotations (e.g., concepts' labels, descriptions) and structural information (relationship between concepts) to remove incorrect and add missing mappings. Lexical and structural information are complementary to the concept

OM-2022: The 17th International Workshop on Ontology Matching, Hangzhou, China, October 23-27, 2022 hoa.ngo@csiro.au (D. Ngo)

^{© 0 2022} Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

representation, but the way to combine them effectively is still a hard challenge [2].

To handle this problem, we propose a method to learn semantic vector representation of ontology concepts from their lexical and structural features, which then can be used with any distance metric for searching and matching concepts among ontologies. Particularly, we propose a self-generated data method from input ontologies and corresponding deep neural network models to encode ontological concepts into semantic embedding vectors.

2. Methodology

Below we list some specific observations that inform our automated approach:

- (a) The matching step usually comes after searching step, in which only small set of concepts will be selected as potential candidates for further analysis. A popular method is based on using a hash function on a concepts' string label to filter out concepts sharing one or few tokens. A drawback of this method is fail when concept labels share no common terms; e.g., Hypoplasia of lower limb vs. Rudimentary leg, which occurs frequently in biomedical ontologies.
- (b) A specific meaning of a word depends on the context where it stands with other words and the domain knowledge it belongs to.
- (c) Structural information presents the semantic of a concept through logic expressions with other concepts in the ontology. However, it is usually used to check consistency of the mapping candidates instead of discovering new mappings because even the same concept may have vary logic expressions in different ontologies.
- (d) Hierarchical information presents a degree of granularity, which can be used for estimating the distance of concepts in the same ontology.
- (e) A common pattern in defining a new concept in biomedical domain is "Lexically suggest, logically define" [3], which means the logical expression of the concept can be inferred from the meaning of its names.

The observations shows the importance of lexical annotations in representation of concepts, but without understanding the meaning of words in their context, potential matching candidates will be ignored. A better concept representation is an embedding vector, which shows promising results in deep learning approaches for sentence embedding [4]. Therefore, we encode every concept label in ontologies into a semantic embedding vector. Then, a concept representation is an average vector embedding of all its label embedding vectors.

On the other hand, observations shows the importance of structural information as well as the vocabulary used in an ontology is domain specific, so concept vector representations should be learned directly from the ontologies in the same domain. Additionally, a distance metric can be used to measure the similarity of vector representation, so vector representation should be learned in a way that the distance of closely related concepts in ontology hierarchy should be closer than that of the unrelated concepts.

2.1. Deep neural network architecture

As the objectives of learning a good concept representation is its use in similarity and ranking distance tasks, Siamese [4] and Triplet [5] neural network models (see Fig. 1) were adopted for training. In summary, these models use a BERT [8] - a transformer-based deep learning for natural language processing (NLP), with a pooling strategy to encode a string input into an embedding vector.

An input instance to the Siamese model consists of two string and a similarity score. The training objective is to fine tune parameters of BERT thus the cosine similarity of embedding vectors of the the two input strings approximates to the given similarity score. Whereas for the Triplet model, an input instance includes three strings: an *Anchor* string, its closely related *Positive* string, and an unrelated *Negative* string. Its training objective is to fine tune BERT so the distance between the embedding vectors of *Anchor* and *Positive* is less than the distance between embedding vectors of *Anchor* and *Negative* vectors. Since there is no public and available machine learning data set for learning concept representation in these formats, we proposed algorithms to self-generate training data for Siamese and Triplet models from the ontologies.



Figure 1: Siamese and Triplet neural network architecture

2.2. Self-generated data from ontologies

To provide training data set for Siamese model, a similarity score is required for every pair of string labels. Lin measure [7] with ancestors' subgraph-based intrinsic information content (AsIIC) [6] has been chosen as they showed the highest correlation with human judgments in several widely-used benchmarks.

Generating training data for the Siamese model works as follows. Computes intrinsic information content, AsIIC value, for every concepts. Randomly choose pairs of concepts and calculate their Lin similarity score. Assigns similarity scores for all pairwise annotated labels of concepts chosen in the previous calculation. Additionally, if a concept has multiple preferred labels or synonymous labels, then the algorithms assigns similarity score 1.0 to each pair of those labels.

There are two methods for generating training data for the Triplet model: a "*hard*" method and an "*adaptive*" method. The "*hard*" method applies two distance ranking rules namely *Synonym* and *Parent-Child* on the hierarchical structure of the given ontology. The Synonym rule states that the two synonym labels are the most similar pair, so they always take the place of *Anchor* and *Positive* inputs; any other concept label will be a *Negative* input. The Parent-Child rule states that the distance from a child to its direct parent is always less than the distance of this child to its siblings, uncles or grandparents. Then, for concept label, the algorithm assigns it as an *Anchor* and randomly assigns the concept's parent as a *Positive*; the *Negative* is randomly chosen from the concept's siblings, uncles or grand-parents.

On the other hand, for the "*adaptive*" method, the *Synonym* rule is used to generate *Anchor* and *Positive* inputs as described above; a pre-trained model is used to generate *Negative* inputs. Particularly, for each concept label, a pre-trained model is used to search the most similar labels from all concepts in the given ontology. By removing all synonymous labels of the searched label from the searching results, we obtain a list of *Negative* inputs. This forces the model to learn to be far more sensitive to what is actually a relevant matching concept rather than just a related one.

3. Experiment and evaluation

In the scope of the short paper, we only design experiments and perform evaluation for ontological concept searching task. For evaluation, we used *clinical concept alignment* between Human Phenotype Ontology (HPO) and SNOMED CT (SCT)¹. It was manually created by terminology experts in our terminology matching project and far more complete than the mappings provided in BioPortal automatically generated by LOOM algorithm². Our dataset contains 5,978 HPO concepts having been matched to SCT concepts. The total annotated labels of those selected HPO concepts are 14,149 labels.

Two types of searching experiments were conducted as follows: 1) **Label searching** illustrates a scenario where a terminologist wants to retrieve the best matched concept for a given query string. 2) **Concept searching** illustrates a scenario where a terminologist wants to match concepts from different ontologies for further data interoperability.

Firstly, we encoded all concepts in SCT into embedding vectors, then index them into Non-Metric Space for quick approximate nearest neighbour searching algorithm [9]. For searching, HPO concepts and their annotated labels are also encoded into an embedding vectors that can be searched from indexing space to return top K nearest neighbours. Recall that a vector representation of a concept is defined by getting the average of its labels' embedding vectors.

Table 1 shows the evaluation results for different approaches. We use Hits@K (K = 1, 5, 10) - a common metrics in information retrieval to evaluate the searching performance of those methods. For a given query, a Hits@K value is equal to 1 if the relevant concept is found in the top K results; otherwise it is 0. The **Baseline** is a simple BM25 search model using the concept

¹HPO ver. 08.11.2019 vs. SCT ver. 29.02.2020

²https://bioportal.bioontology.org/ontologies/HP/?p=mappings

HPO2SCT Query Size	Label searching #labels=14149			Concept searching #concepts=5978		
Hits@K	K=1	K=5	K=10	K=1	K=5	K=10
Baseline (BM25)	0.334	0.486	0.553	0.529	0.718	0.778
Bio-BERT	0.296	0.438	0.488	0.549	0.740	0.794
Siamese BERT	0.616	0.830	0.888	0.728	0.902	0.937
Triplet BERT	0.608	0.797	0.844	0.768	0.923	0.944
Continue Training I	0.645	0.842	0.878	0.782	0.930	0.958
Continue Training II	0.703	0.881	0.916	0.792	0.922	0.947

 Table 1

 Hits@K evaluation on concept and label searching

label as the query. The **Bio-BERT** method uses a pre-trained BioBERT³ to encode a concept label into a embedding vector. **Siamese BERT** and **Triplet BERT** do the same, but the BERT parameters have been fine-tuned with training data self-generated from HPO and SCT. The **Continue Training I** is a continuous training starting from pre-trained Bio-BERT to Siamese BERT and finally Triplet BERT. Here, the data set for the **Triplet BERT** model is generated by using the "*hard*" method. The **Continue Training II** is a continuous training starting from pre-trained Bio-BERT to Triplet BERT using "*hard*" method, then again continue training with Triplet BERT using the "*adaptive*" method.

Our proposed method improves Hits@K by 30% at label level and over 20% at concept level in comparison with the **Baseline** and pre-trained **Bio-BERT** methods in all Hits@K metrics. The experimental results also show that without training on appropriate data, the performance of **Bio-BERT** does not improved from the **Baseline** method, despite the fact that **Bio-BERT** is the state-of-the-art language model achieving the best results in many biomedical NLP tasks. Another interesting point here is that Siamese BERT outperforms Triplet BERT at label level, but worse than that at the concept level, however, after continuous training, the final models achieved the best results at all levels. These experiments demonstrate the importance of self-generated data from the input ontologies in learning ontological concept representation, and it suggests that continuous training can enhance the concepts' vector representation and improve the concept searching performance.

4. Conclusion

We present three methods of self-generating training data from ontologies. A Siamese BERT and Triplet BERT networks learn concept representation that are used for ontology searching and matching tasks. An empirical evaluation on two biomedical ontologies (SNOMED CT & HPO) these methods were effective as matching relevant ontology concepts, outperforming existing BM25 and BERT baselines. These model represent to further step toward fully automated ontology matching that does not require laborious manual effort by humans.

³https://github.com/dmis-lab/biobert

References

- [1] Euzenat, Jerome and Shvaiko, Pavel. Ontology Matching. Springer (2013).
- [2] DH Ngo and Zohra Bellahsene. Overview of YAM++ (not) Yet Another Matcher for ontology alignment task. J. Web Semantics (2016)
- [3] Rector, Alan and Iannone, Luigi. Lexically Suggest, Logically Define: Quality Assurance of the Use of Qualifiers and Expected Results of Post-Coordination in SNOMED CT. Journal of Biomedical Informatics (2012)
- [4] Reimers, Nils and Gurevych, Iryna. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. EMNLP (2019).
- [5] Ngo DH, Kemp M, Truran D, Koopman B, Metke-Jimenez A. Semantic Search for Large Scale Clinical Ontologies. AMIA (2021).
- [6] Ben Aouicha M, Hadj Taieb MA. Computing semantic similarity between biomedical concepts using new information content approach. J Biomed Inform. (2016)
- [7] Lin Dekang. An Information-Theoretic Definition of Similarity. ICML (1998)
- [8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL (2019).
- [9] Malkov, Yu A. and Yashunin, D. A. Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs. IEEE Trans. Pattern Anal. Mach. Intell (2020).