Towards Domain-Specific Ontology Learning

Daniel Yakubov

Abstract

Ontologies are often used as a graphical database in industrial settings. Traditionally, ontologies are curated and maintained by a domain expert. However, since this task requires extensive manual intervention it can be costly, difficult, and time-consuming. These issues motivate the field of Ontology Learning (OL). This paper approaches the problem as an unsupervised task and attempts to learn a domain-specific ontology from a corpus of online posts from a skateboarding forum. This is done using a variety of techniques from NLP including learning embeddings and clustering them hierarchically. Ultimately, the result of this work requires further research to continue, however, the early results are promising.

Introduction

In philosophy, Ontology has a long tradition, dating back to the teachings of Aristotle. It is a field of study that aims to classify all entities in all spheres of existence (Smith 2012). An ontology can be thought of as a graphical database consisting of nodes and links between the nodes. In an ontology, links can represent non-hierarchical relationships, so it is distinct from a taxonomy, in which the links only store the hierarchical 'IS-A' relationship. In applied ontology, one needs to decide to narrow the scope of an ontology, as storing the ontology described would run against computational limitations, such as memory constraints and long inference time due to the many possible paths that an algorithm needs to traverse in the graph. The creation of an applied ontology requires the involvement of a domain specialist and costly labor. Further, applied ontologies need to be maintained in order to keep the information in the graph up-to-date. These logistical problems are the primary motivation for the field of Ontology Learning (OL). Techniques for OL vary greatly with some using more linguistic and logical techniques and others being more grounded in statistical systems (Asim et al 2018). Many existing OL techniques involve the intermediate use of an existing Ontology such as WordNet (Wątróbski 2020). While this direction of work is interesting, this study's approach focuses on learning a novel domain-specific ontology directly from the corpus.

In order to further motivate an automated ontology learning system, we will present two salient cases: 1) The recent large language model boom has led to the development of models that produce near-human text. While the generated text can score well on NLP evaluation benchmarks and human evaluations (OpenAI 2023), the models are capable of 'hallucinating' false information or bringing in information that is not present at inference time (Ji et al. 2023). Using a learned ontology could help at inference time with such models. 2) The detection of deception and misinformation in text is difficult for both human observers and machine learning

systems. Using an ontology as a fact-checker could help remove such texts from the training data before it is used for model training or analysis of a text.

The work in this paper is far from terminal as there are a lot of improvements that could be made and further research directions that could be explored. This work frames OL as an unsupervised task and attempts to induce an ontology from a corpus scraped from an internet skateboarding forum. The reason for the choice of domain is a secondary research direction for this work. Skateboarding tricks are highly compositional, for example, a 'varial' is a trick that consists of a 'kickflip', which is a clockwise roll of a skateboard in the air and a 'shuv', which is a 180-degree rotation of a skateboard. Thus, it can be said that a 'kickflip' + 'shuv' = 'varial'. This work explores if this compositionality is preserved in learned embeddings.

Data

Scraping

The corpus used for this work was a collection of 58,000 posts scraped from the website http://www.skateboard-city.com. Specifically, the posts were collected from the industry news section of the message board. The choice of the industry news section is motivated by the assumption that it would contain fewer irreverent conversations than other sections of the website. When scraping the data, the posts were stored in jsonl format with their thread titles kept near the content of the post. Importantly, the posts were not stored in relation to one another. The reason for this is we wanted to maintain that each post was primarily discussing the topic, not the message it was in response to.

Preprocessing

Preprocessing the data consisted of several steps. The first step was coreference resolution, this is the task of replacing pronouns with the noun that they index. For example, the sentence 'John really likes his cat' gets resolved to be 'John really likes John's cat'. Coreference resolution was done using fastcoref (Otmazgin et al. 2022). One complication in this step comes from a name in the title being coindexed with a pronoun in the text for example:

Thread title: ***Dylan Rieder***

Post: I've only been skating for about two years, but I would always watch video parts because I just loved skateboarding and when ***his*** *Gravis part came out, I was like holy shit. When* ***he*** *backside boardslid that rail into traffic, I flipped out. I checked instagram the other day and saw Gary Rogers' photo of* ***him*** *and I honestly got really emotional,* ***Dylan*** *was amazing and any time someone has to battle cancer, thats sad enough, but when they die from it, thats brutal. RIP* ***Dylan****.*

Because of this effect, the thread title was prepended to the text with the dummy text "The thread title is [TITLE]" where [TITLE] is replaced by the thread title. This allowed for coreference

resolution to consider a name in the title, if there is a name. The next step was case-folding, converting the words in the text into their lemma forms, and removing any stop words. Then, the text is filtered for any non-alphabetic characters, and a Phraser is run on the text in order to detect phrases in the text, so a commonly occurring string, ‘tampa pro’ for example, is stored as a unigram ‘tampa_pro’. This condensation of phrases is needed for learning meaningful embeddings of the text. The ultimate result of this preprocessing is a data structure which is a list of lists that contain word-level unigrams.

Method

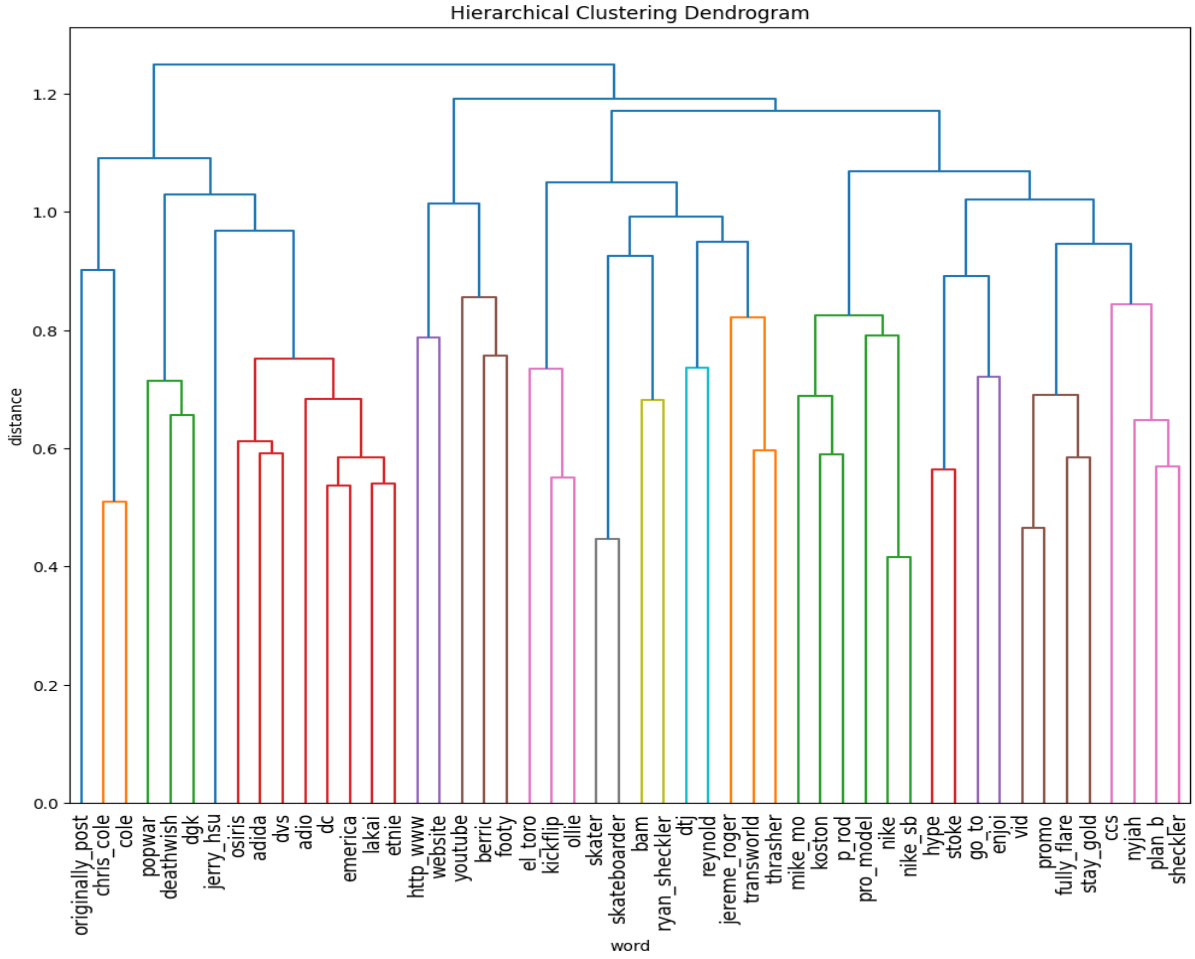
The method used adapts a simplified version of a pipeline described by Wątróbski (2020) for OL. First, using Word2Vec (Mikilov et al 2013) word-level embeddings are computed for each word in the data. Then, in order to narrow down the number of words and identify the entities and concepts unique to the domain of skateboarding, several steps are taken. First, any word that is not a noun or a proper noun is discarded because entities/concepts are assumed to only be encoded by nouns. Then, to isolate the words unique to the domain of skateboarding, two corpora, Brown and NPS chat, are loaded in. Brown is used to represent formal English discourse terminology, and NPS chat is used to represent informal English discourse, the skateboarding discourse intersects with both of these domains. Using these corpora, we filter words from the skateboarding corpus if they appear in either corpus. This narrows down the number of words for analysis to 1429. The remaining words are now domain specific due to the process, they are not found in other representative corpora.

In order to uncover the hierarchical IS-A relationship between the terms, the technique of hierarchical clustering is used on the embedding vector representations of each term. This results in a taxonomy of terms. The discovery of non-hierarchical relationships would be the next step in the development of an OL system, but it was unsuccessful in my attempts.

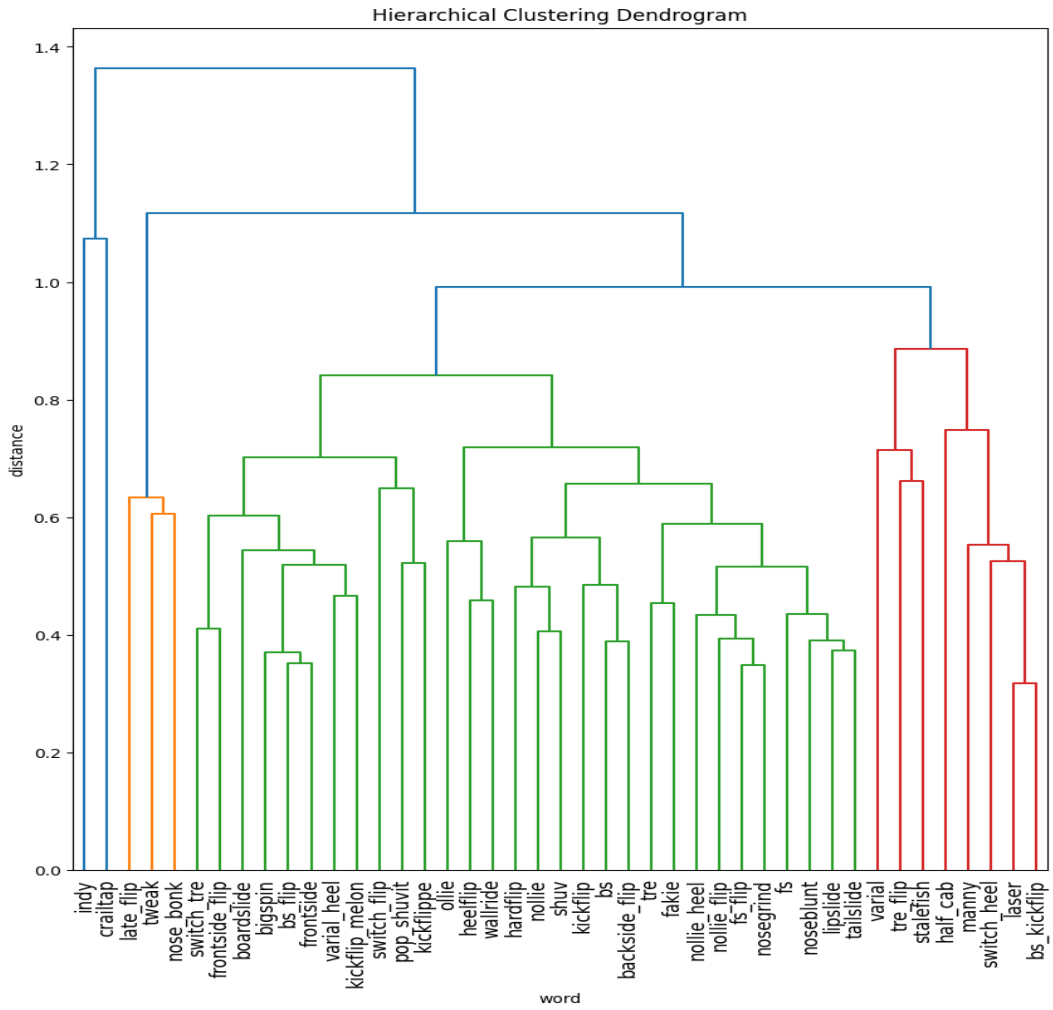
Results & Analysis

The dendrogram below shows the results of the hierarchical clustering on 100 words (limited for visualization). The granularity of some clusters can be a bit arbitrary, for example, the cluster denoted by the red lines on the left side of the dendrogram contains skateboarding brands, however, there is no particular reason why ‘dc’ and ‘emerica’ are clustered together before all the other brands. It also captures the relationship between names that denote the same person, for example, it can be seen that ‘chris_cole’ and ‘cole’ cluster closely together. However, a lot of relationships represented here are not IS-A relationships and require a different analysis technique. (hype, stoke) is a cluster that contains skater slang, but then it clusters with (go_to, enjoi) which is a cluster containing a brand name. It is unclear what relationship this holds. Partly, this could be seen as a fault of using embeddings - clustering embeddings results in words

that are similar in some way, primarily in the contexts they appear in, this does not encode any explicit relationship between the words being represented.



To explore if skateboarding trick compositionality can be captured by embeddings, the data had to be curated. Terms denoting skateboarding tricks were picked out of the corpus of 1429 words. There were 43 tricks in the corpus. In general, the produced taxonomy is very inaccurate in terms of order of the relationships portrayed. For example, ‘kickflip’ attaches to the cluster (bs, backside_flip), despite it being a component of the latter trick (the correct order would be (bs, kickflip) = backside flip. Interestingly, the tricks that are done on ramps are clustered far from the other tricks, on the left (indy, crailtap), so there is some latent knowledge of different contexts where these tricks would occur. Given that the data is scraped from an industry news section it is possible that it did not contain enough text about tricks. Further work in this direction could include scraping from another source, such as a forum where new skateboarders may post about tricks and the relationships are stated more explicitly by helpful commenters. Also, experimenting with different types of embeddings and embedding windows could help here.



Conclusion and Future Work

This work described the need for ontology learning systems and provided an attempt to develop such a system and test it on the domain of skateboarding discourse. The results presented ultimately were not satisfactory and the problem of OL proves to be difficult. Future work in this space could continue and propose methods for discovering non-hierarchical relationships between embeddings of words. Further, future work could attempt to incorporate even more data to see if it alleviates the issues encountered here.

References

- Smith, B. (2012). Ontology. In *The furniture of the world* (pp. 47-68). Brill.
- Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, *55*(12), 1-38.
- Muhammad Nabeel Asim, Muhammad Wasim, Muhammad Usman Ghani Khan, Waqar Mahmood, Hafiza Mahnoor Abbasi, A survey of ontology learning techniques and

applications, *Database*, Volume 2018, 2018, bay101, https://doi.org/10.1093/database/bay101

- Shon Otmazgin, Arie Cattan, and Yoav Goldberg. 2022. F-coref: Fast, Accurate and Easy to Use Coreference Resolution. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 48–56, Taipei, Taiwan. Association for Computational Linguistics.
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Wątróbski, J. (2020). Ontology learning methods from text - an extensive knowledge-based approach. *International Conference on Knowledge-Based Intelligent Information & Engineering Systems*.
- OpenAI* (2023), GPT-4 Technical Report, Openai website.