Improved Concept Embeddings for Learning Prerequisite Chains

Enrico Löffel

Department of Computer Science, Yale University

LILY Lab

Introduction

Word embedding is one the most common techniques for feature learning in Natural Language Processing (NLP), allowing us to retrieve functional and semantic similarities between words and concepts from their vector representations. While most state-of-the-art methods for word embedding perform really well on tasks that require similarity analysis, none of them are optimized for dealing with hierarchical structures or deriving hierarchical relations between entities. Nickel and Kiela (2017) claim that these state-of–the-art methods lack performance in hierarchical sets because they use Euclidean vector spaces, which do not account for hierarchy. They propose instead Poincaré embeddings as a method optimized for data with latent hierarchies, as it embeds word and concepts into a hyperbolic space, which can be thought of as a continuous version of trees. Dhingra et al. (2018) describe a method to obtain non-parametric unsupervised Poincaré word embeddings from natural-language text corpora, using co-occurrences of pairs of concepts within the corpus. In this project, we combine these two methods in order to derive prerequisite relations between concepts in the domain of Natural Language Processing, drawing from a corpus of 7,472 text files collected from online courses, tutorials and academic publications.

Materials

In this project, we use two Datasets:

- LectureBank - manually-collected dataset of 1352 lecture slide presentations from 60 courses covering 5 different domains: Natural Language Processing, Machine Learning, Artificial Intelligence, Deep Learning and Information Retrieval.
- TutorialBank - manually-collected dataset of over 6000 resources, ranging from HTML pages to lecture slides and textbooks, mainly in the domain of Natural Language Processing.

Along with these two datasets, we use a list of 208 annotated topics and the prerequisite relations between them, both part of TutorialBank. Some minor modifications were made to this list of topics in order to account for common abbreviations and different ways to write the concepts.

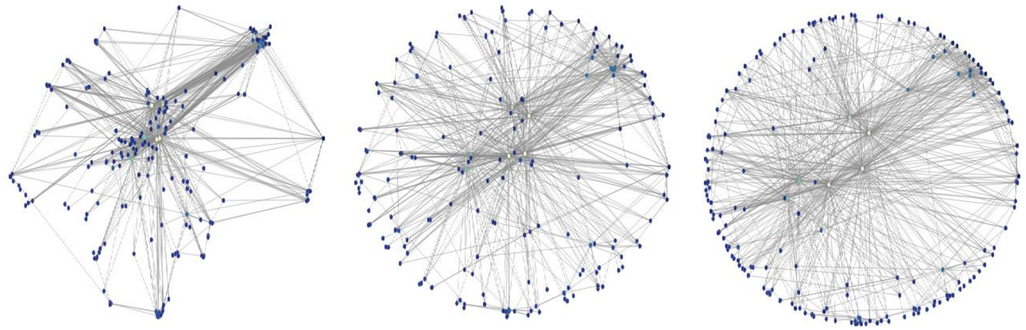

Figure 1. Poincaré embeddings after 15, 50 and 100 epochs, respectively, for the graph of annotated prerequisites. Concepts lower in the hierarchy are closer to the border, while concepts higher in the hierarchy are closer to the center.

Approach

The first step in our approach is to construct co-occurrence weighted graphs $G = \{(w, v, f^c)\}$ from all concept pairs (w, v) appearing within a fixed window of each other in the corpus, where f represents the number of occurrences and $c < 1$ is a downsampling constant. We use windows of size 10, 20 and 50 and also the number of co-occurrences in the entire document. Then, we embed these graphs into a Poincare n-ball using the algorithm created by Nickel and Kiela (2017). We have used n-balls of dimension 2, 3, 5, 10, 50 and 100.

Evaluation Method

We use two evaluation methods to test our results. The first one is trying to reconstruct the annotated prerequisites from the trained embeddings and reporting the Mean Average Precision (MAP) and the Mean Rank. The second, which would be equivalent to calculating the recall, is to rank all topics by increasing norm (the closest the norm of a topic is to zero, the higher it is in the hierarchy) and to verify how many annotated prerequisite relations from the annotated set are satisfied. As a form of comparison, we get a MAP of 0.82 and a recall of 0.97 when training the Poincaré Model with the annotated graph.

$$d(\boldsymbol{u}, \boldsymbol{v}) = \cosh^{-1}\left(1 + 2\frac{\|\boldsymbol{u} - \boldsymbol{v}\|^2}{(1 - \|\boldsymbol{u}\|^2)(1 - \|\boldsymbol{v}\|^2)}\right)$$

Equation 1. Distance between two vectors in a n-dimensional Poincaré unit ball

			Dimensionality					
			2	3	5	10	50	100
Window Size	10	MAP	0.096	0.113	0.112	0.118	**0.119**	0.116
		Recall	**0.561**	0.495	0.509	0.516	0.521	0.520
	20	MAP	0.095	0.124	0.127	0.130	**0.132**	0.131
		Recall	**0.569**	0.548	0.543	0.550	0.549	0.550
	50	MAP	0.091	0.121	0.131	**0.132**	0.131	0.131
		Recall	**0.551**	0.544	0.541	0.541	0.541	0.538
	Doc	MAP	0.083	0.139	0.186	0.190	0.191	**0.191**
		Recall	**0.629**	0.620	0.620	0.617	0.614	0.615

Table 1. Results using $c = 0.5$ and combining LectureBank and TutorialBank

Results

There is strong empirical evidence that the Poincaré word embedding model alone is able to derive hierarchical relations between concepts completely unsupervised, which is something that no other model that uses an Euclidean vector space can. There are a few trends in the results, however, that we should point out. First, increasing the dimensionality increases the Mean Average Precision, as the prerequisite closure is a highly intricate graph and trying to represent all edges in a two-dimensional Poincaré disk leads to many false positives (prerequisite pairs that do not exist even though one concept has lower norm and is close to the other concept). A second trend is that a lower dimensionality results in higher recall, which is reasonable considering it has a much lower precision. The third trend is that calculating the number of co-occurrences in the entire document leads to a much higher Mean Average Precision and Recall than calculating co-occurrences within a fixed window. We can explain this trend considering the nature of the corpora, as two topics would hardly be mentioned in the same lecture, tutorial or paper if they were not correlated. Also, using a fixed window of 10 words leads to 3600 co-occurrence pairs, most of which with weight 1. Using the entire documents leads to 26000 co-occurrence pairs, with several distinct weights. For comparison, the annotated closure has 900 pairs.

Conclusion

Poincaré Embeddings provide a really interesting approach to prerequisite chain learning and to unsupervised learning in hierarchical datasets. The fact that the model achieves much better results when considering co-occurrences in the document as a whole instead of within a fixed window shows that there a strong synergy between using Poincaré Embeddings and clustering documents based on their topics, and that we can further improve the task of generating prerequisite chains by combining the two approaches

References

Maximilian Nickel and Douwe Kiela. 2017. *Poincaré embeddings for learning hierarchical representations.*

Bhuwan Dhingra, Christopher J. Shallue, Mohammad Norouzi, Andrew M. Dai, and George E. Dahl. 2018. *Embedding text in hyperbolic spaces.*

Acknowledgement

Thanks to Dragomir Radev and Alex Fabbri for their guidance throughout this project.