Olive Oil is Made *of* Olives, Baby Oil is Made *for* Babies: Interpreting Noun Compounds using Paraphrases in a Neural Model Supplementary Material

Anonymous ACL submission

Abstract

Automatic interpretation of the relation between the constituents of a noun compound, e.g. *olive oil* (source) and *baby oil* (purpose) is an important task for many NLP applications. Recent approaches are typically based on either noun-compound representations or paraphrases. While the former has initially shown promising results, recent work suggests that the success stems from memorizing single prototypical words for each relation. We explore a neural paraphrasing approach that demonstrates superior performance when such memorization is not possible.

1 Technical Details

Fixed Hyper-parameters. For the paths extraction we consider sentences with up to 32 words and dependency paths with up to 8 edges, including satellites, and keep only 1,000 paths for each noun-compound. We use TensorFlow (Abadi et al., 2016) to train the network, fixing the values of the hyper-parameters after performing preliminary experiments on the validation set. We set the mini-batch size to 10, use Adam optimizer (Kingma and Ba, 2014) with the default learning rate, and apply word dropout with probability 0.1. We initialize the distributional embeddings with the 300-dimensional pre-trained GloVe embeddings (Pennington et al., 2014) and the lemma embeddings (for the path-based component) with the 50-dimensional ones. Noun-compound embeddings are learned using a concatenation of the English Wikipedia and Gigaword corpora.[1]

Tuned Hyper-parameters. For the following hyper-parameters, we choose the values that maximize the F_1 score on the validation sets: the corpus from which paths are extracted (Wikipedia, or a concatenation of Wikipedia and Gigaword), and the number of epochs to train, between 1 and 30. We apply early stopping and stop the training when the F_1 score on the validation set drops 8 points below the best performing score.

2 Dataset Split

Table 1 displays the sizes of each dataset split.

Dataset	Split	Train	Validation	Test
TRATZ-FINE	Lex$_{full}$	4,730	1,614	869
	Lex$_{head}$	9,185	5,819	4,154
	Lex$_{mod}$	9,783	5,400	3,975
	Rand	14,369	958	3,831
TRATZ-COARSE	Lex$_{full}$	4,746	1,619	779
	Lex$_{head}$	9,214	5,613	3,964
	Lex$_{mod}$	9,732	5,402	3,657
	Rand	14,093	940	3,758

Table 1: Number of instances in each dataset split.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467* .

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Doha, Qatar, pages 1532–1543. http://www.aclweb.org/anthology/D14-1162.

[1] Similarly to the original GloVe implementation, we only keep the most frequent 400,000 vocabulary terms, which means that roughly 20% of the noun-compounds do not have vectors and are initialized randomly in the model.