

CogALex-V Shared Task: LOPE

Kanan Luce

University of California,
Berkeley

kanan.luce@berkeley
.edu

Jiaxing Yu

Nanjing University

jiaxingyu@gmail.com

Shu-Kai Hsieh

National Taiwan University

shukaihsieh@ntu.edu.
tw

Abstract

This paper attempts to answer two questions posed by the CogALex shared task: How to determine if two words are semantically related and, if they are related, which semantic relation holds between them. We present a simple, effective approach to the first problem, using word vectors to calculate similarity, and a naive approach to the second problem, by assigning word pairs semantic relations based on their parts of speech. The results of the second task are significantly improved in our post-hoc experiment, where we attempt to apply linguistic regularities in word representations (Mikolov 2013b) to these particular semantic relations.

1 Introduction

Automatic discovery of semantically-related words is one of the most important NLP tasks, and has great impact on the theoretical psycholinguistic modeling of the mental lexicon. In this shared task, we employ the *word embeddings model* (Mikolov 2013a) to reflect paradigmatic relationships between words. Previous work has shown that word representations extracted from simple recurrent neural networks could hierarchically categorize words based on their collocational distribution (Elman 1990). Word representations also hold other regularities. More recently, Mikolov et al. (2013b) showed that word vectors could be added or subtracted to isolate certain semantic and syntactic features. The well-known example is to take the representations for *king*, subtract *man*, and add *woman*. This produces a vector very near by *queen*. This method was tweaked by Levy and Goldberg (2014) by representing the same idea as three pairwise similarities, and is the basis for the post hoc revisions to our system.

The particular semantic relations we are concerned with in this paper are synonymy, antonymy, hypernymy, and meronymy. The shared task consists of two subtasks. The first is to, given a pair of words, identify if they are semantically related. The second task is to determine, if the pair is related, what relation there is between them. We will present first our official system for each subtask, followed by our post-hoc changes.

2 Subtask 1

Subtask 1 was to see if two words were semantically related. For our system, we returned true if word 2 was in the top n similar word vectors for word 1, or vice versa. We used the pretrained Google News vectors (Mikolov 2013a), which are 300 dimensions, and contain a vocabulary of 3 million words, and used the Gensim Word2Vec library (Rehurek and Sojka 2010) to manipulate the data.

We found that the best results were achieved when we considered the top 600 similar words. This number had the best coverage without suffering from too many false positives. We also found it helpful to limit the vocabulary from 3 million to only the top 50,000 most frequently occurring tokens, which eliminated unlikely candidate word-forms.

This work is licensed under a Creative Commons Attribution 4.0 International Licence. Licence details: <http://creativecommons.org/licenses/by/4.0/>

Initially, we found the 600 most similar words by building a dictionary of each word in the training data and its corresponding related words. Because we used the same dictionary made from the training data when running the finalized version on test data, our official submission only looked up words that occurred in the training data. In our post-hoc experiment for subtask 1, we use all the words from the test data as well. The results of both systems are below.

LOPE	P = 0.596	R = 0.886	F1 = 0.713
LOPE-PH	P = 0.623	R = 0.884	F1 = 0.731

Table 1: Compares F1 scores of the original (LOPE) and post-hoc (LOPE-PH) systems on Subtask 1

While the performance of the post-hoc system was slightly better, it did not make substantial gains on the original system. The recall for the two systems was almost the same despite the original system not containing the test data, because as long as one of the two words in a pair was in the dictionary, the system could still find related words.

3 Subtask 2

Subtask 2 asks us to take the related pairs from subtask 1 and determine what their relation is: either synonyms, antonyms, hypernyms, or meronyms. Our original system used a crude method to categorize the pairs based on their parts of speech. In the training data, nouns, verbs, and adjectives occurred at different frequencies for each relation, and we used that information to sort them into their mostly likely categories. Noun-noun pairs were sorted as hypernyms, adjective-adjective pairs were sorted as antonyms, and verb-verb pairs were split between antonyms and synonyms based on where the word pair occurred in the list of 600 similar words. If the word occurred in the first 100 most similar it was sorted as a synonym, otherwise it was sorted as an antonym. The part of speech information of each pair was determined by finding the most frequent shared part of speech between the two words as they appeared in the Brown corpus. While this approach was better than a random baseline, it is not helpful in that it does not provide us with any useful information and the results were lackluster.

We significantly improved our results in the post hoc system by completely changing approach and using a method inspired by Mikolov et al.'s method (2013b) of finding linguistic regularities in word representations. We were curious if this method could be applied to this particular problem of finding differences between synonyms, antonyms, hypernyms, and meronyms.

We initially implemented a method inspired by Levy and Goldberg (2014). We used three word representations, one related pair from the training data and one word from the input pair, in order to predict the other word from the input pair as one of the most similar vectors. The idea being that the cosine similarity of the target fourth word will be different in the case where the semantic relation of the input pair matches that of the training pair.

This assumption was incorrect, and we had to revise our approach. We instead started to find the cosine similarity between the two sets of words (the input set, and a related set from the training data). The cosine similarity was often higher if the two sets shared a relation. While this was inconsistent when comparing only two sets, we found that we could compare (find the cosine similarity of) an input set to each antonym, synonym, hypernym and meronym set in the training data, average the results for each semantic relation, and then assign the input pair to the class that had the highest average.

LOPE	P	R	F1	LOPE-PH	P	R	F1
SYN	0.304	0.191	0.235	SYN	0.089	0.438	0.148
ANT	0.417	0.217	0.285	ANT	0.447	0.405	0.425
HYPER	0.328	0.406	0.363	HYPER	0.199	0.514	0.287
PART_OF	0.000	0.000	0.000	PART_OF	0.411	0.365	0.387
Subtask 2	0.289	0.231	0.247	Subtask 2	0.373	0.414	0.374

Table 2: Compares F1 scores of the original (LOPE) and the post-hoc (LOPE-PH) system on Subtask 2

So if the average of the input pair compared to all of the antonym pairs was higher than it was for the same comparison to all of the synonym, hypernym, and meronym sets, then the input pair was assigned the relation antonym. Using this method, we were able to significantly increase our F1 score for the task and our coverage of which relations we were able to get right (we completely ignored meronyms in the first system). However, we still struggled with some relations more than others. Synonyms in particular had a very poor precision, and the accuracy of the system on synonyms was much lower than for the other three relations.

4 Conclusion

There were two mistakes in the initial version of our system. First, there was no reason not to use the test data rather than the training data when looking at the top 600 similar representations for each word in a pair. The difference, however, in the results was relatively small. A much more significant error was our original system for solving subtask 2, which was both relatively ineffective and didn't show anything interesting..

Despite these errors, we were able to propose a system that while very simplistic and easy to implement, was able to achieve good results compared to the rest of the field. Table 3 shows the results of the various systems in the shared task on both subtask 1 and subtask 2.

Subtask 1		Subtask 2	
Team	F1	Team	F1
GHHH	0.790	LexNET	0.445
Mach5	0.778	GHHH	0.423
LexNET	0.765	LOPE-PH	0.374
ROOT18	0.731	Mach5	0.295
LOPE-PH	0.731	ROOT18	0.262
LOPE	0.713	CGSRC	0.252
HsH-Supervised	0.585	LOPE	0.247
CGSRC	0.431		

Table 3: Results of the different systems in the CogALex shared task, with the addition of our post-hoc system

In particular, our method of comparing the input pair to each related pair in the training data, averaging the results in each relational category, and assigning the input pair the relation with the highest average, appears to be effective at categorizing pairs according to their semantic relationship. Although our exact method was different than that of Mikolov et al. and Levy and Goldberg, it shows that the linguistic regularities they found in word embeddings are useable to find this kind of paradigmatic information about the semantic relationships synonymy, antonymy, hypernymy, and meronymy.

While not at the top of the table for either subtask, we believe we were able to put up respectable results for a simple system. It is possible that with a more complex expansion of the system, we could improve the results even more, particularly by finding ways to increase the accuracy of synonym detection in subtask 2.

4.1 Further Study

As a further study, we would like to attempt the task in Chinese. We argue that relation extraction is a task that could be language/writing system dependent. For example, in Chinese, it would be possible to exploit morpho-semantic relations and the character radical ontology (paradigmatic information embedded in the characters) to re-conduct subtask 2. We are currently underway creating original Chinese language data from Chinese Word Net to mirror the English data, so as to avoid translating polysemous words in English that aren't polysemous in Chinese, such as *cell* (a *cell* could be a small room or a part of an organism).

Reference

- Jeffrey Elman. 1990 . Finding structure in time. *Cognitive Science*, 14, 179-211
- Omar Levy and Yoav Goldberg. 2014. Linguistic regularities in sparse and explicit word representations. In *Proceedings of the Eighteenth Conference on Computational Language Learning*, pages 171-180, Baltimore, Maryland USA, June 26-27 2014. Association for Computational Linguistics
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.
- Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia, June. Association for Computational Linguistics.
- Radim Rehurek and Petr Sojka. 2010. Software framework for topic modeling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45-50, Valletta, Malta, May 22. ELRA
- Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. EVALution 1.0: An evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics*, Beijing