CogALex-V Shared Task: HsH-Supervised – Supervised similarity learning using entry wise product of context vectors

Rosa Tsegaye Aga and **Christian Wartena**
Hochschule Hannover
Hanover, Germany
{rosa-tsegaye.aga, christian.wartena}@hs-hannover.de

Abstract

The CogALex-V Shared Task provides two datasets that consists of pairs of words along with a classification of their semantic relation. The dataset for the first task distinguishes only between related and unrelated, while the second data set distinguishes several types of semantic relations. A number of recent papers propose to construct a feature vector that represents a pair of words by applying a pairwise simple operation to all elements of the feature vector. Subsequently, the pairs can be classified by training any classification algorithm on these vectors. In the present paper we apply this method to the provided datasets. We see that the results are not better than from the given simple baseline. We conclude that the results of the investigated method are strongly depended on the type of data to which it is applied.

1 Introduction

In distributional semantics words are represented by a large number context features. In most cases, words context features are based on co-occurrences number or probabilities with other words. It turns out that words with similar vectors of co-occurrence based features are semantically related. A simple approach to decide whether two words are semantically related or not, can be based directly on the similarity of their associated vectors. This approach has been used in a large number of studies.

In order to improve on the quality reached by this simple approach, a number of papers recently proposed to use derived distributional features to represent each pair of words by a large distributional feature vector. Such a vector can be constructed by taking the pairwise sum or pairwise product of the vectors of two words. Now, the similarity between two words can be learned by a supervised classification method. In the following, we will see, how this method can be applied to the first part of the shared task. Since we have feature representations for each pair of words, we can also try to learn several different relations. We will do so for the second part of the task.

The rest of the paper is organized as follows. Section 2 discusses the related works. In section 3 we will have a short look at the data and the shared task. Section 4 explains the distributional feature construction, pairwise feature generation and the classification methods. In section 5 and 6, we present and discuss the results.

2 Related Work

Supervised approaches have not been used extensively in combination with distributional features. Shimizu et al. (2008) used a learned Mahalanobis distance to rank pairs of synonyms and unrelated words. In order to make the learning computationally feasible they reduced the number of context features massively by selecting the most promising features. Hagiwara (2008) follows a different approach. He constructed features to represent each pair of words. Subsequently a Support Vector Machine is used to learn which pairs are pairs of synonyms and which pairs are not. The features for the pair of words are constructed by pairwise addition or multiplication of the features of each word. Similar approaches are followed by Weeds et al. (2014) and Aga et al. (2016).

Proceedings of the Workshop on Cognitive Aspects of the Lexicon,
pages 114–118, Osaka, Japan, December 11-17 2016.

Alternatively, a pair of words can be represented by a small number of features. These features either represent properties of one of both words, or a property of the pair, e.g. the pointwise mutual information of these words. Possible features include also similarity measures based on the co-occurrence features. The use of such a set of features is followed by Bär et al. (2012), Wartena (2013) and Santus et al. (2016) for example. Turney (2014) combines these type of aggregated features with simple co-occurrence features.

3 Task and data

The CogALex shared task consist of two parts. For the first task pairs of words have to be classified as semantically related or not related. For the second task the type of relation for the related pairs has to be classified further into 4 semantic relations. For both tasks a test and a training set is provided.

The training data for both tasks contain 3054 pair of words. From these pairs 826 are semantically related, the remaining 2228 pairs are not related. The test data consists of 4260 pairs, 1201 of which are related and 3059 are not. In contrast to many other datasets, the words in this set are very heterogeneous: the set contains nouns, adjectives, verbs and even pronouns. For the second task, the relation between the related words is classified into 4 classes: synonymy, antonymy, hypernymy and meronymy. Especially, the combination of different part of speech and antonymy gives rises to unexpected pairs of related words, like *burn–cool* or *anger-calm*. Moreover, pairs like *arm–leg* or *vegetable-meat* are considered as anontonyms and hence related words, while other pairs, that are related somewhat more indirectly, like *breast-leg*, *vegetable-apple* (both co-meronyms) or *run-athlete* (we could consider athletics as a hypernym of running) are classified as unrelated. Thus it becomes clear that the dataset is far from trivial and is a real challenge for automatic classification.

For the construction of the context vectors of the words we use the UKWaC-Corpus (Baroni et al., 2009).

4 Methodology and Experiment

In this section, we will explain the task description, the feature construction for the words, and our approach to the task.

4.1 Feature construction

In DS the meaning of a word is represented by a vector of context features. As context features co-occurrence data with other words in a large text corpus are used.

There are a number of choices that have to be made when building the context vectors for each word. In the following we will use the choices that turned out to yield the best results in a number of different tasks in recent studies by Bullinaria and Levy (2007; 2012) and Kiela and Clark (2014).

First it has to be determined what words are used as context features, i.e. for what words co-occurrence statistics have to be computed. Generally, it is found that mid frequency words are most effective. After some preliminary experiments we found that including all words in the frequency range from $4 \cdot 10^3$ to $1 \cdot 10^6$ in the UKWaC Corpus is a good compromise between optimal results and acceptable storage and computing efforts. Therefore, context words which have frequency range from $4 \cdot 10^3$ to $1 \cdot 10^6$ in the UKWaC Corpus have been considered to construct the context vector for each words. Then each word is now represented by a vector of 17 400 features.

Next we have to determine the size of the window for co-occurrence. If the training corpus is large enough all studies show that smaller windows yield better results. We first remove all stop words and then use a window size of two words on the preprocessed text, respecting sentence boundaries. Syntactic relations are not used to determine the context of a word.

Finally, we use positive pointwise mutual information (PPMI) as a feature weighting, since it was shown to give better results than raw co-occurrence probabilities in a number of different studies. For a context words c and a (target) word t the PPMI is defined as

$$ppmi(c,t) = \max\left(\log \frac{p(c|t)}{p(c)}, 0\right). \quad (1)$$

Task	Method	Precision	Recall	F-score	Accuracy
Task 1	All True (Majority)	0.282	1.000	0.440	0.282
	Cosine	0.590	0.713	0.646	0.780
	Supervised (Addition)	0.362	0.094	0.149	0,698
	HsH-Supervised (Multiplication)	0.577	0.593	0.585	0,760
Task 2	All hypernym (Majority)	0.0897	0.318	0.140	0.0897
	HsH-Supervised (Multiplication)	0.506	0.154	0.229	0,753

Table 1: Performance of the HsH-Supervised method and two baselines for both tasks on the test set

4.2 Representation of word pairs

The similarity of words can be computed by comparing their feature vectors. In order to decide whether two words are semantically related, Hagiwara (2008) proposed a novel approach which is learning an SVM model by taking the distributional features as an input, that were constructed by addition of the context vectors of both words. In addition, recently, distributional features have also been used directly to train classifiers that classify pairs of words as being synonymous or not (Weeds et al., 2014; Aga et al., 2016) and showed good performance on the applied tasks. For the shared task, we have also followed this approach which is using distributional features directly on classifiers. To construct the feature vector for each pair of words, we use multiplication. Pairwise multiplication was shown to give good results in (Weeds et al., 2014) and (Aga et al., 2016).

As a baseline we have been considering the classical cosine similarity between the context vectors of the two words. On the training data, the optimal split has been learned between the related and non-related pairs. For the test data, we thus consider pairs with context vectors that have a cosine above 0.0842 to be semantically related.

As a further simple baseline for the first task we use a classifier that considers each pair as semantically related. In fact, this is a type of majority classifier, that always assigns the largest *evaluated* category.[1] For the second task the largest evaluated category in the training data is the hypernym relation (255 pairs). Thus this classifier assigns hypernym to each pair.

4.3 Supervised Similarity Learning

We have used linear SVM from the liblinear package to learn a model and classify the word pairs represented by one feature vector. Liblinear is very efficient and fast for training large-scale problems as showed by Fan Fan et al. (2008). To find the best combination of parameter values for the cost parameter C and the kernel parameter γ we used grid search. We tested for $-5 \leq \log_2 C \leq 15$ and $-15 \leq \log_2 \gamma \leq 2$ n steps of 0.05. Using cross validation on the training data we found $C = 32$ and $\gamma = 0,00781$ as optimal values. The right selection of the hyper-parameters should minimize the risk of overfitting.

5 Results

The results of the supervised method and our two baselines are given in Table 1. For the first task the supervised method based on the Hadamard-Product of context vectors could not give better results than the simple cosine similarity baseline. The multiplication, however, is much better than addition of vectors and also clearly better than the naive baseline, that considers each pair as related.

For the second task the F1-score of the supervised method is very low, but still far above the naive baseline. Remarkably, the precision is quite high: half of the pairs found for one of the four semantic relations indeed have this relation.

[1] As an anonymous reviewer pointed out, in the special case of task 1 the largest class that is taken into account in the evaluation, happens to be the smallest class. Thus this baseline could also be coined "minority classifier".

6 Discussion and conclusions

In a number of papers pairwise multiplication of context vectors has been used to represent pair of words. The feature vectors for the pairs of words created by multiplication (or another operation of two numbers) then is used to train a supervised model that learns whether the words in the pair are semantically related or not. We have applied this method to the CogALex shared task.

At first sight it is quite surprising that the supervised method stays behind the simple cosine similarity approach, since various publications have reported that this method that we applied is slightly better than cosine similarity.

The main reason for the bad performance of the SVM is probably that the model is overfitting the training data. We expected the SVM with carefull selection of the C-parameter to be quite robust against overfitting. In (Aga et al., 2016) we used the same number of features and could improve a lot over the cosine baseline. In the present study, however, the model clearly is overfitting the training data: when we apply the learned model to the training data we get a result with an accuracy of about 99%, showing that the model indeed overfits the training data.

Furthermore, we used a standard SVM that optimizes for overall accuracy, while the official evaluation for the task is the F1-Score of a small class. In fact the accuracy is quite high and the difference in accuracy between the simple cosine based method and the supervised method is very small. For a discussion on the differences between optimizing on F-Score and accuracy see e.g. Ye et al. (2012)

Finally, we have the impression that the method is successful in recognizing a loose semantic relatedness, but is not able to distinguish between very closely related words (like synonyms) and more loosely related words: In Aga et al. (2016), we studied relatedness of terms in a thesaurus. Here the supervised method also performs well on pairs of terms that are related to each other by some thesaurus relations via at most one intermediate concept. The performance is worst on pairs build from alternative labels for the same concept. Here we have a similar situation, in which we only want to find words with a specific and precise defined semantic relation, while other words, that have other or more loose semantic relations are classified as unrelated. Thus it seems that the findings of the present experiment are in-line with previous results for the same approach.

For future work we will apply dimensionality reduction in order to reduce the number of features and to prevent the SVM from overfitting.

References

Rosa Tsegaye Aga, Christian Wartena, Lucas Drumond, and Lars Schmidt-Thieme. 2016. Learning thesaurus relations from distributional features. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. UKP: Computing Semantic Textual Similarity by Combining Multiple Content Similarity Measures. In *Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval-2012)*, pages 435–440.

Marco Baroni, S. Bernardini, A. Ferraresi, and E. Zanchetta. 2009. The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. *Language Resources and Evaluation 43 (3): 209-226*, 43(3):209–226.

John A. Bullinaria and Joseph P. Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behaviour Research Methods*, 39(3):510–526.

John A. Bullinaria and Joseph P. Levy. 2012. Extracting Semantic Representations from Word Co-occurrence Statistics: Stop-lists, Stemming and SVD. *Behaviour Research Methods*, 44(3):890–907.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, June.

Masato Hagiwara. 2008. A Supervised Learning Approach to Automatic Synonym Identification Based on Distributional Features for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA, Student Research

Workshop. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics*, pages 1–6.

Douwe Kiela and Stephen Clark. 2014. A Systematic Study of Semantic Vector Space Model Parameters. In *2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 21–30, Stroudsburg, PA, USA. Association for Computational Linguistics.

Enrico Santus, Alessandro Lenci, Tin-Shing Chiu, Qin Lu, and Chu-Ren Huang. 2016. Nine features in a random forest to learn taxonomical semantic relations. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Nobuyuki Shimizu, Masato Hagiwara, Yasuhiro Ogawa, Katsuhiko Toyama, and Hiroshi Nakagawa. 2008. Metric learning for synonym acquisition. In *COLING '08 Proceedings of the 22nd International Conference on Computational Linguistics*, pages 793–800.

Peter Turney. 2014. Distributional semantics beyond words: Supervised learning of analogy and paraphrase. *Transactions of the Association for Computational Linguistics*, 1:353–366.

Christian Wartena. 2013. HsH: Estimating semantic similarity of words and short phrases with frequency normalized distance measures. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), Atlanta, Georgia, USA*.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

Nan Ye, Kian Ming Adam Chai, Wee Sun Lee, and Hai Leong Chieu. 2012. Optimizing f-measure: A tale of two approaches. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*.