Word Embedding-based Antonym Detection using Thesauri and Distributional Information

Masataka Ono, Makoto Miwa, Yutaka Sasaki
Department of Advanced Science and Technology
Toyota Technological Institute
2-12-1 Hisakata, Tempaku-ku, Nagoya, Japan
{sd12412, makoto-miwa, yutaka.sasaki}@toyota-ti.ac.jp

Abstract

This paper proposes a novel approach to train word embeddings to capture antonyms. Word embeddings have shown to capture synonyms and analogies. Such word embeddings, however, cannot capture antonyms since they depend on the distributional hypothesis. Our approach utilizes supervised synonym and antonym information from thesauri, as well as distributional information from large-scale unlabelled text data. The evaluation results on the GRE antonym question task show that our model outperforms the state-of-the-art systems and it can answer the antonym questions in the F-score of 89%.

1 Introduction

Word embeddings have shown to capture synonyms and analogies (Mikolov et al., 2013b; Mnih and Kavukcuoglu, 2013; Pennington et al., 2014). Word embeddings have also been effectively employed in several tasks such as named entity recognition (Turian et al., 2010; Guo et al., 2014), adjectival scales (Kim and de Marneffe, 2013) and text classification (Le and Mikolov, 2014). Such embeddings trained based on *distributional hypothesis* (Harris, 1954), however, often fail to recognize antonyms since antonymous words, e.g. *strong* and *weak*, occur in similar contexts. Recent studies focuses on learning word embeddings for specific tasks, such as sentiment analysis (Tang et al., 2014) and dependency parsing (Bansal et al., 2014; Chen et al., 2014). These motivate a new approach to learn word embeddings to capture antonyms.

Recent studies on antonym detection have shown that thesauri information are useful in distinguishing antonyms from synonyms. The state-of-the-art systems achieved over 80% in F-score on GRE antonym tests. Yih et al. (2012) proposed a Polarity Inducing Latent Semantic Analysis (PILSA) that incorporated polarity information in two thesauri in constructing a matrix for latent semantic analysis. They additionally used context vectors to cover the out-of-vocabulary words; however, they did not use word embeddings. Recently, Zhang et al. (2014) proposed a Bayesian Probabilistic Tensor Factorization (BPTF) model to combine thesauri information and existing word embeddings. They showed that the usefulness of word embeddings but they used pre-trained word embeddings.

In this paper, we propose a novel approach to construct word embeddings that can capture antonyms. Unlike the previous approaches, our approach directly trains word embeddings to represent antonyms. We propose two models: a Word Embedding on Thesauri information (WE-T) model and a Word Embeddings on Thesauri and Distributional information (WE-TD) model. The WE-T model receives supervised information from synonym and antonym pairs in thesauri and infers the relations of the other word pairs in the thesauri from the supervised information. The WE-TD model incorporates corpus-based contextual information (distributional information) into the WE-T model, which enables the calculation of the similarities among in-vocabulary and out-of-vocabulary words.

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 984–989,
Denver, Colorado, May 31 – June 5, 2015.

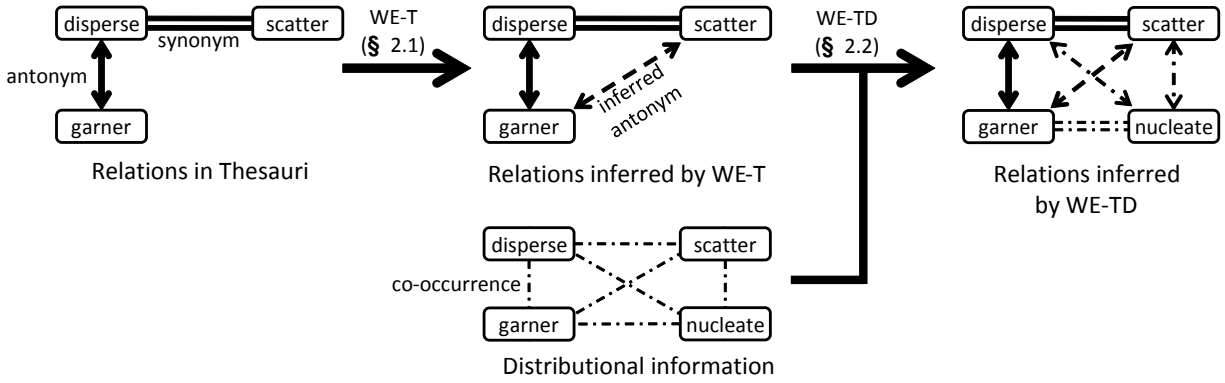

Figure 1: Overview of our approach. When we use the thesauri directly, *disperse* and *garner* are known to be antonymous and *disperse* and *scatter* are known to be synonymous, but the remaining relations are unknown. WE-T infers indirect relations among words in thesauri. Furthermore, WE-TD incorporates distributional information, and the relatedness among in-vocabulary and out-of-vocabulary words (*nucleate* here) are obtained.

2 Word embeddings for antonyms

This section explains how we train word embeddings from synonym and antonym pairs in thesauri. We then explain how to incorporate distributional information to cover out-of-vocabulary words. Figure 1 illustrates the overview of our approach.

2.1 Word embeddings using thesauri information

We first introduce a model to train word embeddings using thesauri information alone, which is called the WE-T model. We embed vectors to words in thesauri and train vectors to represent synonym and antonym pairs in the thesauri. More concretely, we train the vectors by maximizing the following objective function:

$$\begin{aligned} &\sum_{w \in V} \sum_{s \in S_w} \log \sigma(sim(w, s)) \\ &+\alpha \sum_{w \in V} \sum_{a \in A_w} \log \sigma(-sim(w, a)) \end{aligned} \quad (1)$$

V is the vocabulary in thesauri. S_w is a set of synonyms of a word w, and A_w is a set of antonyms of a word w. $\sigma(x)$ is the sigmoid function $\frac{1}{1+e^{-x}}$. α is a parameter to balance the effects of synonyms and antonyms. $sim(w_1, w_2)$ is a scoring function that measures a similarity between two vectors embedded to the corresponding words w_1 and w_2. We use the following asymmetric function for the scoring function:

$$sim(w_1, w_2) = \mathbf{v}_{w_1} \cdot \mathbf{v}_{w_2} + b_{w_1} \quad (2)$$

$\mathbf{v}_w$ is a vector embedded to a word w and b_w is a scalar bias term corresponding to w. This similarity score ranges from minus infinity to plus infinity and the sigmoid function in Equation (1) scales the score into the $[0, 1]$ range.

The first term of Equation (1) denotes the sum of the similarities between synonym pairs. The second term of Equation (1) denotes the sum of the dissimilarities between antonym pairs. By maximizing this objective, synonym and antonym pairs are tuned to have high and low similarity scores respectively, and indirect antonym pairs, e.g., synonym of antonym, will also have low similarity scores since the embeddings of the words in the pairs will be dissimilar. We use AdaGrad (Duchi et al., 2011) to maximize this objective function. AdaGrad is an online learning method using a gradient-based update with automatically-determined learning rate.

2.2 Word embeddings using thesauri and distributional information

Now we explain a model to incorporate corpus-based distributional information into the WE-T model, which is called the WE-TD model.

We hereby introduce Skip-Gram with Negative Sampling (SGNS) (Mikolov et al., 2013a), which the WE-TD model bases on. Levy and Goldberg (2014) shows the objective function for SGNS can

be rewritten as follows.

$$\sum_{w \in V} \sum_{c \in V} \{\#(w,c) \log \sigma(sim(w,c)) + k\#(w) P_0(c) \log \sigma(-sim(w,c))\} \quad (3)$$

The first term represents the co-occurrence pairs within a context window of C words preceding and following target words. $\#(w,c)$ stands for the number of appearances of a target word w and its context c. The second term represents the negative sampling. k is a number of negatively sampled words for each target word. $\#_p(w)$ is the number of appearances of w as a target word, and its negative context c is sampled from a modified unigram distribution P_0 (Mikolov et al., 2013a). We employ the subsampling (Mikolov et al., 2013a), which discards words according to the probability of $P(w) = 1 - \sqrt{\frac{t}{p(w)}}$. $p(w)$ is the proportion of occurrences of a word w in the corpus, and t is a threshold to control the discard. When we use a large-scale corpus directly, the effects of rare words are dominated by the effects of frequent words. Subsampling alleviates this problem by discarding frequent words more often than rare words.

To incorporate the distributional information into the WE-T model, we propose the following objective function, which simply adds this objective function to Equation 1 with an weight β:

$$\begin{aligned} &\beta\{\sum_{w \in V} \sum_{s \in S_w} \log \sigma(sim(w,s)) \\ &+\alpha \sum_{w \in V} \sum_{a \in A_w} \log \sigma(-sim(w,a))\} \\ &+ \sum_{w \in V} \sum_{c \in V} \{\#(w,c) \log \sigma(sim(w,c)) \\ &+k\#(w) P_0(c) \log \sigma(-sim(w,c))\} \end{aligned} \quad (4)$$

This function can be further arranged as

$$\sum_{w \in V} \sum_{c \in V} \{A_{w,c} \log \sigma(sim(w,c)) + B_{w,c} \log \sigma(-sim(w,c))\} \quad (5)$$

Here, the coefficients $A_{w,c}$ and $B_{w,c}$ are sums of corresponding coefficients in Equation 4. These terms can be pre-calculated by using the number of appearances of contextual word pairs, unigram distributions, and synonym and antonym pairs in thesauri.

The objective is maximized by using AdaGrad. We skip some updates according to the coefficients $A_{w,c}$ and $B_{w,c}$ to speed up the computation; we ignore the terms with extremely small coefficients ($< 10^{-5}$) and we sample the terms according to the coefficients when the coefficients are less than 1.

3 Experiments

3.1 Evaluation settings

This section explains the task setting, resource for training, parameter settings, and evaluation metrics.

3.1.1 GRE antonym question task

We evaluate our models and compare them with other existing models using GRE antonym question dataset originally provided by Mohammad et al. (2008). This dataset is widely used to evaluate the performance of antonym detection. Each question has a target word and five candidate words, and the system has to choose the most contrasting word to the target word from the candidate words (Mohammad et al., 2013). All the words in the questions are single-token words. This dataset consists of two parts, development and test, and they have 162 and 950 questions, respectively. Since the test part contains 160 development data set, We will also report results on 790 (950-160) questions following Mohammad et al. (2013).

In evaluating our models on the questions, we first calculated similarities between a target word and its candidate words. The similarities were calculated by averaging asymmetric similarity scores using the similarity function in Equation 2. We then chose a word which had the lowest similarity among them. When the model did not contain any words in a question, the question was left unanswered.

3.1.2 Resource for training

For supervised dataset, we used synonym and antonym pairs in two thesauri: WordNet (Miller, 1995) and Roget (Kipfer, 2009). These pairs were provided by Zhang et al. (2014)[1]. There were 52,760 entries (words), each of which had 11.7 synonyms on average, and 21,319 entries, each of which had 6.5 antonyms on average.

[1] https://github.com/iceboal/word-representations-bptf

	Dev. Set			Test Set (950)			Test Set (790)		
	Prec.	Rec.	F	Prec.	Rec.	F	Prec.	Rec.	F
Encarta lookup[†]	0.65	0.61	0.63	0.61	0.56	0.59	—	—	—
WordNet & Roget lookup[¶]	1.00	0.49	0.66	0.98	0.45	0.62	0.98	0.45	0.61
WE-T	0.92	0.71	0.80	0.90	0.72	0.80	0.90	0.72	0.80
WordNet + Affix heuristics + Adjacent category annotation[§]	0.79	0.66	0.72	—	—	—	0.77	0.63	0.69
WE-D	0.09	0.08	0.09	0.08	0.07	0.07	0.07	0.07	0.07
Encarta PILSA + S2Net + Embedding[†]	0.88	0.87	0.87	0.81	0.80	0.81	—	—	—
WordNet & Roget BPTF[‡]	0.88	0.88	0.88	0.82	0.82	0.82	—	—	—
WE-TD	0.92	0.91	0.91	0.90	0.88	0.89	0.89	0.87	0.88

Table 1: Results on the GRE antonym question task. [†] is from Yih et al. (2012), [‡] is from Zhang et al. (2014), and [§] is from Mohammad et al. (2013). [¶] slightly differs from the result in Zhang et al. (2014) since thesauri can contain multiple candidates as antonyms and the answer is randomly selected for the candidates.

Error Type	Description	# Errors	Example Target	Example Gold	Example Predicted
Contrasting	Predicted answer is contrasting, but not antonym.	7	reticence dussuade	loquaciousness exhort	storm extol
Degree	Both answers are antonyms, but gold has a higher degree of contrast.	3	postulate	verify	reject
Incorrect gold	Gold answer is incorrect.	2	flinch	extol	advance
Wrong expansion	Gold and predicted answers are both in the expanded thesauri.	1	hapless	fortunate	happy
Incorrect	Predicted answer is not contrasting.	1	sessile	obile	ceasing
Total		14	—	—	—

Table 2: Error types by WE-TD on the development set.

We obtained raw texts from Wikipedia on November 2013 for unsupervised dataset. We lowercased all words in the text.

3.1.3 Parameter settings

The parameters were tuned using the development part of the dataset. In training the WE-T model, the dimension of embeddings was set to 300, the number of iteration of AdaGrad was set to 20, and the initial learning rate of AdaGrad was set to 0.03. α in Equation 1 were set to 3.2, according to the proportion of the numbers of synonym and antonym pairs in the thesauri. In addition to these parameters, when we trained the WE-TD model, we added the top 100,000 frequent words appearing in Wikipedia into the vocabulary. The parameter β was set to 100, the number of negative sampling k was set as 5, the context window size C was set to 5, the threshold for subsampling[2] was set to 10^{-8}.

3.1.4 Evaluation metrics

We used the *F-score* as a primary evaluation metric following Zhang et al. (2014). The *F-score* is the harmonic mean of *precision* and *recall*. *Precision* is the proportion of correctly answered questions over answered questions. *Recall* is the proportion of correctly answered questions over the questions.

[2]This small threshold is because this was used to balance the effects of supervised and unsupervised information.

3.2 Results

Table 1 shows the results of our models on the GRE antonym question task. This table also shows the results of previous systems (Yih et al., 2012; Zhang et al., 2014; Mohammad et al., 2013) and models trained on Wikipedia without thesauri (WE-D) for the comparison.

The low performance of WE-D illuminates the problem of distributional hypothesis. Word embeddings trained by using distributional information could not distinguish antonyms from synonyms.

Our WE-T model achieved higher performance than the baselines that only look up thesauri. In the thesauri information we used, the synonyms and antonyms have already been extended for the original thesauri by some rules such as ignoring part of speech (Zhang et al., 2014). This extension contributes to the larger coverage than the original synonym and antonym pairs in the thesauri. This improvement shows that our model not only captures the information of synonyms and antonyms provided by the supervised information but also infers the relations of other word pairs more effectively than the rule-based extension.

Our WE-TD model achieved the highest score among the models that use both thesauri and distributional information. Furthermore, our model has small differences in the results on the development and test parts compared to the other models.

3.3 Error Analysis

We analyzed the 14 errors on the development set, and summarized the result in Table 2.

Half of the errors (i.e., seven errors) were caused in the case that the predicted word is contrasting to some extent but not antonym ("Contrasting"). This might be caused by some kind of semantic drift. In order to predict these gold answers correctly, constraints of the words, such as part of speech and selectional preferences, need to be used. For example, "venerate" usually takes "person" as its object, while "magnify" takes "god." Three of the errors were caused by the degree of contrast of the gold and the predicted answers ("Degree"). The predicted word can be regarded as an antonym but the gold answer is more appropriate. This is because our model does not consider the degree of antonymy, which is out of our focus. One of the questions in the errors had an incorrect gold answer ("Incorrect gold"). We found that in one case both gold and predicted answers are in the expanded antonym dictionary ("Wrong expansion"). In expanding dictionary entries, the gold and predicted answers were both included in the word list of an antonym entries. In one case, the predicted answer was simply wrong ("Incorrect").

4 Conclusions

This paper proposed a novel approach that trains word embeddings to capture antonyms. We proposed two models: WE-T and WE-TD models. WE-T trains word embeddings on thesauri information, and WE-TD incorporates distributional information into the WE-T model. The evaluation on the GRE antonym question task shows that WE-T can achieve a higher performance over the thesauri lookup baselines and, by incorporating distributional information, WE-TD showed 89% in F-score, which outperformed the conventional state-of-the-art performances. As future work, we plan to extend our approaches to obtain word embeddings for other semantic relations (Gao et al., 2014).

References

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 809–815, Baltimore, Maryland, June. Association for Computational Linguistics.

Wenliang Chen, Yue Zhang, and Min Zhang. 2014. Feature embedding for dependency parsing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 816–826, Dublin, Ireland, August. Dublin City University and Association for Computational Linguistics.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, July.

Bin Gao, Jiang Bian, and Tie-Yan Liu. 2014. Wordrep: A benchmark for research on learning word representations. *ICML 2014 Workshop on Knowledge-Powered Deep Learning for Text Mining*.

Jiang Guo, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Revisiting embedding features for simple semi-supervised learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 110–120, Doha, Qatar, October. Association for Computational Linguistics.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(23):146–162.

Joo-Kyung Kim and Marie-Catherine de Marneffe. 2013. Deriving adjectival scales from continuous space word representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1625–1630, Seattle, Washington, USA, October. Association for Computational Linguistics.

Barbara Ann Kipfer. 2009. *Roget's 21st Century Thesaurus*. Philip Lief Group, third edition edition.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196. JMLR Workshop and Conference Proceedings.

Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2177–2185. Curran Associates, Inc.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

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.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41, November.

Andriy Mnih and Koray Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2265–2273. Curran Associates, Inc.

Saif Mohammad, Bonnie Dorr, and Graeme Hirst. 2008. Computing word-pair antonymy. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 982–991, Honolulu, Hawaii, October. Association for Computational Linguistics.

Saif M. Mohammad, Bonnie J. Dorr, Graeme Hirst, and Peter D. Turney. 2013. Computing lexical contrast. *Computational Linguistics*, 39(3):555–590, September.

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)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1555–1565, Baltimore, Maryland, June. Association for Computational Linguistics.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 384–394, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wen-tau Yih, Geoffrey Zweig, and John Platt. 2012. Polarity inducing latent semantic analysis. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1212–1222, Jeju Island, Korea, July. Association for Computational Linguistics.

Jingwei Zhang, Jeremy Salwen, Michael Glass, and Alfio Gliozzo. 2014. Word semantic representations using bayesian probabilistic tensor factorization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1522–1531. Association for Computational Linguistics.