Structural-fitting Word Vectors to Linguistic Ontology for Semantic Relatedness Measurement

Yang-Yin Lee
Department of Computer Science & Information Engineering, National Taiwan University
yylee@nlg.csie.ntu.edu.tw

Ting-Yu Yen
Department of Computer Science & Information Engineering, National Taiwan University
tyyen@nlg.csie.ntu.edu.tw

Hen-Hsen Huang
Department of Computer Science & Information Engineering, National Taiwan University
hhhuang@nlg.csie.ntu.edu.tw

Hsin-Hsi Chen
Department of Computer Science & Information Engineering, National Taiwan University
hhchen@ntu.edu.tw

ABSTRACT

With the aid of recently proposed word embedding algorithms, the study of semantic relatedness has progressed and advanced rapidly. In this research, we propose a novel *structural-fitting* method that utilizes the linguistic ontology into vector space representations. The ontological information is applied in two ways. The fine2coarse approach refines the word vectors from fine-grained to coarse-grained terms[1] (word types), while the coarse2fine approach refines the word vectors from coarse-grained to fine-grained terms. In the experiments, we show that our proposed methods outperform previous approaches in seven publicly available benchmark datasets.

KEYWORDS

Word embedding; semantic relatedness; linguistic ontology; structural-fitting; retrofitting

1 INTRODUCTION

The distributed representation of word (word embedding) has drawn great interests in recent years due to their abilities to acquire syntactic and semantic information from large unannotated corpora [1–3]. The research community quickly observed the effectiveness of word embedding for semantic relatedness measurement, one of fundamental natural language processing tasks that predicts the similarity between a pair of words. More recently, the research of combining word embedding and linguistic resource is gaining strength, exploring the usage of linguistic resources such as WordNet [4], FreeBase [5] and the paraphrase database (PPDB) [6, 7] on tasks such as word similarity [8] and sentiment analysis [9].

This paper differs from previous works in that it employs linguistic ontology in a gradual way into a trained word embedding. Our proposed *structural-fitting* is a post-processing method for generating low-dimensional word embedding in the spirit of *retrofitting* [10]. Although some researches adopted the semantic relationship (e.g., *synonym*, *antonyms*, etc.) into word embedding [11], the granularity of the relationship in ontology is not considered. For example, in PPDB the terms *automobile*, *car* and *wagon* are in the same coarse-grained paraphrase set. However, it is clear that the pair (*automobile*, *car*) is more similar than (*automobile*, *wagon*). Different collecting criteria of the paraphrase set can result in different recall-precision tradeoffs. The fine-grained collection of paraphrases usually carries high precision but low recall (e.g., only *automobile* and *car*), whereas the coarse-grained collection of paraphrases usually results in low precision but high recall (e.g., *automobile*, *car* and *wagon*). Our proposed models, fine2coarse and coarse2fine, come from the idea that the word vectors should be gradually retrofitted from fine-to-coarse or from coarse-to-fine manner. Intuitively, when first applying the fine-grained retrofitting, highly synonymous word vectors should become closer to each other, such as (*automobile*, *car*). At this point, however, some words that are only moderately similar are not retrofitted yet, such as (*automobile*, *wagon*). After running the coarse-grained retrofitting, the word pair (*automobile*, *wagon*) will be closer to each other. However, it is not as close as (*automobile*, *car*), since (*automobile*, *car*) retrofitted twice in this scheme. In coarse2fine, all the words *automobile*, *car* and *wagon* will be retrofitted in the first run. However, only *automobile* and *car* will be retrofitted in the fine-grained run. As a result, the strength of the synonymous relationship can be learned. Of course, our proposed model can be extended to other relationships given a similar scheme.

By utilizing structural-fitting to GloVe word embedding, we show that our proposed methods can outperform previous approaches in publicly available English semantic relatedness datasets, including MEN [12], RG65 [13], WordSim-353 (WS353) [14], SimLex-999 (SL999) [15], MTurk [16] and Rare Words (RW) [17]. We also test our methods in a Chinese WordSim dataset CWS297 [18] using Tongyici Cilin [19]. In CWS297, we show that the improvement ratio in Chinese dataset is larger than that in English datasets when utilizing the hierarchical synonym ontology Tongyici Cilin.

2 STRUCTURAL-FITTING OF WORD VECTORS

Let $V = \{w_1, \dots, w_n\}$ be a vocabulary of a trained word embedding and $|V|$ be its size. Let $\Omega = \{\Omega_1, \dots, \Omega_m\}$ be a fine-to-coarse ontology of m layers. Each layer can be represented as an undirected

[1]Since we do not consider the phrase level structural-fitting in this research, the term *word* and *term* are used interchangeably.

CIKM'17 , November 6–10, 2017, Singapore

ACM ISBN 978-1-4503-4918-5/17/11...$15.00
https://doi.org/10.1145/3132847.3133152

graph (V_l, E_l) for $1 \leq l \leq m$ with $V_1 \subseteq \cdots \subseteq V_m$ and $E_1 \subseteq \cdots \subseteq E_m$. For each term w_i, the edge $(w_i, w_j) \in E_l \subseteq V_l \times V_l$ indicating a semantic relationship of interest (e.g., paraphrase). In Ω, Ω_1 contains an ontology with finest relationship rules and has the lowest recall, but the highest average precision, while Ω_m contains an ontology with coarsest relationship rules and has the highest recall, but the lowest average precision.

2.1 Fine2coarse Approach

The *retrofitting* model is a recently proposed learning framework to run belief propagation on a graph constructed from lexicon-derived relational information to update word embedding. The matrix $\hat{Q}$ will be the pre-trained collection of vector representations $\hat{q}_i \in \mathbb{R}^d$, where d is the length of a word vector. Each $w_i \in V$ is learned using a standard word embedding technique (e.g., GloVe [2] or word2vec [1]). The objective of retrofitting is to learn a new matrix $Q = (q_1, \ldots, q_n)$ such that the word vectors are close to its adjacent vertices, meanwhile constraining the distance between the pre-trained and the new word vectors. The objective to be minimized for a given ontology layer Ω_l is:

$$\Psi(Q_l) = \sum_{i=1}^{n} \left[\alpha_i \|q_i - \hat{q}_i\|^2 + \sum_{(i,j) \in E_l} \beta_{ij} \|q_i - q_j\|^2 \right] \quad (1)$$

where α and β values control the relative strengths of associations. The updating formula from $\hat{q}_i$ to q_i would be:

$$q_i^l = \frac{\sum_{j:(i,j) \in E_l} \beta_{ij} q_j + \alpha_i \hat{q}_i}{\sum_{j:(i,j) \in E_l} \beta_{ij} + \alpha_i} \quad (2)$$

for the lth layer. More details of the derivation of the formula can be found in [10]. Then the ontological information is updated gradually and iteratively. A formal description of our proposed structural-fitting method is shown in Algorithm 1. In our algorithm, we also introduce a discounting parameter γ to control the strength of each layer.

Algorithm 1 Fine2coarse

Input: A pre-trained word embedding $\hat{Q}$, a fine-to-coarse ontology of m layers Ω, hyper-parameters α, β and γ, number of iterations max_it.

Output: A structural-fitted word embedding Q

```
for l = 1 to m do
    V' = V_l
    for it = 1 to max_it do
        for i = 1 to |V'| do
            q_i^l ← (Σ_{j:(i,j)∈E_l} γ_l β_ij q_j + α_i q̂_i) / (Σ_{j:(i,j)∈E_l} γ_l β_ij + α_i)
return Q
```

2.2 Coarse2fine Approach

The coarse2fine approach retrofits the word vectors in a reverse way. The algorithm for coarse2fine is similar to Algorithm 1. In coarse2fine, the main difference is that the ontological information is adopted from the coarsest layer to the finest layer whilst the other steps remain the same.

Experimentally, 10 iterations are sufficient to minimize the objective function from a set of starting vectors to produce effective structural-fitted vectors.

3 BENCHMARK DATASETS

We downloaded six word similarity benchmark data sets from the web: RG65 [13], WS353 [14], MEN [12], SL999 [15], MTurk [16] and RW [17]. Published over 50 years, RG65 can be considered the most widely adopted semantic relatedness benchmark. Note that WS353 contains all the word pairs from RG65.

For evaluating Chinese structural-fitting, CWS297 [18] is applied. CWS297 is a transcription from English WS353 by two undergraduate students with excellent English understandings. The similarity scores containing in CWS297 were re-scored by twenty native Chinese speakers and used in Semeval-2012 [20]. This dataset consists of 297 word pairs. In order to perform a fair comparison, other parameters remain the same as the default settings in English.

4 EXPERIMENTS

This research considers cosine similarity for computing word similarity between two word vectors [8, 10, 11]. The metric used in this paper is Spearman correlation coefficient. In the experiments, if a test dataset has missing words (the words that do not appear in the word embedding), we remove those missing words from the dataset. Note that our reported results of vanilla word embedding may be slightly different from other papers due to the treatment of missing words and the similarity computation method. Some researches use zero vector to represent the missing words, whereas some use the average of all word vectors of the word embedding to represent the missing word. However, within this research the reported performance can be compared due to the same similarity computation method and the same missing word processing method.

GloVe. The main word embedding used in this research, GloVe, is a log-bilinear regression model that tries to resolve the drawbacks of local context window approaches (e.g., skip-gram model [1]) and global factorization approaches (e.g., latent semantic analysis) on word analogy and semantic relatedness tasks. The global vectors in GloVe are trained using unsupervised learning on aggregated global word-word co-occurrence statistics from a corpus. GloVe utilizes the probability ratio derived from the co-occurrence matrix to capture the relatedness between words. The objective of GloVe is to factorize the log-count matrix and to find the word embedding that satisfies this ratio.

For English structural-fitting, GloVe pre-trained word vectors are used as input. The word vectors were trained on 6 billion tokens from Wikipedia 2014 + Gigaword 5. The linguistic resource used in this study is PPDB [6, 7]. PPDB is an automatically created massive resource of paraphrases. In our selected lexical level English PPDB, each pair of words is semantically equivalent in some degree. Following the resource used in retrofitting, we use PPDB 1.0's

Table 1: Spearman (ρ) correlation of six English semantic relatedness datasets of structural-fitting with GloVe

	GloVe.6B.50d						GloVe.6B.300d
	RG65	MEN	WS353	SL999	MTurk	RW	WS353
GloVe	0.595	0.652	0.496	0.265	0.619	0.340	0.601
r1-ppdb (xl)	0.689	0.686	0.515	0.399	0.651	0.357	0.632
r1-ppdb (m)	0.614	0.660	0.511	0.298	0.630	0.356	0.610
r1-ppdb (s)	0.618	0.662	0.510	0.292	0.629	0.349	0.610
fine2coarse (m-xl)	0.692	0.689	0.528	0.394	**0.659**	**0.366**	0.634
fine2coarse (s-xl)	0.693	**0.690**	0.530	0.392	0.658	0.362	0.638
coarse2fine (xl-m)	0.700	0.688	0.525	0.411	0.657	0.361	0.634
coarse2fine (xl-s)	**0.703**	0.689	0.525	0.410	0.658	0.358	0.638
fine2coarse (s-m-xl)	0.684	0.687	**0.533**	0.378	0.656	0.365	**0.640**
coarse2fine (xl-m-s)	0.699	0.688	0.527	**0.413**	0.658	0.361	0.634

XL version (**xl**) [7], which contains over 500k paraphrases derived from a large collection of texts. In 2015, a new version of PPDB was released [6]. This latest version of resource is integrated with our proposed models as well in the following sizes: M (**m**, over 400k paraphrases) and S (**s**, over 200k paraphrases). In the 2-layers experiment, we set the first γ to 1 and the second to 0.7, whereas in the 3-layers experiment, γs are set to 1, 0.7 and 0.5, respectively. In order to ensure a careful comparison, for α and β we follow the parameter usage in retrofitting. All α_i set to 1 and β_{ij} to be $\text{degree}(i)^{-1}$.

For training Chinese word embedding, Chinese Gigaword[2] (CGW) corpus is adopted. CGW is a 4GB raw text acquired from Chinese news by Linguistic Data Consortium[3] (LDC). Jieba[4] Chinese Text Segmentation toolkit was employed to perform the Chinese word segmentation on CGW. We trained 50-, 100-, 200-, and 300-dimensional GloVe word embeddings using CGW. In the structural-fitting experiments, the Chinese synonym dataset Tongyici Cilin was applied [19]. Developed by Harbin Institute of Technology Center for Information Retrieval, Tongyici Cilin organizes words in a hierarchical structure. With around 90k terms, each term has been assigned a seven-bit code for five levels. All the terms in the same class of the fifth level category (7 bits) can be regarded as of similar meaning (the finest grained level), while the first level category (1 bit) is the coarsest grained level. In the experiments, we use the top 7, 5 and 4 bits in the 3-layers model and the top 7 and 5 bits in the 2-layers model.

5 RESULTS AND DISCUSSION

Table 1 shows the performance of the English structural-fitting models with GloVe.6B.50d. Besides the 50d version, we also show the result of GloVe.6B.300d on WS353. Bold scores are best overall. We list the results of the vanilla GloVe word embedding (row 1), its retrofitted word embedding (r1-ppdb [10], row 2 to 4) and the structural-fitted word embedding (rows 5 to 10). In our experiments, the xl-m-s model achieves robust performance on different datasets and is the best model on average. In general, the structural-fitted models outperform the models that only run retrofitting once. We also find that coarse2fine is better than fine2coarse, but the performance difference between them is relatively small. In WS353, our results show that the GloVe.6B.50d model is more effective than the GloVe.6B.300d model (i.e., the improvement ratio of 50d is larger than that of 300d). Our hypothesis is that the GloVe.6B.300d uses more dimensionality, which may already contain more information than GloVe.6B.50d, so the beneficial of GloVe.6B.300d from structural-fitting is smaller.

Table 2: ρ of CWS297 of Chinese structural-fitting

	50	100	200	300
GloVe	0.488	0.505	0.499	0.505
r1-TongyiciCilin	0.543	0.544	0.531	0.530
fine2coarse 2 layers	**0.555**	0.552	0.542	0.534
fine2coarse 3 layers	0.553	**0.557**	**0.555**	**0.549**
coarse2fine 2 layers	0.550	0.543	0.527	0.529
coarse2fine 3 layers	0.552	0.544	0.536	0.534

Table 2 shows the results of Chinese structural-fitting. The method of using retrofitting is listed in row 2 (r1-TongyiciCilin). Compared to English structural-fitting, the performance gain is more significant. In addition, the 3-layers approach performs better than the 2-layers approach. We suspect that the reason might be Tongyici Cilin is a well-structured ontology with clear lexical hierarchy. We also found that when the dimensionality is smaller, the performance gain is larger. This phenomenon happens in both English and Chinese datasets. Finally, the fine2coarse approach performs better than the coarse2fine approach. We hypothesize that this occurs because the words in the fine-grained Tongyici Cilin are more critical and thus carefully retrofitted at the beginning. Then in the coarse-grained steps, the influence of moderate synonymous words are smaller due to the decaying of γ.

For the parameter sensitivity analysis, Figure 1 shows the ρ of fine2coarse and coarse2fine of 2 layers on CWS297 with GloVe 100d under different γs (the second γ). Retrofitting models of using the fine-grained Cilin (r1-fine) and coarse-grained Cilin (r1-coarse) are compared. When $\gamma = 0$ the structural-fitting model degenerates to retrofitting's method. As can be seen, the structural-fitting models can further improve the word similarity tasks under different γs, showing the benefit of using structural-fitting. Fine2Coarse performs

[2] https://catalog.ldc.upenn.edu/LDC2009T14
[3] https://www.ldc.upenn.edu/
[4] https://github.com/fxsjy/jieba

well when γ is in the range of 0.4–0.5. The performances in large γs are not the best, indicating that the coarse-grained information may contain some noise (words that are only moderate synonymous). In coarse2fine, the best performance is around $\gamma = 0.8$. Different from fine2coarse, coarse2fine performs well when γ is large, showing the effectiveness of giving more strength to a carefully created synonym ontology.

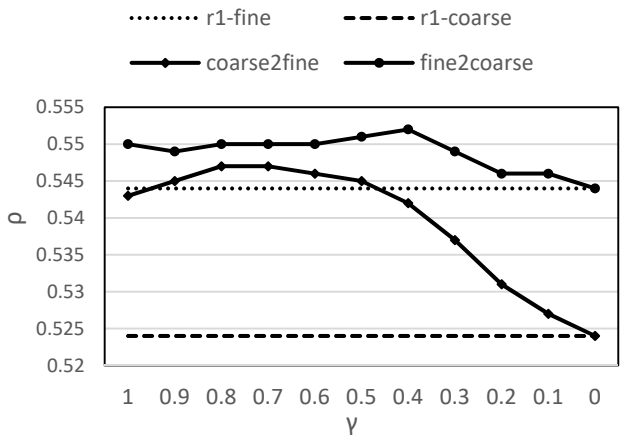

Figure 1: Selection of γ.

6 CONCLUSIONS AND FUTURE WORKS

This paper proposes a novel *structural-fitting* model that takes into account the structural information of a given ontology. The ontological information is applied in two ways. The fine2coarse approach refines the word vectors from fine-grained to coarse-grained terms, while the coarse2fine approach refines the word vectors from coarse-grained to fine-grained terms. In the experiments, we show that our proposed methods outperform previous approaches in several publicly available benchmark datasets. Since only the paraphrase and synonym relationships are considered in our proposed model currently, the performance of using antonym or other relationships remain unknown. In the future, we would like to test our model with contrasting information to see if structural-fitting can benefit from it. Furthermore, whether better performance could be achieved by structural-fitting the entire system with a joint loss rather than current post-processing approach is another issue that worth exploration.

ACKNOWLEDGMENTS

This research was partially supported by Ministry of Science and Technology, Taiwan, under grants MOST-104-2221-E-002-061-MY3, MOST-105-2221-E-002-154-MY3 and MOST-106-2923-E-002-012-MY3, and National Taiwan University under grant NTUCCP-106R891305.

REFERENCES

[1] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado and J. Dean 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* (2013), 3111–3119.

[2] J. Pennington, R. Socher and C.D. Manning 2014. Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*. 12, (2014), 1532–1543.

[3] W. Yih, G. Zweig and J.C. Platt 2012. Polarity inducing latent semantic analysis. *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (2012), 1212–1222.

[4] G.A. Miller 1995. WordNet: a lexical database for English. *Communications of the ACM*. 38, 11 (1995), 39–41.

[5] K. Bollacker, C. Evans, P. Paritosh, T. Sturge and J. Taylor 2008. Freebase: a collaboratively created graph database for structuring human knowledge. *Proceedings of the 2008 ACM SIGMOD international conference on Management of data* (2008), 1247–1250.

[6] E. Pavlick, P. Rastogi, J. Ganitkevich and C.C.-B. Ben Van Durme 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL 2015)* (Beijing, China, Jul. 2015).

[7] J. Ganitkevitch, B. Van Durme and C. Callison-Burch 2013. PPDB: The Paraphrase Database. *HLT-NAACL* (2013), 758–764.

[8] J. Goikoetxea, E. Agirre and A. Soroa 2016. Single or Multiple? Combining Word Representations Independently Learned from Text and WordNet. *AAAI* (2016), 2608–2614.

[9] J. Wieting, M. Bansal, K. Gimpel, K. Livescu and D. Roth 2015. From Paraphrase Database to Compositional Paraphrase Model and Back. *Transactions of the Association for Computational Linguistics*. 3, (2015), 345–358.

[10] M. Faruqui, J. Dodge, S.K. Jauhar, C. Dyer, E. Hovy and N.A. Smith 2015. Retrofitting word vectors to semantic lexicons. *Proc. of NAACL*. (2015).

[11] N. Mrkšić, D.Ó. Séaghdha, B. Thomson, M. Gašić, L. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen and S. Young 2016. Counter-fitting Word Vectors to Linguistic Constraints. *Proceedings of HLT-NAACL* (2016).

[12] E. Bruni, N.-K. Tran and M. Baroni 2014. Multimodal Distributional Semantics. *J. Artif. Intell. Res.(JAIR)*. 49, (2014), 1–47.

[13] H. Rubenstein and J.B. Goodenough 1965. Contextual correlates of synonymy. *Communications of the ACM*. 8, 10 (1965), 627–633.

[14] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman and E. Ruppin 2001. Placing search in context: The concept revisited. *Proceedings of the 10th international conference on World Wide Web* (2001), 406–414.

[15] F. Hill, R. Reichart and A. Korhonen 2016. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*. (2016).

[16] K. Radinsky, E. Agichtein, E. Gabrilovich and S. Markovitch 2011. A word at a time: computing word relatedness using temporal semantic analysis. *Proceedings of the 20th international conference on World wide web* (2011), 337–346.

[17] T. Luong, R. Socher and C.D. Manning 2013. Better Word Representations with Recursive Neural Networks for Morphology. *CoNLL* (2013), 104–113.

[18] L. Qiu, Y. Zhang and Y. Lu 2015. Syntactic Dependencies and Distributed Word Representations for Analogy Detection and Mining. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015* (2015), 2441–2450.

[19] J. Lin 1983. *Tongyici cilin*. Shanhai cishu.

[20] P. Jin and Y. Wu 2012. Semeval-2012 task 4: evaluating chinese word similarity. *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation* (2012), 374–377.