ITNLP-AiKF at SemEval-2017 Task 1: Rich Features Based SVR for Semantic Textual Similarity Computing

Wenjie Liu, Chengjie Sun, Lei Lin and Bingquan Liu
School of Computer Science and Technology
Harbin Institute of Technology
Harbin, China
{wjliu, cjsun, linl, liubq}@insun.hit.edu.cn

Abstract

Semantic Textual Similarity (STS) devotes to measuring the degree of equivalence in the underlying semantic of the sentence pair. We proposed a new system, ITNLP-AiKF, which applies in the SemEval 2017 Task1 Semantic Textual Similarity track 5 English monolingual pairs. In our system, rich features are involved, including Ontology based, word embedding based, Corpus based, Alignment based and Literal based feature. We leveraged the features to predict sentence pair similarity by a Support Vector Regression (SVR) model. In the result, a Pearson Correlation of 0.8231 is achieved by our system, which is a competitive result in the contest of this track.

1 Introduction

Semantic Evaluation (SemEval) contest devotes to pushing the research of semantic analysis, which attracts many participants and promote a lot of groundbreaking achievements in natural language processing (NLP) field. Semantic textual similarity (STS) task works for computing word and text semantics, which has made extensive attraction to the researchers and NLP community since SemEval 2012 (Agirre et al., 2012).

In STS 2017, The organizers added monolingual Arabic and Cross-lingual Arabic-English semantic calculation in order to increase the difficulty in the contest. The task definition is given two sentences participating systems are asked to predict a continuous similarity score on a scale from 0 to 5 of the sentence pair, with 0 indicating that the semantics of the sentences completely independent and 5 semantic equivalence. The evaluation criterion uses Pearson Correlation Coefficient, which computing the correlation between golden standard scores and semantic system predicted scores.

In our system, in order to predict similarity score of two sentences, we trained a Support Vector Regression (SVR) model with abundant features including Ontology based features, Word Embedding based features, Corpus based features, Alignment based features and Literal based features. All the English training, trial and evaluation data set released by previous STS tasks in SemEval were used to construct our system. Our best system achieved 0.8231 Pearson Correlation coefficient in the SemEval 2017 evaluation data set, and the winner achieved 0.8547.

2 Feature Engineering

In our system, many features are tried to promote the performance of our system. Five kinds of features are used: Ontology based features, Word Embedding based features, Corpus based features, Alignment based features and Literal based features.The following is a detailed description of the five kinds features.

2.1 Ontology Based Features

WordNet (Miller, 1995) is used to exploit Ontology based features. WordNet is a large lexical database of English. In WordNet, nouns, verbs, adjectives and adverbs are divided into sets of cognitive synonyms called synsets. Each synonym expresses a distinct concept. WordNet measures two words similarity based on Path_similarity, Res_similarity, Lin_similarity, Wup_similarity, Lch_similarity and so on. In our system, we directly use WordNet APIs provided by NLTK toolkit (Bird, 2006) to calculate the similarity of two words.

Path_similarity measure is based on the shortest path similarity measure. The Path_similarity for-

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 159–163,
Vancouver, Canada, August 3 - 4, 2017.

mula is defined as Eq 1:

$$Sim_{path}(c_1, c_2) = 2 * deep\_max - len(c_1, c_2) \quad (1)$$

where c_1 and c_2 are concepts, *deep_max* is a fixed value of the WordNet and len(c_1,c_2) is the shortest path of concepts c_1 an c_2 in WordNet.

Lch_similarity (Leacock et al., 1998) measure two words similarity by using the depth of concepts in the WordNet hierarchy tree. The Lch_similarity formula is as Eq 2:

$$Sim_{lch}(c_1, c_2) = -log(\frac{len(c_1, c_2)}{2 * deep\_max}) \quad (2)$$

Res_similarity (Resniks Measure) calculates similarity based on two concepts common information content in the taxonomy. The Res_similarity formula is defined as Eq 3:

$$\begin{aligned} Sim_{res}(c_1, c_2) &= -\log P(lso(c_1, c_2)) \\ &= IC(lso(c_1, c_2)) \end{aligned} \quad (3)$$

where $lso(c_1, c_2)$ is the lowest subsumer of concepts c_1 and c_2 in the taxonomy. The value of Lch_similarity and Res_similarity is not in $[0, 1]$, so we need to scale features into $[0, 1]$.

Lin_similarity (Lin, 1998) considers the similarity depending on the commonality and differences of the information contained in the different meaning concepts. The Lin_similarity formula is defined as Eq 4:

$$Sim_{lin}(c_1, c_2) = \frac{2 * IC(lso(c_1, c_2))}{IC(c_1) + IC(c_2)} \quad (4)$$

Wup_similarity (Wu and Palmer, 1994) measures similarity based on the path of conception node, shared parent node and root node. The Wup_similarity formula is defined as Eq 5:

$$\begin{aligned} &sim_{wup}(c_1, c_2) = \\ &\frac{2 * depth(lso(c_1, c_2))}{len(c_1, c_2) + 2 * depth(lso(c_1, c_2))} \end{aligned} \quad (5)$$

We can use two vectors S_1 and S_2 to represent two sentences. For each word in S_1 or S_2, search for the most similar word in another sentence by above methods. For S_1, add all elements together, which are divided by the length of S_1, and then get the value of V_1. Do the same calculation for S_2, and then get the value of V_2. Computing the harmonic mean (denoted by harmonic_mean) of V_1 and V_2, and the result is the similarity of the two sentences. The harmonic mean is defined as Eq 6:

$$harmonic\_mean = \frac{2}{\frac{1}{V_1} + \frac{1}{V_2}} \quad (6)$$

2.2 Word Embedding Based Features

Word Embedding maps words or phrases from defined vocabulary with dense vectors of real values, which have been applied as features in document classification (Sebastiani, 2002), question answering (Tellex et al., 2003), and named entity recognition (Turian et al., 2010) tasks. In our system, we obtained word vectors using two kinds of unsupervised models: Word2Vec (Mikolov et al., 2013) and Global Vectors (GloVe) (Pennington et al., 2014), which can produce high-quality word vectors from millions of corpus data. With the obtained word vectors, the following sentences similarities are calculated: W2V_similarity, IDFV_similarity, S2V_similarity, Text_similarity, WFSV_similarity.

In order to get a better word vector, we used full Wikipedia English corpus to train Word2Vec vectors (400 dimensions) and the Global vector of twitter (200 dimensions) provided by GloVe.

W2V_similarity measures two sentences similarity by using word vectors. The W2V_similarity formula is defined as Eq 7:

$$\begin{aligned} W2V\_Sim(S_1, S_2) = Cos\_Dis(&\frac{\sum_{w \in S_1} W2V(w)}{len(S_1)} \\ &, \frac{\sum_{w \in S_2} W2V(w)}{(len(S_2)}) \end{aligned} \quad (7)$$

where $W2V(w)$ is the word embedding vector, and $len(S_1)$, $len(S_2)$ is the length of sentence.

The cosine similarity is defined as Eq 8:

$$Cos\_Dis(V_1, V_2) = \frac{V_1 \cdot V_2}{\|V_1\| \cdot \|V_2\|} \quad (8)$$

S2V_similarity is another method that measures two sentences similarity directly, by using the following formula as Eq 9:

$$S2VSim(S_1, S_2) = \frac{1}{\frac{len(S_1)}{\sum_{w \in S_1} maxSim(w,S_2)} + \frac{len(S_2)}{\sum_{w \in S_2} maxSim(w,S_1)}} \tag{9}$$

maxSim(w,S) is to find the maximum similarity value between one word in one sentence and all words in another sentence, which is defined as Eq 10.

$$maxSim(w, S) = Max\{Cos\_Dis(W2V(w), W2V(w_s)), w_s \in S\} \tag{10}$$

Text_similarity uses maxSim method and the weight of tf-idf to calculate the pair of sentence. Text_similarity measures (Mihalcea et al., 2006) two sentences similarity uses the following formula as Eq 11:

$$\begin{aligned} &Text\_sim(S_1, S_2) \\ &= \frac{1}{2}(\frac{\sum_{w \in S_1}(maxSim(w, S_2) * idf(w))}{\sum_{w \in S_1} idf(w)} \\ &+ \frac{\sum_{w \in S_2}(maxSim(w, S_1) * idf(w))}{\sum_{w \in S_2} idf(w)}) \end{aligned} \tag{11}$$

IDF_W2V similarity and Freq_W2V similarity represent sentence vector with word embedding, word frequency and word tf-idf. IDF_W2V similarity and Freq_W2V similarity formula are as Eq 12 and Eq 13:

$$IDFV(S) = \sum_{w \in S} IDF(w) * \frac{W2V(w)}{norm(W2V(w))} \tag{12}$$

$$WFSV(S) = \sum_{w \in S} WF(w) * \frac{W2V(w)}{norm(W2V(w))} \tag{13}$$

where IDF(w) and WF(w) are the word tf-idf and frequency based on all Wikipedia english corpus.

After getting the sentence vectors, comput cosine distance between two vectors and the value is a feature of two sentences.

2.3 Corpus Based Features

Latent semantic analysis (LSA) is a technique of global matrix factorization methods, to analyse the relationships between a set of documents and the words. Based on optimal vector space structure, LSA method can leverage statistical information efficiently, and be always used to measure word-to-word similarity.

There are several publicly available tools to construct LSA models, such as SemanticVectors Package (Widdows and Ferraro, 2008) and S-Space Package (Jurgens and Stevens, 2010) can be used to generate LSA space vectors. For this part, we directly use the word vectors provided by SEMILAR[1] (Stefanescu et al., 2014) to calculate the features: W2V_LSI_similarity, S2V_LSI_similarity, Text_LSI_similarity, IDF_LSI_similarity, WFSV_LSI_similarity.

2.4 Alignment Based Features

Alignment similarity based on monolingual alignment measures sentences similarity. Alignment try to discover similar meaning word pairs by exploiting the semantic and contextual similarities. In our work, we directly use the monolingual word aligner provided by (Sultan et al., 2014a,b). Alignment similarity uses the following formula Eq 14:

$$sts(S_1, S_2) = \frac{n_c^a(S_1) + n_c^a(S_2)}{n_c(S_1) + n_c(S_2)} \tag{14}$$

where $n_c^a(S_1)$ and $n_c^a(S_2)$ is the amount of word alignment in two sentences, and $n_c(S_1)$ and $n_c(S_2)$ is the length of sentence.

2.5 Literal Based Features

For literal similarity, we use the edit distance and jaccard distance to calculate sentences similarity. Edit distance also known as Levenshtein Distance, is the minimum step of editing operations from one sentence to another.

Firstly, for jaccard distance, we extracted part-of-speech tagging of each word from a sentence. Then calculate jaccard distance by using the formula defined by Eq 15:

$$J(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|} \tag{15}$$

where S_1 and S_2 are the tag of each word in a sentence, which ignores the order. We use the NLTK toolkit part-of-speech tagging.

[1] http://www.semanticsimilarity.org/

	Ans-Ans	**Qus-Qus**	**HDL**	**Postediting**	**Plagiarism**
Ontology Based	**0.5926**	**0.6041**	0.7192	0.8136	0.7349
Word2vec Based	0.5838	0.6012	**0.7395**	**0.8233**	**0.8053**
GloVe Based	0.5360	0.5827	0.7172	0.7508	0.7478
Corpus Based	0.3737	0.4378	0.6157	0.7334	0.7356
Alignment Based	0.4842	0.5827	0.7172	0.7508	0.7478
Literal Based	0.4860	0.5232	0.6715	0.8108	0.7339
All	0.6248	0.6315	0.8106	0.8307	0.8132

Table 1: The Pearson Correlation on SemEval 2016 evaluation data sets.

3 Experiments and Results

In our system, We build our data set by collecting all off-the-shelf English data sets which released by prior STS evaluations (except the evaluation data set of STS 2016). After that, 80% data set are used as train data set and 20% as valid data set. In our system, we trained SVR model, and the SVR parameters are set as Table 2.

parameter	**kernel**	**C**	**gamma**	**epsilon**
value	rbf	0.1	auto	0.0

Table 2: parameter setting in SVR.

Ontology based, Word embedding based, Corpus based, Alignment based and Literal based features are used in SVR model respectively, in order to explore the effect of each kind of features. We used SemEval 2016 evaluation data set to test the performance of different feature set, and the results of Pearson Correlation coefficients are shown in Table 1.

The Table 1 indicates Word2Vec performed better in HDL, Postediting, Plagiarism data set, and WordNet performed better in Ans-Ans, Qus-Qus data set. The reason maybe that training Word2vec uses all the English corpus of Wikipedia, and it can learn better word vectors. WordNet can make full uses of lexical information to match the synonyms between two sentences.

We also used SemEval 2017 evaluation data to test our system, and adding each kind of feature one by one. The result of Pearson Correlation coefficients are shown in Table 3.

From Table 3, we can see Ontology based features, Corpus based features and Literal based features outperformed others in SemEval 2017 evaluation data set.

Feature	**Pearson correlation**
Alignment Based	0.7527
Ontology Based	0.7816
Word2vec Based	0.7823
GloVe Based	0.7836
Corpus Based	0.8104
Literal Based	0.8231
All	0.8231

Table 3: The Pearson Correlation on SemEval 2017 evaluation data sets.

4 Conclusion and Future Works

In this paper, we describe our system in the Semantic Textual Similarity task1 subtask 5 English monolingual similarity in SenEval 2017. We used 5 kinds of features and SVR model to build the ultimate system. We find that Ontology based feature, Word Embedding based feature and Alignment based feature performed better in some aspects of semantic similarity calculation. With the limitation of time, we do not try other methods. In our future work, we are going to attempt LSTM tree method to calculate sentences similarity.

Acknowledgment

This work is sponsored by the National High Technology Research and Development Program of China (2015AA015405) and National Natural Science Foundation of China (61572151 and 61602131).

References

Eneko Agirre, Daniel M. Cer, Mona T. Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In *Proceedings of the 6th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2012,* *Montréal, Canada, June 7-8, 2012*. pages 385–393. http://aclweb.org/anthology/S/S12/S12-1051.pdf.

Steven Bird. 2006. NLTK: the natural language toolkit. In *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*. http://aclweb.org/anthology/P06-4018.

David Jurgens and Keith Stevens. 2010. The s-space package: An open source package for word space models. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden, System Demonstrations*. pages 30–35. http://www.aclweb.org/anthology/P10-4006.

Claudia Leacock, Martin Chodorow, and George A. Miller. 1998. Using corpus statistics and wordnet relations for sense identification. *Computational Linguistics* 24(1):147–165.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27, 1998*. pages 296–304.

Rada Mihalcea, Courtney Corley, and Carlo Strapparava. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*. pages 775–780. http://www.aaai.org/Library/AAAI/2006/aaai06-123.php.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *CoRR* abs/1301.3781. http://arxiv.org/abs/1301.3781.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM* 38(11):39–41. https://doi.org/10.1145/219717.219748.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*. pages 1532–1543. http://aclweb.org/anthology/D/D14/D14-1162.pdf.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM Comput. Surv.* 34(1):1–47. https://doi.org/10.1145/505282.505283.

Dan Stefanescu, Rajendra Banjade, and Vasile Rus. 2014. Latent semantic analysis models on wikipedia and TASA. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014.*. pages 1417–1422. http://www.lrec-conf.org/proceedings/lrec2014/summaries/403.html.

Md. Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014a. Back to basics for monolingual alignment: Exploiting word similarity and contextual evidence. *TACL* 2:219–230. https://tacl2013.cs.columbia.edu/ojs/index.php/tacl/article/view/292.

Md. Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014b. Dls$@$cu: Sentence similarity from word alignment. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014.*. pages 241–246. http://aclweb.org/anthology/S/S14/S14-2039.pdf.

Stefanie Tellex, Boris Katz, Jimmy J. Lin, Aaron Fernandes, and Gregory Marton. 2003. Quantitative evaluation of passage retrieval algorithms for question answering. In *SIGIR 2003: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 28 - August 1, 2003, Toronto, Canada*. pages 41–47. https://doi.org/10.1145/860435.860445.

Joseph P. Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*. pages 384–394. http://www.aclweb.org/anthology/P10-1040.

Dominic Widdows and Kathleen Ferraro. 2008. Semantic vectors: a scalable open source package and online technology management application. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*. http://www.lrec-conf.org/proceedings/lrec2008/summaries/300.html.

Zhibiao Wu and Martha Stone Palmer. 1994. Verb semantics and lexical selection. In *32nd Annual Meeting of the Association for Computational Linguistics, 27-30 June 1994, New Mexico State University, Las Cruces, New Mexico, USA, Proceedings.*. pages 133–138. http://aclweb.org/anthology/P/P94/P94-1019.pdf.