BIT at SemEval-2016 Task 1: Sentence Similarity Based on Alignments and Vector with the Weight of Information Content

Hao Wu [1], **Heyan Huang***[1], **Wenpeng Lu** [2]
[1]Beijing Engineering Research Center of High Volume Language Information Processing
and Cloud Computing Applications, School of Computer Science,
Beijing Institute of Technology, Beijing, 100081, China
[2]QiLu University of Technology
wuhao123@bit.edu.cn, hhy63@bit.edu.cn, lwp@qlu.edu.cn

Abstract

This paper describes three unsupervised systems for determining the semantic similarity between two short texts or sentences submitted to the SemEval 2016 Task 1, all of which make use of only off-the-shelf software and data making them easy to replicate. Two systems achieved a similar Pearson correlation coefficient (0.64661 by simple vector, 0.65319 by word alignments). We include experiments on using our alignment based system on evaluation data from the 2014 and 2015 STS shared task. The results suggest that beyond the core similarity algorithm, other factors such as data preprocessing and use of domain-specific knowledge are also important to similarity prediction performance.

1 Introduction

Given two short texts or sentences, similarity systems or models should output a score that reflects how similar the two texts are in meaning. Semantic textual similarity (STS) formalizes an operation that is an important component of many natural language processing systems and has generated substantial interest within the research community (Agirre et al., 2012; Agirre et al., 2013; Agirre et al., 2014; Agirre et al., 2015). STS methods can be applied in example-based machine translation, machine translation evaluation, information retrieval, text summarization, question answering, and recommendation systems.

2 System Overview

In STS 2016, we submitted three system runs, and all of which were unsupervised. They could be generally divided into two kinds: vector based and alignment based.

2.1 Run 1: Simple Vector Method

In this run, we use a sentence vector derived from word embeddings obtained from word2vec (Mikolov et al., 2013). Using these sentence level vector representations, the similarity between two texts can be computed using the cosine operation.

We train word embeddings by running the word2vec toolkit1 over the fifth edition of the Gigaword corpus (LDC2011T07). We preprocess the Gigaword data with the following tools from the Moses machine translation toolkit (Koehn et al., 2007): the data is tokenized using tokenizer.perl; truecase.perl4 is used to standardize capitalizing.

As illustrated in Equation (1), we construct the sentence vector $\vec{s}$ by simply summing together the word embeddings, $\vec{t_i}$, associated with each token in a sentence.

$$\vec{s} = \sum_{i=1}^{|s|} \vec{t_i}\ . \tag{1}$$

Here $|s|$ is the number of tokens that the sentence contains.

The similarity between a pair of sentences is computed as the cosine of their associated sentence level embedding vectors.

*Corresponding author

Proceedings of SemEval-2016, pages 1270–1274,
San Diego, California, June 16-17, 2016. ©2016 Association for Computational Linguistics

2.2 Run 2: Weighted Vector Method

The above method weights all word embeddings equally. We submitted an alternative run that weights the word embeddings by the information content (IC) of the concepts referenced by their word sense tagged tokens (Resnik, 1995). Word sense disambiguation is performed using BabelNet (Navigli and Ponzetto, 2012) with the WordNet (Miller, 1995) sense inventory. NLTK (Bird, 2006) is used to obtain the frequencies of words belongs to the WordNet synset. The probability associated with each concept is estimated over the BNC[1] using add one smoothing. Following Resnik (1995), we then compute the information content of each concept as follows:

$$IC\,(c) = -\log P(c). \tag{2}$$

Here $P(c)$ refers to the statistical frequency of concept c.

This method allows us to compute IC based weights only for the nouns and verbs covered by WordNet. We heuristically set the weight of adjectives and adverbs to 5 and other words to 2.

2.3 Run 3: Word Alignment Method

Our final run differs from the vector based methods described above and follows a popular alternative approach to assessing sentence similarity through word alignments. We make use of Sultan et al. (2014a)'s open-source monolingual word aligner with default parameters and the similarity formula proposed in Sultan et al. (2015). An unsupervised system based on Sultan et al. (2015)'s similarity formula above took fifth place at STS 2015. Its predecessor, based on a similar formula, took 1st place at STS 2014. As shown in Equation (3), similarity is computed as

$$sts\left(S^{(1)}, S^{(2)}\right) = \frac{n_c^a\left(S^{(1)}\right) + n_c^a\left(S^{(2)}\right)}{n_c\left(S^{(1)}\right) + n_c\left(S^{(2)}\right)}. \tag{3}$$

Here $n_c^a\left(S^{(i)}\right)$ and $n_c\left(S^{(i)}\right)$ are the number of content words and the number of aligned content words in sentence $S^{(i)}$, respectively.

No.	Dataset	Total Pairs	Pairs with GS
1	answer-answer	1572	254
2	headlines	1498	249
3	plagiarism	1271	230
4	postediting	3287	244
5	question-question	1555	209
	Total	9183	1186

Table 1: Test sets at SemEval STS 2016.

3 Data

As shown in Table 1, the 2016 STS shared task included 5 distinct datasets. Systems were required to annotated between 1,498 and 3,287 pairs per dataset. System performance was evaluated on a subset of each dataset consisting of between 209 to 255 gold standard (GS) pairs.

The GS similarity scores for each pair range from 0 to 5, with the values having the corresponding interpretations:

5 indicates completely equivalence; 4 expresses mostly equivalent with differences only in some unimportant details; 3 means roughly equivalent but with differences in some important details; 2 means non-equivalence but sharing some details; 1 means the pairs only share the same topic; and 0 represents no overlap in similarity.

We note that there is a big gap between 0 and 1 in GS metric: Intuitively, within the range [1,5], scores linearly represent the similarity between two texts. However, there is a much larger conceptual range of topical similarity that spans from pairs on the exact same topic to those that are completely dissimilar.

4 Evaluation

The evaluation metric is the Pearson correlation coefficient (PCC) (Brownlee, 1965) between system output and the gold standard. PCC is used for each individual test set, and the final evaluation is measured by weighted mean of PCC on all datasets (Agirre et al., 2012).

4.1 STS 2016 Results

Performances of our three systems on each of STS 2016 test sets are showed in Table 2, and the last two columns show the results of the following modified versions of Run 2 and Run 3.

Run 2': Word embedding vectors are normalized to have length=1, and the heuristic IC weights are

[1] http://ota.ox.ac.uk/desc/2554

No.	Run 1	Run 2	Run 3	Run 2’	Run 3’
1	.48863	.37565	**.54530**	.52210	**.64349**
2	.62804	.55925	**.78140**	.69421	**.80295**
3	.80106	.75594	**.80473**	.78410	**.81391**
4	**.79544**	.77835	.79456	.79666	**.79863**
5	**.51702**	.51643	.29972	**.58535**	.57826
Mean	.64661	.59560	**.65319**	.67668	**.73044**

Table 2: Performance on STS 2016. The last row shows weighted mean which is the final evaluation metric, and the last two columns describe modified versions of Run 2 and Run 3.

adjusted as follows: 6 for adjectives and adverbs and 3 for other words.

Run 3’: If there is no content word aligned, we make use of longest common substring algorithm to obtain the longest common consecutive words (LCCW) of the compared sentences. Similarity is computed as

$$sts\left(S^{(1)}, S^{(2)}\right) = \frac{2 \times \left|LCCW\left(S^{(1)}, S^{(2)}\right)\right|}{|S^{(1)}| + |S^{(2)}|}. \tag{4}$$

Here $\left|LCCW\left(S^{(1)}, S^{(2)}\right)\right|$ is the number of words that are present in the LCCW of $S^{(1)}$ and $S^{(2)}$.

Words are classified as content words if they are either nouns, verbs, adjectives or adverbs with a small number of exceptions. We elected to classify *think, know, want* and *act* as non-content words based on their IDF scores.

From Table 2, we make the following observations:

1. From our submitted systems, we obtain the best overall results from Sultan et al. (2015)’s word alignment based method (0.65319). However, the simple vector method (0.64661) is very close in performance with only a 0.00658 absolute difference in the overall correlation scores.

2. Weighting the raw word embeddings by their IC degraded performance on all of the datasets. Run 2’ normalized the word embedding vectors before taking the IC weighted vector sum, resulting in significantly improved performance over both the submitted Run 2 as well as over Run 1’s simple summation of the embedding vectors. This shows any reweighting of the word level embedding vectors needs to account for differences in the magnitude of the raw embeddings.

Dataset 2015	Run 3	DLS15u	Best15	Run 3’
answers-forums	.6404	.6821	.7390	.6675
answers-students	.7543	.7879	.7879	.7590
belief	.6724	.7325	.7717	.7189
headlines	.7671	.8238	.8417	.8009
images	.7927	.8485	.8713	.8496
Weighted Mean	.7426	.7919	.8015	.7757

Dataset 2014	Run 3	DLS14-2	Best14	Run 3’
deft-forum	.452	.483	.531	.484
deft-news	.608	.766	.781	.772
headlines	.734	.765	.784	.753
images	.794	.821	.834	.830
OnWN	.705	.859	.875	.824
tweet-news	.694	.764	.792	.723
Weighted Mean	.688	.761	.761	.746

Table 3: Table 3: Run 3 and Run 3 performance on STS 2014 and 2015. For each year, the third column shows the performance of the submitted unsupervised system with the best overall performance for that year. The forth column shows the best per dataset performance across submitted unsupervised systems.

3. The best performance of all of our systems is achieved by Run 3’, which included additional logic to handle pairs with no aligned content words. However, both Run 3 and Run 3 performed particularly badly on the question-question dataset. Inspecting the data reveals that some sentence pairs have a GS score of 0 even when there is some level of similarity between what is being asked, such as ”What’s the best way to store asparagus?” vs ”What’s the best way to store unused sushi rice?”. We also observe that many pairs in this dataset set have similarly structured sentences with particular core words playing a decisive role.

4.2 Results on Past Test Sets

In order to better frame the performance of our systems, we examined the performance of Run 3 and Run 3’, our word alignment base systems, on the STS shared task evaluation sets from 2014 to 2015. Recall that our method is unsupervised and most comparable to Sultan et al. (2015)’s unsupervised system submission. We contrast our results with both Sultan et al. (2014b)’s best shared task submission and to Sultan et al. (2015)'s supervised extension to the unsupervised system. The results are shown in Table 3.

At the SemEval 2014 STS task, unsupervised DLS@CU2014-run 2 (Sultan et al., 2014b) achieved the highest final PCC score across all 38 submitted systems runs. Sultan et al. (2015) submitted a supervised system that contained only two features: (1) the similarity score from the unsupervised Sultan et al. (2015) system; (2) a complementary feature based on the cosine of the sentence level vector representations obtained by averaging word-level embedding vectors. This system took first place in the 2015 shared task, with the unsupervised Sultan et al. (2015) system coming in 5th place. Our Run 3 and Run 3' systems are identical to Sultan et al. (2015)'s unsupervised system except for differences in text preprocessing. We observe that our performance may have been diminished by not performing the following preparation steps:

1. Our systems didn't use a spelling correction module, such as a levenshtein distance of 1 between a misspelt word and a correctly spelt word before running the aligner or finding word vectors.

2. Knowledge of domain-specific stop words wasnot taken into account in submitted systems.

We suspect these contributed to the performance gap between our system and even the very similar Sultan et al. (2014b) submission.

5 Conclusions and Future Work

At SemEval 2016, we submittted three unsupervised STS systems: simple vector method, weighted vector method and word alignment method. Two make use of sentence level embedding vectors and the other applies a known well performing method for calculating STS similarity scores that is based on monolingual word alignments. We observe that both types of systems are able to achieve a similar PCC. Based on observations obtained by running our system on evaluation sets from earlier years, we believe our system could have been improved by including more of the text preprocessing steps performed in prior work.

First, our systems should introduce a spelling correction module to deal with misspelt words, which is a good way to increase the recall of the input. Second, domain-specific knowledge should be taken into account, such as domain-specific stop words, which can adapt to requirements posed by different data domains and applications. In future work, we hope to investigate the use of domain-specific weights for words as well as other methods for term weighting such as TF-IDF.

Acknowledgments

The work described in this paper is mainly supported by State Key Program of National Natural Science Foundation of China under Grant 61132009, National Programs for Fundamental Research and Development of China (973 Program) under Grant 2013CB329303 and National Natural Science Foundation of China under Grant 61202244.

The authors would like to thank Daniel Cer for his insightful comments to the improvement in technical contents and paper presentation.

References

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: A pilot on semantic textual similarity. In *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*, pages 385–393. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. SemEval-2013 shared task: Semantic textual similarity, including a pilot on typed-similarity. In *In* SEM 2013: The Second Joint Conference on Lexical and Computational Semantics. Association for Computational Linguistics*. Citeseer.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. SemEval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 81–91.

Eneko Agirre, Carmen Banea, et al. 2015. SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability. In *Proceedings* *of the 9th International Workshop on Semantic Evaluation (SemEval 2015), June*.

Steven Bird. 2006. NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

Kenneth Alexander Brownlee. 1965. *Statistical theory and methodology in science and engineering*, volume 150. Wiley New York.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, pages 177–180. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

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

Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. *International Joint Conference on Artificial Intelligence(IJCAI)*.

Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014a. Back to basics for monolingual alignment: Exploiting word similarity and contextual evidence. *Transactions of the Association for Computational Linguistics*, 2:219–230.

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 2014)*, pages 241–246.

Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2015. DLS@CU: sentence similarity from word alignment and semantic vector composition. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, pages 148–153.