UQeResearch: Semantic Textual Similarity Quantification

Hamed Hassanzadeh[1], Tudor Groza[1,2], Anthony Nguyen[3], Jane Hunter[1]
[1]School of ITEE, The University of Queensland, St Lucia, QLD, Australia
[2]Garvan Institute of Medical Research, Darlinghurst, NSW, Australia
[3]Australian e-Health Research Centre, CSIRO, Brisbane, QLD, Australia
h.hassanzadeh@uq.edu.au, t.groza@garvan.org.au,
anthony.nguyen@csiro.au, jane@itee.uq.edu.au

Abstract

This paper presents an approach for estimating the Semantic Textual Similarity of full English sentences as specified in Shared Task 2 of SemEval-2015. The semantic similarity of sentence pairs is quantified from three perspectives - structural, syntactical, and semantic. The numerical representations of the derived similarity measures are then applied to train a regression ensemble. Although none of these three sets of measures is able to represent the semantic similarity of two sentences individually, our experimental results show that the combination of these features can precisely assess the semantic similarity of the sentences. In the English subtask our system's best result ranked 35 among 73 system runs with 0.7189 average Pearson correlation over five test sets. This was 0.08 correlation points less than the best submitted run.

1 Introduction

Semantic textual similarity (STS) aims to automatically estimate the relatedness of the meaning of sentences (Agirre et al., 2015). The literature consists of a series of well-established frameworks to explore a deeper understanding of the semantic relationship between entities, ranging from ontological reasoning to compositional as well as distributional semantics (Cohen et al., 2009). However, automatically estimating the semantic similarity of full sentences is still a challenging task.

Our system aims to quantify the similarity of pairs of sentences by encoding a variety of relatedness features in a vector of attributes and then predicting their similarity scores by employing machine-learning algorithms. Different syntactic, semantic, and structural similarity measures have been applied to quantify the similarity of texts. We have chosen to approach the estimation of similarity as a regression problem. Hence, we use the quantified similarity of sentence pairs to train a regressor that can then be applied to predict similarity scores for the unseen pairs. The paper is structured as follows: Section 2 presents the proposed similarity measures. In Section 3, the regression models are introduced and the experimental results are discussed in detail. The conclusions are summarized in Section 4.

2 Similarity Measures

In this section we describe the similarity measures we have employed to calculate semantic relatedness of pairs of sentences.

2.1 Syntactic Similarity Measures

Bags of words overlap: A simple measure for computing the similarity of a sentence pair is the number of words they have in common. Although a pair of sentences with the same bag of words (i.e. unordered list of all words of a sentence) can convey completely different meanings, this measure along with some structural measures can form an effective criterion for semantic comparison.

Bags of lemmatised/stemmed words overlap: The value of this feature is computed using the same method as above, however, instead of using bags of words, it uses bags of lemmas / stems.

Set similarity of lemmatised effective words: There are a number of words in a sentence that do not play effective roles in modelling the meaning of that sentence, such as determiners (*the, a, an*) and preposition or subordinating conjunctions (*in, on, at*). We remove these terms from the bag of words of a sentence and we call the remaining

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 123–127,
Denver, Colorado, June 4-5, 2015. ©2015 Association for Computational Linguistics

words the set of effective words. In this measure we lemmatise the effective words and compare the resulting sets of lemmas for a pair of sentences.

Jaccard similarity of sets of words/lemmas: A sentence can be considered as a set of words. To incorporate this perspective, we calculate the Jaccard similarity coefficient of a pair of sentences.

Windows of words overlap: We perform a sliding window of different sizes (from window of two words up to the size of the smaller sentence in a pair) over a pair of sentences. Afterwards we compute the total number of equal windows of words of two sentences. Also, we keep the size of the longest equal window of words that two sentences share together. Due to varying sizes of sentences and therefore varying sizes and number of windows, we normalise each of these measures to reach a comparable value between zero and one. The same window-based measures can be alternatively be calculated by only considering effective words in sentences and also, from a grammatical perspective, by only considering Part of Speech (POS) tags of the constituent words of sentences.

Ratio of shared skipped bigrams: Skipped bigrams are the pairs of words which are created by combining two words of a sentence that are located in arbitrary positions. The set of these bigrams can then be used as a basis for similarity comparison. We create the skipped bigrams of participating verbs, nouns, adjectives, and adverbs of a sentence (we ignore other unimportant terms) and then calculate the intersection of each set of these bigrams with the corresponding set from the other sentence in a pair.

Pairwise Sentence Polarity: We investigate the presence of some lexical elements that act as negation agent, e.g., *not, neither, no*, etc. We apply the NegEx algorithm (Chapmana et al., 2001) to find the negation in sentences and then we perform pairwise comparison of the polarity of sentences.

Ratio of Sentence Lengths: The relative length of two sentences (length of smaller sentence over the longer one) provides a simple measure of similarity. However, this naïve attribute of a pair can be useful when combined with other more conceptual measures.

2.2 Structural Similarity Measures

Ratio of number of clauses: The meaning of a sentence can be inferred from the meaning of its clause(s). Consequently, the equality of the clauses of a pair of sentences provides another measure for assessing the relatedness of those sentences. In this case, the level of equality is calculated by analysing the parse tree of each sentence and finding the number of clauses that each sentence is composed of. The ratio of this clause-level equality is then obtained by dividing the smaller number of clauses by the larger number of clauses for each pair. Parse trees were produced with the Stanford Parser (Klein et al., 2003).

Reduced parse tree overlap: While the previous measure only considered the shallow size-based comparison, this measure provides a more in-depth analysis of the structural similarity. More concretely, it quantifies the overlap of the parsed trees for each sentence, composed of only the POS tags of the effective words.

2.3 Semantic Similarity Measures

Role-based word-by-word similarity: In order to compute this measure, we first split the sentences into clauses and determine the subject, predicate and object within each clause. Each of these roles is then transformed into a bag of lemmatised words, which is then compared to corresponding bags of lemmatised words denoting the same role in the other sentence. The similarity between the two bags of words is calculated using a mixture of two well-known semantic similarity measures – i.e., Lin (1998) and Wu & Palmer (1994), both having WordNet (Miller, 1995) as background knowledge. Due to WordNet's lower coverage of verbs, for the words in the predicate bags we compute the similarity between words using FrameNet (Fillmore et al., 2003) and by comparing sets of corresponding frames of words in each bag.

Semantic similarity of effective words: Given the sets of effective words of a pair of sentences, we compute their similarity using the same method as above, however, without taking into account the underlying roles – i.e., it is computed in a sentence-wide manner.

Cosine similarity of Information Content (IC) vectors: We map the sequence of words in a sentence to a vector of corresponding numeric values. In order to create this vector we use the notion of Information Content (IC) (Resnik, 1995). The relatedness of a given pair can then be estimated by employing a distance measure between the two vectors, such as the cosine similarity.

Role-based POS tags alignment: For this similarity measure we get the POS tags of each word in the subject and object phrases of a sentence and form a sequence of these tags. We then employ Needleman-Wunsch algorithm (Needleman et al., 1970) for aligning these sequences of POS tags to find their similarity ratio.

WordNet/FrameNet based synonym similarity: Other sets of vocabulary-based similarity measures can be devised by getting all the synonyms of each word of sentences and considering them in the comparison process. One of these measures can be calculated by applying WordNet for obtaining synonyms of words. For this WordNet synonymy measure, the corresponding synsets of all the lemmas of the effective words in sentences are retrieved from WordNet. The sets of synsets of a pair of sentences are then compared to each other and the ratio of their similarity is calculated. Another similar measure can be calculated using FrameNet as the background knowledge instead of WordNet.

Cosine similarity of the best senses: This measure uses a WordNet-based word sense disambiguation approach to find the best senses of effective words of a pair. These senses are then used to form vectors of best senses, which can then be compared using cosine similarity.

Normalised set similarity for best senses synsets: Similar to the previous measure, we apply word sense disambiguation to retrieve the best senses for all words of the sentence, and subsequently create a set of synsets which can be compared to the corresponding set of synsets extracted from the other sentence.

Normalised set similarity of the best senses skipped bigrams: We create a set of skipped bigrams of best senses of words instead of the skipped bigrams of words of a sentence and then calculate each pair's sets similarity.

Similarity of sets of associated terms: Our last two sets of features make use of vector space models, using Wikipedia English articles as the background corpus and Hyperspace Analogue to Language (HAL) model to produce term vectors (Lund et al., 1996) by employing the SemanticVector library (Widdows et al., 2008). The associated terms for words of a sentence form a set that can be compared with a corresponding set of another sentence – for example, by calculating their intersection. The resulting value is normalised by size of the smallest set.

Cosine similarity of matrices of associated terms vectors: For this last feature, we use the numerical representation (vector) of each term, retrieved from the distributional model, to form a matrix of associated terms vectors for a sentence. To enhance the effectiveness of this similarity measure, only vectors of effective words of a sentence are used to build the matrix.

3 Results

In this section, the results from applying our system to STS 2015 (Task 2) are presented. Before discussing the results, we firstly describe the experimental setup and training process.

3.1 Experimental Setup

All the data released in STS 2012, 2013, and 2014 was permitted to be used to develop and train the systems. All the data sets consist of pairs of sentences along with their human annotated similarity scores. The similarity scores ranged from 0 to 5, with 0 representing completely dissimilar pairs and 5 representing perfect similarity (or equality). In order to evaluate the English STS systems, five test sets were provided. Although the test data in total consists of 8500 pairs, a subset of the instances of each test set was sampled and used for the final official evaluations by the organizers. The official measurement criterion for evaluation is the Pearson correlation. It should be mentioned that prior to computing the measures the punctuations were removed from sentences to avoid naïve token-level matching of them in some similarity measures.

3.2 Experiments Over Training Data

We first performed a number of experiments over the training data in order to prepare the final regression system. The training set consists of 10592 annotated pairs, achieved by merging previous SemEval STS data sets. We approached the semantic similarity estimation as a regression problem. Hence, we investigated different regression algorithms and Table 1 lists their evaluation results. The WEKA implementations of these algorithms have been used in our system (Hall et al., 2009).

Algorithm	**Pearson Correlation**	**Root mean squared error**
Regression Algorithms		
RepTree	0.6747	1.1207
K*	0.6968	1.1497
Linear Regression	0.6809	1.1088
Regression By Classification		
Regression by Random Forest	0.7745	0.964
Regression by KNN	0.7139	1.0651
Regression Ensemble		
Ensemble	**0.7813**	**0.9484**

Table 1: Experiments on training data (5-fold cross validation).

The first part of Table 1 shows the results achieved by selected regression approaches. Among these algorithms, K* achieved the best Pearson correlation. In regression by classification, the continuous similarity scores are discretised to nominal values. Then, a classifier was used to categorize instances into the resultant nominal classes. In our experiments, the continuous range of 0 to 5 scores is discretised into 10 bins. The best results have been achieved by applying Random Forest as the base classifier. Finally, the ensemble of regressors is composed of three meta-regressors: bagging, random SubSpace, and regression by discretisation. Regression by discretisation follows precisely the same methodology as above. The bagging strategy uses RepTree as its first level regressor, while the random SubSpace employs the K* algorithm. The final outputs of the ensemble are the average of the prediction values from all of the regressors. This ensemble gained the best correlation amongst all of the models.

3.3 Results Over Test Data and Discussions

We submitted three different runs to the English STS 2015 Task 2. The same regression ensemble has been applied to all three runs. The main difference between them is related to the data that was used for training. The data used to train the *run1* system were STS 2012 train and test sets, STS 2013 test set, and STS 2014 test set. In the second system (*run2)*, we used all the run1 data as well as one additional data set which was the training set of the SICK corpus (Marelli et al., 2014). It was introduced in SemEval-2014 Task 1. Contrary to STS corpora, the similarity scores from the SICK corpus ranged from 1 to 5 (instead of 0 to 5). We gave a unique numerical ID to each pair in the data sets, which were then kept in the feature vectors as well. In *run3,* exactly the same data was used as *run1* but without the IDs in the feature vectors.

	run1	**run2**	**run3**
answers-forums	0.5923	0.6132	**0.6188**
answers-students	0.6876	**0.6882**	0.6757
belief	0.5904	0.6229	**0.7178**
headlines	0.7521	**0.7602**	0.7549
images	0.7817	**0.7855**	0.7769
Means	0.7032	0.7130	**0.7189**
Rank	40	37	**35**

Table 2: Our systems' results over test sets.

Table 2 lists the results of our system runs. It can be observed that the third run achieved better overall correlations compared with the other two. By applying the additional data set (i.e. training set of the SICK corpus) the average correlation slightly improved (i.e. in run2). However, as previously mentioned, the difference in scoring the semantic similarities (0-5 vs. 1-5) caused the regressor model to fail to encode the scores properly (especially for lower similarity scores). In addition, as a side experiment, but contrary to the positive experience gained from SemEval-2014 semantic relatedness Task, the unique numerical ID had a negative impact over the outcome of the system (comparing run1's results – with IDs, to run3's – without IDs).

4 Conclusions

This paper describes the system we submitted to SemEval-2015 Task 2: STS in order to estimate semantic similarity of full English sentences. We approached the task as a regression problem. An ensemble of regressors as well as a variety of similarity measures was proposed. These measures (that compared syntactic, semantic, and structural aspects) were extracted from pairs of sentences. Our system's best result ranked 35 among 73 submitted runs with 0.7189 average Pearson correlations over five test sets. This was 0.08 correlation points less than the best submitted run.

Acknowledgments

This research is funded by the Australian Research Council (ARC) Discovery Early Career Researcher Award (DECRA) -- DE120100508. It is also partially supported by CSIRO Postgraduate Studentship.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 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)*, Denver, CO.

Wendy W. Chapmana, Will Bridewellb, Paul Hanburya, Gregory F. Coopera, and Bruce G. Buchanan. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of Biomedical Informatics, 34*(5), 301-310.

Trevor Cohen and Dominic Widdows. 2009. Empirical distributional semantics: Methods and biomedical applications. *Journal of Biomedical Informatics, 42*(2), 390-405.

Charles J. Fillmore, Christopher R. Johnson, and Miriam R.L. Petruck. 2003. Background to FrameNet. *International Journal of Lexicography, 16*(3), 235-250.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explorations, 11*(1).

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the Conference 41st Annual Meeting of the Association for Computational Linguistics*, pages 423-430.

Dekang Lin. 1998. An Information-Theoretic Definition of Similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296-304.

Kevin Lund and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods Instruments & Computers, 28*(2), 203-208.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Re-sources and Evaluation (LREC-2014)*, Reykjavik, Iceland.

George A. Miller. 1995. Wordnet - a Lexical Database for English. *Communications of the ACM, 38*(11), 39-41.

Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins *Journal of Molecular Biology, 48*(3), 443 - 453.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 448-453.

Dominic Widdows and Kathleen Ferraro. 2008. Semantic Vectors: A Scalable Open Source Package and Online Technology Management Application. In *Sixth International Conference on Language Resources and Evaluation, Lrec 2008*, pages 1183-1190.

Zhibiao Wu and Martha Palmer. 1994. Verbs Semantics and Lexical Selection. In *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, pages 133-138, Las Cruces, New Mexico.