OPI-JSA at SemEval-2017 Task 1: Application of Ensemble learning for computing semantic textual similarity

Martyna Śpiewak, Piotr Sobecki and Daniel Karaś
National Information Processing Institute
al. Niepodległości 188b, 00-608 Warsaw, Poland
{mspiewak, psobecki, dkaras}@opi.org.pl

Abstract

Semantic Textual Similarity (STS) evaluation assesses the degree to which two parts of texts are similar, based on their semantic evaluation. In this paper, we describe three models submitted to STS SemEval 2017. Given two English parts of a text, each of proposed methods outputs the assessment of their semantic similarity.

We propose an approach for computing monolingual semantic textual similarity based on an ensemble of three distinct methods. Our model consists of recursive neural network (RNN) text auto-encoders ensemble with supervised a model of vectorized sentences using reduced part of speech (PoS) weighted word embeddings as well as unsupervised a method based on word coverage (TakeLab). Additionally, we enrich our model with additional features that allow disambiguation of ensemble methods based on their efficiency. We have used Multi-Layer Perceptron as an ensemble classifier basing on estimations of trained Gradient Boosting Regressors.

Results of our research proves that using such ensemble leads to a higher accuracy due to a fact that each member-algorithm tends to specialize in particular type of sentences. Simple model based on PoS weighted Word2Vec word embeddings seem to improve performance of more complex RNN based auto-encoders in the ensemble. In the monolingual English-English STS subtask our Ensemble based model achieved mean Pearson correlation of .785 compared with human annotators.

1 Introduction

The objective of a system for evaluating semantic textual similarity, is to produce a value which serves as a rating of semantic similarity between pair of text samples. Such task certainly could not be regarded as toy problem, the results could be used to solve multiple real-world problems, e.g. plagiarism detection. We used described methods in STS task in the SemEval 2017 competition (Bethard et al., 2017).

2 Methods

2.1 Data

For the purpose of this research we have used datasets provided by the SemEval challenge organizers containing English sentence pairs coming from several sources. STS Task objective is to produce a value in the range between 0.0 and 5.0, which assessing semantic similarity of a given pair of sentences. Intermediate levels are corresponding to partial similarity such as rough or topical equivalence but with differing details. In this study, we have used all English datasets provided by the challenge organizers until this year to train our supervised models.

2.2 Models

The core of the system is based on widely used Gradient Boosting algorithm. The main novelty of described system lies in the formulation of its feature vectors.

Each feature vector can be divided into two main parts: similarity scores and sentences' descriptors. The process of feature extraction compiles similarity scores of three distinct methods (described later in detail) — effectively forming an ensemble. Additionally, for every pair of sentences, following descriptors are also attached to feature vector:

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 139–143,
Vancouver, Canada, August 3 - 4, 2017.

lengths of the evaluated sentences, Word2Vec coverage as well as two boolean predicates — one of them indicates if a sentence is a question and another one indicating if sentence contains numbers. Word2Vec coverage is defined as follows:

$$G_{lv}(S_i) = \frac{|S_i \cap G|}{|S_i|}$$

where S_i denotes set of all words present in ith sentence and G is a set of all words available in Word2Vec.

The logic behind introduction of these descriptors is based on observations made during evaluation of each separate method. Overall they all achieved a similar Pearson score, but accuracy of every method in context of particular instances of sentence pairs was different. For example, model based on cosine similarity of Word2Vec vectors performed worse in case of long sentences and when the sentences contained words not present in Word2Vec. Ideally introduction of sentences' descriptors to feature vectors would let the regressor "pick" the right method for each case by learning the correlations between features exhibited by sentences and performance of particular method. This hypothesis has been proven true, which is further backed by achieved results.

We used the implementation of Gradient Boosting and Multi-layer Perceptron (MLP) from scikit-learn library (Pedregosa et al., 2011). Facilities present in mentioned library were also used for evaluation using 3-fold crossvalidation and hyper-parameters optimization using grid search method. We have used low number of folds in Cross Validation to prevent over-fitting.

2.2.1 TakeLab

This method contributes three components for feature vector used by the meta-regressor. These components correspond to three word similarity measures defined by (Šarić et al., 2012) — ngram overlap, weighted word overlap and WordNet-augmented word overlap. Authors of (Šarić et al., 2012) use Google Books Ngrams for computing information content used in the weighted word overlap measure — we, in comparison, use the frequency list from British National Corpus (Leech, 2016).

Mentioned overlaps were implemented in Java programming language. The WS4J library was used for computing the WordNet path lengths between words with Wu-Palmer method. The OpenNLP library was used for both lemmatization and PoS-tagging. For complete overview of TakeLab measures see (Šarić et al., 2012).

2.2.2 Run 1: Part of Speech weighted Word2Vec Similarity (PoS-Word2Vec)

Described model is based on a well-documented Word2Vec (Mikolov et al., 2013) method of textual information encoding that allows vectorized representation of words, enforces vector space proximity for semantically similar words.

Given sentence pairs (x, y) of words length (n_i, n_j), part of speech (PoS) weights of words w_{x_n} and w_{y_n} and vector representation of words v_{x_n} and v_{y_n} coming from given sentences x and y, respectively.To evaluate vector similarity we have used cosine similarity between vectors x and y:

$$cos(x, y) = \frac{x \cdot y}{||x|| \cdot ||y||}$$

We have extracted following features for each sentence pair, to produce resulting vector r:

- cosine similarity of the mean of word vectors in each sentence

$$r(0) = cos\left(\frac{\sum_{k=1}^{n_i} v_{x_k}}{n_i}, \frac{\sum_{k=1}^{n_j} v_{y_k}}{n_j}\right)$$

- cosine similarity of the mean of word vectors in each sentence weighted by the PoS of the word

$$r(1) = cos\left(\frac{\sum_{k=1}^{n_i} w_{x_k} \cdot v_{x_k}}{\sum_{k=1}^{n_i} w_{x_k}}, \frac{\sum_{k=1}^{n_j} w_{y_k} \cdot v_{y_k}}{\sum_{k=1}^{n_j} w_{y_k}}\right)$$

Furthermore, we have analyzed cross sentence word-wise cosine similarity:

$$M(i, j) = cos(v_{x_i}, v_{y_j}),$$

and obtained maximum, PoS weighted, cross sentence word similarity vector v:

$$v(k) = \max_{j=1,\ldots,n_j} M(k, j) \cdot w_x,$$

for $k = 1, \ldots, n_i$, and

$$v(k) = \max_{i=1,\ldots,n_i} M(i, k - n_i) \cdot w_y,$$

for $k = n_i, \ldots, n_i + n_j$.

We have extracted following statistical features from the resulting vector v and added to the resulting vector r:, Mean , Kurtosis, Skewness, Standard deviation, Maximum value, Minimum value, Percentiles (5th, 25th, 75th and 95th).

$$r(3) = \text{mean}(v)$$
$$r(4) = \text{kurtosis}(v)$$
$$r(5) = \text{skewness}(v)$$
$$r(6) = \text{sd}(v)$$
$$r(7) = \max(v)$$
$$r(8) = \min(v)$$
$$r(9) = \text{percentile}(v, 5)$$
$$r(10) = \text{percentile}(v, 25)$$
$$r(11) = \text{percentile}(v, 75)$$
$$r(12) = \text{percentile}(v, 95)$$

We have used precomputed Word2Vec vectors from GloVe dataset (300 dimensions) (Pennington et al., 2014) for words in sentence pairs and British National Corpus dataset (Leech, 2016) to obtain information about PoS of given word. PoS weights have been experimentally assigned using results from random walk evaluated using Spearman correlation. Statistical moments and percentiles have been experimentally selected during manual trial and error optimization. We trained Gradient Boosting Regressor on the extracted features and evaluated it using 3 fold cross validation to prevent over-fitting.

2.2.3 Run 2: Skip Thoughts Vectors

Skip-thought vectors is an encoder-decoder model (Kiros et al., 2015), which is based on an RNN encoder with GRU acivations and an RNN decoder with a conditional GRU. Instead, in our approach, we only used skip-thought vectors' encoder pretrained on the BookCorpus dataset (Zhu et al., 2015), which maps words to a sentence vector. We determined skip-thought vectors as generic features for all sentences.

Next, we computed component-wise features for given pair of sentences. Denoting a and b as two skip-thought vectors, we computed their component-wise features: product $a \cdot b$, absolute difference $|a - b|$, and the other statistics between sentence pairs used by (Socher et al., 2011). For two compared sentenced the used statistics are as follows:

- 1 if sentences contain exactly the same numbers or no numbers and 0 otherwise,
- 1 if both sentences contain the same numbers,
- 1 if the set of numbers in one sentence is a strict subset of the numbers in the second sentence,
- the percentage of words in one sentence which are in the second sentence and vice-versa,
- the mean of the ratios the number of words in one sentence by the numbers of words in the other sentence.

Finally, we concatenated all aforementioned features together as a final features vector. Again Gradient Boosting Regressor was trained on the obtained features.

2.2.4 Run 3: Ensemble

Using all English pair of sentences from previous years of this task with the available gold scores we computed TakeLab score and trained Gradient Boosting algorithm on PoS weighted Word2Vec features (Run 1) and skip thoughts vectors (Run 2). We used GridSearchCV function with 3 fold cross validation from scikit-learn library to determine the best parameters of Gradient Boosting algorithm according to Pearson measure, separately for each run. Next, we obtained three values as features of Multi-layer Perceptron to determined the final predicted gold scores for each pair of sentences.

3 Results

The purpose of the STS task is to assess the semantic similarity of two sentences. Sentences are scored using the continuous interval $[0, 5]$, where 0 denotes a complete dissimilarity and 5 implies a complete semantic equivalence between the sentences. The final result is the Pearson score between the fixed gold scores and the predicted values from the user system (Agirre et al., 2016).

Table 1: The official results on the test dataset for Subtask 5 (english-english).

Method	Pearson score
Run 3: Ensemble	0.7850
Run 2: Skip Thoughts Vectors	0.7342
STS Baseline	0.7278
Run 1: PoS-Word2Vec	0.6796

As mentioned above, our intention was to create a system to measure the level of paraphrasing, which may be applied to Polish pair of sentences in a relatively easy way in the future. It is worth

noticing that the Run 1 and the Run 2 strongly depend on particular language tools, e.g. Word2Vec or a corpus using to train Skip Thoughts Vectors. Furthermore, we did not have appropriate datasets to train these tools for other languages, so we decided to only take part in the Subtask 5 for English pair of sentences. In Table 1 we present the official results only for this subtask.

As was expected the best score was obtained for the ensemble approach. Due to the fact that used pair of sentences had a different format, the final regressor chose which method is better for a particular type of sentence (see Table 2).

Analysis of PoS-Word2Vec method clearly shows that overestimation occurs when subject in compared sentences differs. However cases of underestimation display lack of representation of idioms and use of informal speech. Overall the method seems to be too focused on the meaning of particular words. On the other hand, TakeLab exhibits poor performance in case of nearly-duplicate pairs of sentences. This doesn't come as much of surprise due to the way all TakeLab measures estimate similarity between sentences. This in turn translates to overestimation in cases when two sentences have high word coverage, but effectively differ in semantic meaning (see first example in Table 2). Skip thoughts vectors approach has the biggest problem with significant differences between the length of compared sentences, then there are also over and underestimation error. Also, this method does not handle near-duplicated sentences that sentences differ in only one or two words, and the different words are not synonyms.

4 Conclusion

In this paper, we have presented the OPI-JSA system submitted by our team for SemEval 2017, Task 1, Subtask 5. The proposed system uses a lot of different tools to encode a sentence to a features vector. We used machine learning algorithms to predict the gold score for given pairs of sentences which measure their similarity. Additionally, we showed that an ensemble method improved the performance of our system. The best results we have obtained is equal to 0.785 according to a Pearson's correlation while placing OPI - JSA as 36 of all reported solutions (77) and 16 of 32 teams in the Subtask 5.

Table 2: Examples of maximum over and underestimation of STS evaluation for proposed methods and sentence pairs. Error corresponds to difference between assessed STS and gold scores.

TakeLab	**Overestimation**	**Error**
What kind of socket is this?	What kind of bug is this?	4,54
The act of annoying someone or something	The act of liberating someone or something.	4,36
What is the difference between shawarma and gyros?	What is the difference between portamento and glissando?	4,26

TakeLab	**Underestimation**	**Error**
The lady peeled the potatoe.	A woman is peeling a potato.	-4,05
Utter fucking nonsense.	That doesn't make any sense.	-3,96
Eurozone backs Greek bailout	Eurozone agrees Greece bail-out	-3,87

PoS-Word2Vec	**Overestimation**	**Error**
The activity of examining or assessing something	The activity of protecting someone or something.	3,88
What is the significance of the cat?	What is the significance of the artwork?	3,72
Live Blog: Ukraine In Crisis	Live Blog: Iraq In Turmoil	3,71

PoS-Word2Vec	**Underestimation**	**Error**
Murray ends 77-year wait for British win	Murray wins Wimbledon title ends Britains 77year agony	-3,94
The process must happen in the blink of an eye.	The process must be held in a heartbeat.	-3,87
What the what?! ?: Voice of Charlie Brown arrested, charged. ?	Good grief! Charlie Brown actor charged	-3,45

Skip Thoughts Vectors	**Overestimation**	**Error**
Vietnamese citizens need a visa to visit the USA.	Nepalese citizens require a visa to visit the UK.	2,52
The PCA (format used by the company and its Apple iPods taken from them), meanwhile, is less course.	AAC (the format used by Apple and its iPods), meanwhile, is less current.	2,18
The act of purchasing back something previously sold.	The act of explaining	2,08

Skip Thoughts Vectors	**Underestimation**	**Error**
This frame covers words that name locations as defined politically, or administratively.	The territory occupied by a nation	-2,57
Someone or something that is the agent of fulfilling desired expectations	Someone (or something) on which expectations are centered.	-1,88
The quality of being important, worthy of attention	The quality of being important and worthy of note.	-1,76

References

Eneko Agirre, Carmen Banea, Daniel M. Cer, Mona T. Diab, Aitor Gonzalez-Agirre, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In Steven Bethard, Daniel M. Cer, Marine Carpuat, David Jurgens, Preslav Nakov, and Torsten Zesch, editors, *SemEval@NAACL-HLT*. The Association for Computer Linguistics, pages 497–511.

Steven Bethard, Marine Carpuat, Marianna Apidianaki, Saif M. Mohammad, Daniel Cer, and David Jurgens, editors. 2017. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Association for Computational Linguistics, Vancouver, Canada. http://www.aclweb.org/anthology/S17-2.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-thought vectors. *CoRR* abs/1506.06726. http://arxiv.org/abs/1506.06726.

Geoffrey Leech. 2016. *Word frequencies in written and spoken english: based on the british national corpus*. Routlege.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. 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*, Curran Associates, Inc., pages 3111–3119. http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* https://doi.org/10.3115/v1/d14-1162.

Richard Socher, Eric H. Huang, Jeffrey Pennington, Andrew Y. Ng, and Christopher D. Manning. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Proceedings of the 24th International Conference on Neural Information Processing Systems*. Curran Associates Inc., USA, NIPS'11, pages 801–809. http://dl.acm.org/citation.cfm?id=2986459.2986549.

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text 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*. Association for Computational Linguistics, Stroudsburg, PA, USA, SemEval '12, pages 441–448. http://dl.acm.org/citation.cfm?id=2387636.2387708.

Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *arXiv preprint arXiv:1506.06724* .