Semantic textual similarity

Kristijan Biščanić, Karlo Dumbović, Nino Jagar

University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
`{kristijan.biscanic,karlo.dumbovic,nino.jagar}@fer.hr`

Abstract

This paper focuses on building a system that can assess the semantic similarity between two texts. Each text consists of only one sentence, so more sophisticated approach should be used than just simple word overlap. The sentences are first being preprocessed and then specific features are extracted. The features are being used to train a support vector regression model. The model outputs similarity score which is then compared with human similarity judgements. The performance of the model is being evaluated using Pearson and Spearman correlation coefficients.

1. Introduction

Semantic similarity is a metric defined over a set of documents or terms, where the idea of distance between them is based on likeness of their meaning or semantic content as opposed to similarity which can be estimated regarding their syntactical representation. We have implemented a system that can measure the semantic similarity between two sentences whose context is previously unknown and of no matter to us. The idea of building such system comes from the necessity of automated similarity estimating in a number of study fields such as biomedical informatics, geoinformatics, linguistics and natural language processing (NLP).

The main task of the system is to compute certain features which act as a representation of sentence similarity. The features are being computed from preprocessed sentences. Preprocessing is done kako?... Exact features used are: ngram overlap, WordNet-augmented overlap, weighted word overlap, vector-space similarity, 2 types of normalized differences, shallow NERC and numbers overlap. Calculation of the features will be briefly described in forthcoming sections. These features are used as an input to a support vector regression model. The model is being hyper-optimized by grid search over different values of regularization factor, nesto i nesto. On it's output, the model gives us the similarity judgement based on the features extracted. The similarity is scored as a real number from 0 to 5, where 0 represents no similarity, while 5 represents maximum similarity. Model estimations are compared with human judgements, and the accuracy of the model is measured using Pearson and Spearman correlation coefficients.

2. Overview of the field

Semantic textual similarity is discussed at length in (Šarić et al., 2012). The system described there directly inspired the implementation of our system.

TODO - jos?

TODO - glupo citiranje ne radi jer je retardirani latex odlucio izbrisat mojih mukotrpnih sat vremena posla

3. Description of the System

We used a Support Vector Regression model (SVR) as our learning model. Our system first does a preprocessing step, and then we compute various features from preprocessed sentences.

3.1. Preprocessing

To make our system more robust to small differences in inputs, we use the following preprocessing steps on each sentence:

1. All dashes, brackets, slashes and hyphens are stripped;
2. Various quotes are replaced with regular ' and ";
3. Words are lowercased for calculation of most features;
4. Names of currencies are stripped from the values, e.g., *€EUR100* becomes *€100*;
5. Words are tokenized using the pre-trained NLTK Punkt tokenizer;
6. Tokens *'m*, *n't* and *'re* are changed to *am*, *not* and *are*, respectively;
7. If a compound appears in one sentence, and it also appears in the other sentence but as two consecutive words, then they are replaced by that compound. E.g., *foot ball* from first sentence will be replaced with word *football* if it appears in the other sentence;
8. Words are POS-tagged using the Maxent Treebank POS-tagger from NLTK;
9. For calculation of some features, we removed stopwords using the NLTK stopwords corpus;
10. We performed lemmatization for calculation of some features using the WordNet corpus from NLTK;

3.2. Ngram Overlap

The ngram overlap between two sentences is the harmonic mean of the degree to which the first sentence overlaps with the second and the degree to which the second sentence overlaps with the first.

First we compute S_1 and S_2, sets of consecutive ngrams from first and second sentence, respectively. The ngram overlap is then calculated using the following equation:

$$overlap(S_1, S_2) = 2 \cdot \frac{|S_1 \cap S_2|}{|S_1| + |S_2|} \tag{1}$$

Our system uses overlap scores based on unigrams, bigrams and trigrams. We calculated the overlap on both regular words and lemmas.

3.3. Ostali featurei

valjda ce stat svi? smijemo imat max 3 strane bez referenci

4. Results

4.1. Model Training

We used LIBSVM to train a separate SVR model for each training set. The model was hyper-optimized by grid search with nested cross-validation (in terms of Pearson correlation) to find the optimal parameters C, g and p. Final prediction results are then trimmed to a 0-5 interval. For the surprise test set SMTnews we trained our system on SMTeuroparl train set, and for the OnSw test set we trained the system on the union of all provided train sets. The performance on train sets is shown in Table 1.

TODO - koji su dobri/losi featurei?

Table 1: Cross-validated results on train sets

Set	Pearson	Spearman	C	g	p
MSRvid	1.0000	1.0000	1	1	1
MSRpar	1.0000	1.0000	1	1	1
SMTeuroparl	1.0000	1.0000	1	1	1

4.2. Test Set Results

We evaluated our model using Pearson and Spearman correlation coefficients. The performance on test sets is shown in Table 2. Aggregate performance according to three aggregate evaluation measures proposed in (Agirre et al., 2012) are shown in Table 3.

TODO - na kakvim vrstama primjera failamo i zakaj?

Table 2: Performance on test sets

Set	Pearson	Spearman
MSRvid	1.0000	1.0000
MSRpar	1.0000	1.0000
SMTeuroparl	1.0000	1.0000
SMTnews	1.0000	1.0000
OnSw	1.0000	1.0000

Table 3: Aggregate performance on test sets

	Pearson	Spearman
ALL	1.0000	1.0000
ALLnrm	1.0000	1.0000
Mean	1.0000	1.0000

5. Conclusion and Future Work

In this paper we presented our system for assessing the semantic textual similarity between two short texts based on machine learning.

TODO

References

Eneko Agirre, Daniel Cer, Mona Diaba, 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.

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.

Alexander Budanitsky and Graeme Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Yuri Lin, Jean-Baptiste Michel, Erez Lieberman Aiden, Jon Orwant, Will Brockman, and Slav Petrov. 2012. Syntactic annotations for the google books ngram corpus. In *Proceedings of the ACL 2012 system demonstrations*, pages 169–174. Association for Computational Linguistics.

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*, pages 385–393. Association for Computational Linguistics.