USFD at SemEval-2016 Task 1: Putting different State-of-the-Arts into a Box

Ahmet Aker, Frederic Blain, Andres Duque*, Marina Fomicheva+, Jurica Seva, Kashif Shah and **Daniel Beck**
University of Sheffield, UK
University of Madrid, Spain*
Universitat Pompeu Fabra, Spain+
ahmet.aker@ f.blain@ j.seva@ kashif.shah@ sheffield.ac.uk
andres.duq.fer@gmail.com, marina.fomicheva@upf.edu
debeck1@sheffield.ac.uk

Abstract

In this paper we describe our participation in the STS Core subtask which is the determination of the monolingual semantic similarity between pair of sentences. In our participation we adapted state-of-the-art approaches from related work applied on previous STS Core subtasks and run them on the 2016 data. We investigated the performance of single methods but also the combination of them. Our results show that Convolutional Neural Networks (CNN) are superior to both the Monolingual Word Alignment and the Word2Vec approaches. The combination of all the three methods performs slightly better than using CNN only. Our results also show that the performance of our systems varies between the datasets.

1 Introduction

Semantic Textual Similarity (STS) is a metric which aims to determine the likeness between two short textual entities. Therefore, STS is widely used in many research areas such as Natural Language Processing, for a large amount of tasks like Information Retrieval (IR), Natural Language Understanding (NLU) or even Machine Translation (MT) evaluation, for which STS allows capturing more information than traditional metrics based on n-grams match like BLEU (Papineni et al., 2002).

Part of the SemEval campaign, STS competition benefits from a growing interest over the year (Agirre et al., 2012; Agirre et al., 2013; Agirre et al., 2014; Agirrea et al., 2015). In 2016, participants had the possibility to compete in two tasks: (i) STS split into 'STS Core' and 'Cross-lingual STS' which are respectively an English monolingual and English/Spanish bilingual subtasks; (ii) iSTS which focuses on the interpretable aspect of STS assessment. Introduced for the first time in 2015, iSTS became a standalone task this year.

This paper describes our participation in the STS Core subtask and is organised as follows: we first describe in Section 2 methods we have adapted to tackle the task. Next, we present and discuss our results (Section 3). Finally, Section 4 will conclude this system description paper with some remarks.

2 System description

For adressing monolingual semantic similarity between two sentences different methods have been proposed in related work. In our monolingual STS participation we have focused on some methods that were reported as state-of-

Proceedings of SemEval-2016, pages 609–613,
San Diego, California, June 16-17, 2016.

the-art systems. Given that those methods have been reported separately a natural question that arises from this is what the performance is when those methods are combined. This is exactly what we did and what we propose in this paper. We adapted methods from related work that have been applied on the monolingual STS task and used their combined version on the new 2016 monolingual task. Methods adapted and the strategy we used to combine them are described in the following sections.

2.1 Monolingual Alignment

We use an alignment-based approach, which was among the top-perfoming submissions in the past year's task (Sultan et al., 2015). Sentence similarity score is computed in two separate steps. First, an alignment between related words in the input sentences is established using Monolingual Word Aligner (MWA) (Sultan et al., 2014). Next, sentence similarity is calculated based on the proportion of aligned content words:

$$sim(S^1, S^2) = \frac{n_c^a(S^1) + n_c^a(S^2)}{n_c(S^1) + n_c(S^2)} \quad (1)$$

where $n_c(S^i)$ and $n_c^a(S^i)$ are the number of content words and the number of aligned content words in sentence *i*, respectively. MWA makes alignment decisions based on lexical similarity and contextual evidence. Lexical similarity component identifies word pairs that are possible candidates for alignment. Context words are considered as evidence for alignment if they are lexically similar and have the same or equivalent syntactic relations with the words to be aligned. Word pairs are aligned in decreasing order of a weighted sum of their lexical and contextual similarity.

2.2 Convolutional Neural Network Score

Another method adapted in our system is a Convolutional Neural Network (CNN), which generates a similarity score for each pair of sentences. More specifically, we replicated the system presented in (He et al., 2015), using previous SemEval data for training the network and generating similarity values between 0 and 5, for each of the test sentences given by the organizers. In the following subsections, we briefly summarize the CNN system, although specific details can be found in the cited paper.

Word Representation

Words in the sentences to be compared are first transformed into vectors using the GloVe word embeddings (Pennington et al., 2014). These embeddings are trained on 840 billion tokens, and the resulting vectors are 300-dimensional. Hence, each sentence $sent$ with n words will be transformed into a matrix $M_{sent} \in \mathbb{R}^{n \times 300}$. Hence, $sent_{i:j}$ denotes the word embeddings of words i to j inside the sentence, $sent_i^{[k]}$ denotes the k-th dimension of word embedding i and $sent_{i:j}^{[k]}$ the k-th dimension of words i to j inside the sentence.

Sentence modelling

The technique makes use of two different types of filters for extracting features from the sentences: holistic and per-dimension. Holistic filters generate a vector representing a "temporal" convolution, this is, complete regions of the word sequence. On the other hand, per-dimension filters perform spatial convolutions, limited to a predefined dimension k. After the application of those filters, the last step of the convolutional layer is to perform pooling operations over the vectors generated by the filters, in order to convert those vectors into scalar values. The system defines three types of pooling: max, min and $mean$. Finally, the system also defines a $group$ as a specific convolution layer (with either holistic or per-dimension filters) with width ws and a specific type of pooling, which operates over a sentence. A set of groups composed by convolution layers with the same width which explore different pooling functions is called a $block$.

Similarity Measurement Layer

Once that the sentence representation is done through the use of filters and pooling functions, a way to compare two sentences has to be defined. The comparison of two different sentences is made over local regions of the sentence representation. For this purpose, the system consider the output of the convolutional layers in order to perform both "horizontal" and "vertical" comparisons. Given two vectors, each of them representing the same region of an input sentence, a new output vector is created by measuring cosine distance, L_2 Euclidean distance and element-wise absolute difference. This output vector is added to an accumulated vector which stores the outputs of all the compared regions of the input sentences. The final output of the system generates an output similarity score through a final log-softmax layer, which receives the accumulated vector. System parameters (window size, number of filters, learning rate, regularization parameter, hidden units) are maintained with respect to the original work in which the system is presented.

2.3 Word2Vec

Word embeddings using Word2vec (Mikolov et al., 2013) have been extensively used to measure the semantic similarity between words. Our word embeddings comprise the vectors published by Baroni et al. (2014). To measure the similarity between a pair of sentences we first remove from each sentence stop-words as well as punctuations, query for each word its vector representation and create a averaged sum of the word vectors. The number of remaining words in each sentence is used to average that sentence. Finally, we use the resulting averaged sum vectors and determine their similarity using cosine.

2.4 Model Combination

In this section, we present the experiments to combine all methods described in previous sections. We formulated the problem as a regression task where we are given multiple features capturing different attributes of inputs along with gold labels and we predict the output for unseen examples. Support Vector Regression (SVR) (Chang and Lin, 2001) is the most commonly used algorithm for such tasks. We used the 3 features described in previous sections and trained SVR models to estimate a continuous score within [0,5]. We evaluated different settings of these models including various available kernels. Based on optimum performance, we used a radial basis function (RBF) kernel, which has been shown to perform very well in quality estimation tasks (Callison-Burch et al., 2012). Kernel parameters are optimised using grid search with 5-fold cross-validation. The correlation scores using SVR as learning algorithm are reported in Table 3. It can be seen that adding all features together improves the results in as compared to the individual scores for each of the methods.

Application on SemEval Data

The described system has been trained with data from previous Semantic Textual Similarity tasks of SemEval, more specifically data from 2012, 2013, 2014 and 2015. The final training dataset is composed by 13,560 sentence pairs.

3 Results

In SemEval STS Core Task the performance of each method (or system) is evaluated using Pearson Correlation that measures the linear correlation between the system's outputs and gold-standard data. A score of 1 indicates 100% correlation and 0 no correlation at all. Table 3 shows the individual results for methods adapted in this work as well as the result when all methods are combined together as a single system.

We can first observe that the best single performing method is CNN, with an overral achievement of 0.727 correlation score (see last column). The other two approaches achieve

Method	answer-answer	headlines	plagiarism	postediting	question-question	OverALL
CNN	0.510	0.818	**0.834**	0.792	0.685	0.727
MWA	0.436	0.704	**0.749**	0.587	0.579	0.611
Word2vec	0.276	0.642	**0.787**	0.750	0.688	0.622
Combination	0.508	0.820	**0.838**	0.794	0.689	0.728

Table 1: Pearson correlation between the prediction and gold standard data.

substantially lower correlation figures with a drop of about 0.10. Monolingual Word Alignment is the less performing method achieving 0.611, just behind Word2Vec with 0.622. From our results we can also remark that the combined version of all the three methods lead to 0.728 which is similar to the CNN method only.

Secondly, if we look at the results for each dataset individually, we can also remark that the performance of our system drastically change from one to another. While all approaches perform poorly on the answer-answer data, they achieve high correlation scores on the headlines, post-editing and plagiarism data sets. The latter being the data on which our approaches are the most efficient with a minimum of 0.749 correlation.

4 Conclusions and further studies

In this paper we described our participation to the STS Core Task for SemEval 2016. We adapted state-of-the-art methods from previous studies applied on the monolingual semantic similarity task and run them on the 2016 data. We investigated the performance of single methods individually but also combined altogether.

Overall, our results show that the CNN-based approach was the most effective compared to the others individual approaches. The combination of those methods achieves slightly better results than CNN only. We can also observe that results vary between the different dataset: they are highly satisfactory on both the headlines and plagiarism dataset, but perform poorly on the answer-answer data, especially Word2Vec with only 0.276 Pearson Correllation compared to gold-standard.

For the future work we aim to conduct a deeper analysis about the performance of our different systems. This will also include the understanding concerning their effectiveness over on diverse datasets. Finally, we would like to investigate the QuEst framework (Specia et al., 2013) as additional method to describe the semantic similarity between sentences:

QuEst framework

For our future work we aim to use the QuEst framework (Specia et al., 2013) and extract features to capture the semantic similarity between monolingual sentences. These features are used and have shown to perform well in the WMT shared tasks on QE. They include simple counts, e.g. number of tokens in the segments, language model probabilities and perplexities, number of punctuation marks in source and target segments, among other features reflecting the complexity and fluency of the given segments. Though these features are originally designed to estimate the quality of machine translation, we aim to adapt and explore their potential in addition to the methods discussed in previous sections.

Acknowledgements

The research leading to these results has received funding from the EU - Seventh Framework Program (FP7/2007-2013) under grant agreement n610916 SENSEI.

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. sem 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 Agirrea, Carmen Baneab, Claire Cardiec, Daniel Cerd, Mona Diabe, Aitor Gonzalez-Agirrea, Weiwei Guof, Inigo Lopez-Gazpioa, Montse Maritxalara, Rada Mihalceab, 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)*, pages 252–263.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *ACL (1)*, pages 238–247.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia, editors. 2012. *Proceedings of the Seventh Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Montréal, Canada, June.

Chih-Chung Chang and Chuan-bi Lin. 2001. Training v-support vector classifiers: theory and algorithms. *Neural computation*, 13(9):2119–2147.

Hua He, Kevin Gimpel, and Jimmy Lin. 2015. Multi-perspective sentence similarity modeling with convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1576–1586.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Juillet.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543.

Lucia Specia, Kashif Shah, Jose G. C. De Souza, Trevor Cohn, and Fondazione Bruno Kessler. 2013. Quest - a translation quality estimation framework. In *In Proceedings of the 51th Conference of the Association for Computational Linguistics (ACL), Demo Session*.

Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014. 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. 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.