STS-UHH at SemEval-2017 Task 1: Scoring Semantic Textual Similarity Using Supervised and Unsupervised Ensemble

Sarah Kohail* LT Group, CS Department, Universität Hamburg

Amr Rekaby Salama* NATS Group, CS Department, Universität Hamburg

Chris Biemann LT Group, CS Department, Universität Hamburg

{kohail,salama,biemann}@informatik.uni-hamburg.de

Abstract

This paper reports the STS-UHH participation in the SemEval 2017 shared Task 1 of Semantic Textual Similarity (STS). Overall, we submitted 3 runs covering monolingual and cross-lingual STS tracks. Our participation involves two approaches: unsupervised approach, which estimates a word alignment-based similarity score, and supervised approach, which combines dependency graph similarity and coverage features with lexical similarity measures using regression methods. We also present a way on ensembling both models. Out of 84 submitted runs, our team best multi-lingual run has been ranked 12^{th} in overall performance with correlation of 0.61, 7^{th} among 31 participating teams.

1 Introduction

Semantic Textual Similarity (STS) measures the degree of semantic equivalence between a pair of sentences. Accurate estimation of semantic similarity would benefit many Natural Language Processing (NLP) applications such as textual entailment, information retrieval, paraphrase identification and plagiarism detection (Agirre et al., 2016). In an attempt to support the research efforts in STS, the SemEval STS shared Task (Agirre et al., 2017) offers an opportunity for developing creative new sentence-level semantic similarity approaches and to evaluate them on benchmark datasets. Given a pair of sentences, the task is to provide a similarity score on a scale of 0..5 according to the extent to which the two sentences are considered semantically similar, with 0 indicating that the semantics of the sentences are completely unrelated and 5 signifying semantic equivalence. Final performance is measured by computing the Pearson's correlation (ρ) between machine-assigned semantic similarity scores and gold standard scores provided by human annotators. Since last year, the STS task have been extended to involve additional subtasks for cross-lingual STS. Similar to the monolingual STS task, the cross-lingual task requires the semantic similarity measurement for two snippets of text that are written in different languages. In contrast to last year's edition (Agirre et al., 2016), the task is organized into 6 sub-tracks and a primary track, which is the average of all of the secondary sub-tracks results. Secondary sub-tracks involve scoring similarity for monolingual sentence pairs in one language (Arabic, English, Spanish), and cross-lingual sentence pairs from the combination of two different languages (Arabic-English, Spanish-English, Turkish-English).

Our paper proposes both supervised and unsupervised systems to automatically scoring semantic similarity between monolingual and cross-lingual short sentences. The two systems are then combined with an average ensemble to strengthen the similarity scoring performance. Out of 84 submissions, our system is placed 12^{th} with an overall primary score of 0.61.

2 Related Work

Since 2012 (Agirre et al., 2012), the STS shared task has been one of the official shared tasks in SemEval and has attracted many researchers from the computational linguistics community (Agirre et al., 2017). Most of the state-of-the-art approaches often focus on training regression models on traditional lexical surface overlap features. Recently, deep learning models have achieved very promising results in semantic textual sim-

*These authors contributed equally to this work

Proceedings of the 11th International Workshop on Semantic Evaluations (SemEval-2017), pages 175–179,
Vancouver, Canada, August 3 - 4, 2017.

ilarity. The top three best performing systems from STS 2016 used sophisticated deep learning based models (Rychalska et al., 2016; Brychcín and Svoboda, 2016; Afzal et al., 2016). The highest correlation score was obtained by Rychalska et al. (2016). They proposed a textual similarity model that combines recursive auto-encoders (RAE) from deep learning with WordNet award penalty, which helps to adjusts the Euclidean distance between word vectors.

3 System Description

Our contribution in the STS shared task includes three different systems: supervised, unsupervised and supervised-unsupervised ensemble. Our models are mainly developed to measure semantic similarity between monolingual sentences in English. For the cross-lingual tracks, we leverage the Google translate API to automatically translate other languages into English. In the following subsections, we describe our data preprocessing and present our three systems.

3.1 Data Preprocessing

We use all the previously released datasets since 2012 to train and evaluate our models. The final total number of training examples is 14 619. We use StanfordCoreNLP[1] pipeline to tokenize, lemmatize, dependency parse, and annotate the dataset for lemmas, part-of-speech (POS) tags, and named entities (NE). Stopwords are removed for the purpose of topic modeling and TfIdf computation.

3.2 Unsupervised Model

Inspired by (Sultan et al., 2015; Brychcín and Svoboda, 2016), our unsupervised solution calculates a similarity score based on the alignment of the input pair of sentences. As presented in Figure 1, given a pair of sentences $S1, S2$, the alignment task builds a set of matched pair of words $match(w_i, w_j)$ where w_i is a word in sentence $S1$, and w_j is a word in sentence $S2$. Each matched pair has a score on the scale [0-1]. This matching score indicates the strength of the semantic similarity between the aligned pair of words, with 1 representing the highest similarity match.

As shown in Figure 2, after preprocessing, the system starts with matching exact similar words

[1] http://stanfordnlp.github.io/CoreNLP/

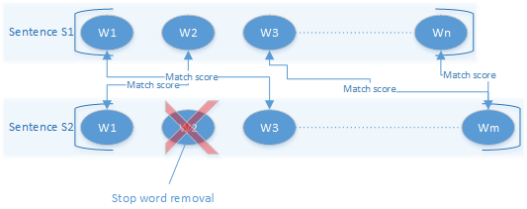

Figure 1: Unsupervised sentence alignment

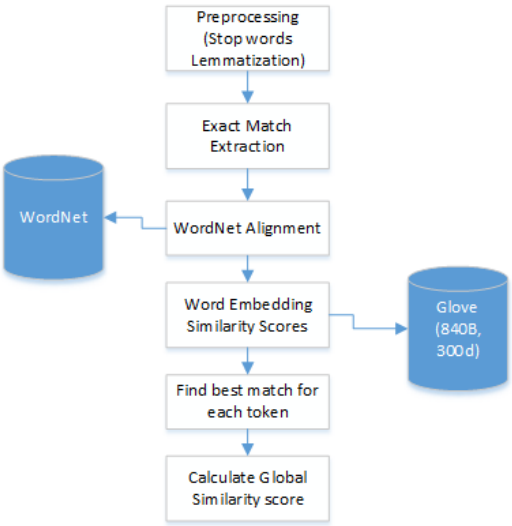

Figure 2: Unsupervised solution overview

(lemmas), and words that share similar WordNet hierarchy (synonyms, hyponyms, and hypernyms). We consider these two types of aligning as exact match with score 1.

As a last step of the alignment process, we handle the words that have not been matched in the preceding steps. The solution uses Glove word embeddings (Pennington et al., 2014) to calculate the matching score. Glove (840B tokens, 2.2M vocab) represent the word embeddings in 300d vector. We calculate the cosine distance between the unmatched words and all the words in the other sentence. Using a greedy strategy, we pick up the best match of each word.

The global similarity is calculated using a weighted matches scores as shown in equation (1).

$$Score = \frac{\sum TfIdf(w_i) * match(w_i, w_j)}{\sum TfIdf(S1, S2)} \quad (1)$$

For all w_i in $S1$ or $S2$, and $match(w_i, w_j)$ is the best match score for W_i with word W_j from the other sentence. $TfIdf(S1, S2)$ is the sum of the term frequency inverse document frequency of the words in $S1, S2$. The final alignment score is [0-

1], so we scale it into the [0-5] range.

3.3 Supervised Model

To generate our supervised model, we extract the following features:

I **Bag-of-Words:** for each sentence a |V|-dimension vector is generated, where V includes the unique vocabulary from both sentences. Entries in single vectors correspond to the frequency of the word in the respective sentence. Cosine similarity between these vectors serves as a feature.

II **Distributional Thesaurus (DTs) Expansion Feature:** Each non-stopword is expanded to its most similar top 10 words using the API for the Distributional Thesaurus (DTs) by Biemann and Riedl (2013).

III **POS Tags Longest Common Subsequence:** We measure the length of the longest common subsequence of POS tags between sentence pairs. Additionally, we also average this length by dividing it by the total number of tokens in each sentence separately.

IV **Topic Similarity Feature:** To model the topical similarity between two documents, we use Latent Dirichlet Allocation (LDA, (Blei et al., 2003))[2] model trained on a recent Wikipedia dump. To guarantee topic distribution stability, we run LDA for 100 repeated inferences. Then for each token, we assign the most frequent topic ID (Riedl and Biemann, 2012).

V **Dependency-Graph Features:** Following Kohail (2015), each sentence S is converted into a graph using dependency relations obtained from the parser. We define the dependency graph $G_S = \{V_S, E_S\}$, where the graph vertices $V_S = \{w_1, w_2, \ldots, w_n\}$ represent the tokens in a sentence, and E_S is a set of edges. Each edge e_{iy} represents a directed dependency relation between w_i and w_y. We calculate TfIdf on three levels and weight our dependency graph using the following conditions:
Word TfIdf: Considering only those words that satisfy the condition: TfIdf $(w_i) > \alpha_1$
Pair TfIdf: Word pair are filtered based on the condition: TfIdf $(w_i, w_y) > \alpha_2$
Triplet TfIdf: Considering only those triples (word, pair and relation), which satisfies the condition: TfIdf $(w_i, w_y, e_{iy}) > \alpha_3$
Similarities are then measured on three levels by representing each sentence as a vector of words, pairs and triples, where each entry in one vector is weighted using TfIdf. We used New York Times articles within the years 2004-2006, as a background corpus for TfIdf calculation.

VI **Coverage Features:** As a text gets longer, term frequency factors increase, and thus having a high similarity score is likelier for longer than for shorter texts. Coverage features measures the number of one-to-one tokens, edges and relations correspondence between the dependency graphs of a pair sentences as described in (Kohail and Biemann, 2017).

VII **NE Similarity:** We measure similarity based on the shared named entities between the pair of text.

VIII **Unsupervised Dependency Alignment score:** Using a Glove word embedding, we include the score of the cosine similarity between the syntactic heads of the matched words aligned in the unsupervised model (Sec. 3.2), as presented in equation (2).

$$score = \frac{\sum TfIdf(\widehat{w_i}) * Cos\_sim(\widehat{w_i}, \widehat{w_j})}{\sum TfIdf(S1, S2)} \quad (2)$$

For all w_i in $S1$ or $S2$, we calculate the weighted cosine similarity between its syntactic dependency head: $\widehat{w_i}$ and the syntactic head of the matched word: $\widehat{w_j}$.

These features are fed into three different regression methods[3]: Multilayer Perceptron (MLP)[4] neural network, Linear Regression (LR) and Regression Support Vector Machine (RegSVM). To evaluate our preliminary pre-testing models, we perform 10-fold cross-validation.

[2]The implementation was used in this work is available at: http://gibbslda.sourceforge.net/

[3]We used the WEKA (Witten et al., 2016) implementation with default parameters, if not mentioned otherwise

[4]Hidden layers = 2, Learning rate = 0.4, momentum = 0.2

System	Primary	Track 1 AR-AR	Track 2 AR-EN	Track 3 SP-SP	Track 4a SP-EN	Track 4b SP-EN	Track 5 EN-EN	Track 6 EN-TR
Run1	0.57	0.61	0.59	0.72	0.63	0.12	0.73	0.60
Run2	**0.61**	0.68	0.63	0.77	0.72	0.05	0.80	0.59
Run3	-	-	-	-	-	-	**0.81**	-
Ens.*	0.63	0.68	0.66	0.80	0.73	0.11	0.82	0.63
Basel.	-	0.60	-	0.71	-	-	0.73	-
Top	0.73	0.75	0.75	0.85	0.83	0.34	0.85	0.77

Table 1: Results obtained in terms of Pearson correlation over three runs for all the six sub-tracks in comparison with the baseline and the top obtained correlation in each track. The primary score represents the weighted mean correlation. Ens.* represents the results after adding the expansion and topic modeling features.

3.4 Ensembling Supervised and Unsupervised models

We create an ensemble model by by averaging the supervised and unsupervised models predictions.

4 Experimental Results

We report our results in Table 1. Overall we submitted 3 runs: **Run1** uses the unsupervised approach discussed earlier in Sec. 3.2, **Run2** uses a supervised MLP neural network trained as described in Sec. 3.3, and **Run3** uses the ensemble average system described in Sec. 3.4. Due to time constraints and technical issues, only evaluation for English monolingual track was given. Additionally, we were not able to compute the topic modeling and expansion features. We included the missing features later after the task deadline. Final ensemble results are given under Ens.*. According to the results, we can make following observations:

- Our results significantly outperform the baseline provided by the task organizers for monolingual tracks by a large margin.
- The ensemble outperforms the individual ensemble members.
- Results obtained in monolingual, especially English, are markedly higher than in cross-lingual tracks. This might be due to noise introduced by the automatic translation.
- Results of track 4b appears to be significantly worse compared to other tracks results. In addition to the machine translation accuracy challenge, the difficulty of this track lies in providing longer sentences with less informative surface overlap between the sentences compared to other tracks.

5 Conclusion

We have presented and discussed our results on the task of Semantic Textual Similarity (STS). We have shown that combining supervised and unsupervised models in an ensemble provides better results than when each is used in isolation. 31 teams participated in the task with 84 runs. Our best system achieves an overall mean Pearson's correlation of 0.61, ranking 7^{th} among all teams, 12^{th} among all submissions. Future work includes building a real multi-lingual model by projecting phrases from different languages into the same embedding space. In the current solution, we consider hyponyms/hypernyms as synonyms. The system gives an exact match score for these word pairs. In the future, we tackle finding a way to give calculated dynamic scores for such kind of alignment to do not equalize them with exact matches.

Acknowledgment

This research was supported by the Deutscher Akademischer Austauschdienst (DAAD).

References

Naveed Afzal, Yanshan Wang, and Hongfang Liu. 2016. MayoNLP at SemEval-2016 Task 1: Semantic Textual Similarity based on Lexical Semantic Net and Deep Learning Semantic Model. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California, pages 674–679.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona 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 *Proceedings of SemEval*. San Diego, California, pages 497–511.

Eneko Agirre, Daniel Cer, Mona Diab, Iigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation. In *Proceedings of SemEval*. Vancouver, Canada.

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*. Montreal, Canada, SemEval '12, pages 385–393.

Chris Biemann and Martin Riedl. 2013. Text: Now in 2D! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling* 1(1):55–95.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* 3:993–1022.

Tomáš Brychcín and Lukáš Svoboda. 2016. UWB at SemEval-2016 Task 1: Semantic Textual Similarity using Lexical, Syntactic, and Semantic Information. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California, pages 588–594.

Sarah Kohail. 2015. Unsupervised Topic-Specific Domain Dependency Graphs for Aspect Identification in Sentiment Analysis. In *Proceedings of the Student Research Workshop associated with RANLP*. Hissar, Bulgaria, pages 16–23.

Sarah Kohail and Chris Biemann. 2017. Matching, Reranking and Scoring: Learning Textual Similarity by Incorporating Dependency Graph Alignment and Coverage Features. In *18th International Conference on Computational Linguistics and Intelligent Text Processing*. Budapest, Hungary.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. pages 1532–1543.

Martin Riedl and Chris Biemann. 2012. Sweeping through the topic space: Bad luck? roll again! In *Proceedings of the Joint Workshop on Unsupervised and Semi-Supervised Learning in NLP*. Association for Computational Linguistics, Avignon, France, ROBUS-UNSUP '12, pages 19–27.

Barbara Rychalska, Katarzyna Pakulska, Krystyna Chodorowska, Wojciech Walczak, and Piotr Andruszkiewicz. 2016. Samsung Poland NLP Team at SemEval-2016 Task 1: Necessity for diversity; combining recursive autoencoders, WordNet and ensemble methods to measure semantic similarity. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. San Diego, California, pages 602–608.

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 (SemEval 2015)*. Association for Computational Linguistics, Denver, Colorado, pages 148–153.

Ian H. Witten, Eibe Frank, Mark A. Hall, and Christopher J. Pal. 2016. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.