BL.Research at SemEval-2022 Task 8: Using various Semantic Information to evaluate document-level Semantic Textual Similarity

Sébastien Dufour[†], Mohamed Mehdi Kandi[*], Karim Boutamine[†], Camille Gosset[*], Mokhtar Boumedyen Billami[*], Christophe Bortolaso[*], Youssef Miloudi[†*]
[†]CARL Berger-Levrault, 361 All. des Noisetiers, 69760 Limonest, France
[*]Berger-Levrault, 64 Rue Jean Rostand, 31670 Labège, France
{sebastien.dufour, youssef.miloudi}@carl.eu, {mohamed.kandi, karim.boutamine, camille.gosset, mb.billami, christophe.bortolaso}@berger-levrault.com

Abstract

This paper presents our system for document-level semantic textual similarity (STS) evaluation at SemEval-2022 Task 8: "Multilingual News Article Similarity". The semantic information used is obtained by using different semantic models ranging from the extraction of key terms and named entities to the document classification and obtaining similarity from automatic summarization of documents. All these semantic information's are then used as features to feed a supervised system in order to evaluate the degree of similarity of a pair of documents. We obtained a Pearson correlation score of 0.706 compared to the best score of 0.818 from teams that participated in this task. Our source code can be found at GitHub[1].

1 Introduction

Measuring semantic textual similarity has been a research subject in natural language processing, information retrieval and artificial intelligence for many years. Accurate modelling of textual similarity is fundamental for many applications. Previous efforts have focused on comparing a short text with a long text (e.g., Web search), two sentences or other short text sequences (e.g., paraphrase recognition, image retrieval by captions and Twitter tweets search). There are many other tasks requiring computing the semantic similarity between two long texts (e.g., document classification, document clustering and tracking the similarity of news coverage between different regions).

Semantic textual similarity (STS) measures the level of semantic equivalence between two textual contents. In this paper, we are interested to develop a system that identifies news articles that provide similar semantic information. More specifically, given two news articles, we aim to compute their similarity based on four characteristics: geolocation, time, shared entities, and shared narratives. To achieve this goal, we must identify important elements of the news articles content, such as the discussed event, location, time, and people involved. More precisely, the task 8 of SemEval-2022 edition focuses on the analysis of documents (long texts) that can share different semantic information and can be from different natural languages in both monolingual and cross-lingual settings.

The STS task has been held in SemEval since 2012. More precisely, the semantic similarity systems between sentences using paraphrase datasets have been proposed (Agirre *et al.*, 2012). In the 2013 edition of Joint Conference on Lexical and Computational Semantics – *SEM (Agirre *et al.*, 2013), STS task was focused in using cultural heritage items which are described with metadata such as title, author, or description. Thereafter, systems that compare snippets of text are proposed in the 2016 edition of SemEval (Agirre *et al.*, 2016). Afterwards, the 2017 edition of SemEval proposed to compare sentences in a dimension of monolingual and cross-lingual contents (Cer et al., 2017). In the last few years, many semantic

[1] https://github.com/jln-brtn/BL.Research-at-SemEval-2022-Task-8

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1221 - 1228

similarity datasets and systems have been explored. Among the elements that characterize this new edition of SemEval-2022 compared to 2017 is the fact that we are interested in measuring the similarity between long news contents and not sentences, snippets of texts or short texts.

We encountered many challenges: First, the content scraping of the provided URL (in the first attempt, we had empty or incomplete content in many examples). Then, we have many languages (new languages in the test set are not present in the train set). Finally, it was challenging to choose the type of the predicted score: decimal or integer, which is different according to the number of human annotators per document.

In this paper, we present our approach for SemEval-2022 Task 8. The system that we propose is based on the computation of different scores by document pairs. These scores are used as features to train a supervised system. Regarding the multiple languages, we have developed a dedicated model for some of them, and translations in a pivot language for others.

The paper is organized as follows: we describe the problem and the data in section 2. Then, we give an overview of the related work in section 3. Next, we present the proposed system in section 4. We detail the performed experiments and the results in sections 5 and 6. Finally, we conclude the paper in section 7.

2 Background

2.1 Problem Description

The objective of this task at SemEval-2022 is to compare the similarity between two news articles and to be as consistent as possible with manual annotations provided by native annotators. We identified several challenges and issues in this task:

- **Scraping of the data**: probably not foreseen, because the final access to a clean and complete text was not always possible. For example, scraping sometimes empty, incomplete, or not correct (name of the newspaper, badly identified characters, and others).
- **Multilingual aspect of this task**: several languages used, some of which difficult or poorly modeled like Russian, Chinese, Turkish, and also text pair analysis where text content are provided by different languages.
- **Similarity evaluation of texts**: on a precise topic basis, on concordant geographical or temporal elements, on a similarity of tone and common style.

2.2 Data Description

Data used to train consisted of a total of 4,964 pairs of news articles written in seven different languages, namely: English (EN), French (FR), Spanish (SP), German (DE), Polish (PL), Arabic (AR) and Turkish (TR). The pairs were formed either with the same language or with different languages. In the training corpus, seven couple types are monolingual and only one is cross-lingual (DE_EN). For the test corpus, we have a total of 4,953 pairs of news articles and ten different languages with three new languages, namely: Russian (RU), Chinese (ZH) and Italian (IT). In this corpus, we have eight cross-lingual couple types with seven couple types that never seen in train corpus. Table 1 describes the number of document pairs for each corpus and each language pair.

Couple Type	Monolingual		Couple Type	Cross-lingual	
	Train	Test		Train	Test
EN_EN	**1,800**	236	DE_EN	577	190
FR_FR	72	111	SP_EN	—	498
SP_SP	570	243	PL_EN	—	64
DE_DE	857	611	ZH_EN	—	223
PL_PL	349	224	SP_IT	—	320
AR_AR	274	298	DE_FR	—	116
TR_TR	465	275	DE_PL	—	35
RU_RU	—	287	FR_PL	—	11
ZH_ZH	—	**769**			
IT_IT	—	442			

Table 1: statistics on the number of examples of train and test corpus.

Each pair of documents was annotated by one to eight annotators based on seven score categories that are “Geography”, “Entities”, “Time”, “Narrative”, “Overall”, “Style”, and “Tone”. For each category of each pair, a score on a 4-point scale was given by the available annotators then the average resulted in floats or integers. The score ranges from 1 as most similar to 4 as least similar. In addition to the mentioned scoring categories, the article pair identifiers, languages, and URLs were given for each pair. Using the URLs, we were able to retrieve data from the articles such as: titles,

texts, keywords, tags, authors, publication date, abstracts, and meta-descriptions along with other irrelevant information. Finally, evaluation data were released later by organizers in the same format as train data except for the previous cited scoring categories as our system were judged based on one of those scores that is the "Overall score".

3 Related Work

Semantic textual similarity deals with determining how similar two pieces of texts are. This can be done by assigning a rating from 1 to 4 (or 1 to 5) for more similar to less similar content pairs. Related tasks are paraphrasing or duplication identification. There are many interesting works for STS, an Evaluation Toolkit for Universal Sentence Representations, named SentEval has been proposed (Conneau and Kiela, 2018). It includes 17 downstream tasks[2], including common STS tasks from 2012-2016. We describe in this section some existing systems from the previous editions of SemEval.

Tian *et al.* (2017) proposed three feature-engineered models using Random Forest, Gradient Boosting, and XGBoost regression methods. Their features are based on n-gram overlap; edit distance, longest common prefix/suffix/substring, tree kernels, word alignments, to cite a few. They also propose four deep learning methods. The difference between the methods is the approach to sentence embeddings using either: averaged word embeddings, projected word embeddings, a deep averaging network, or LSTM (Long-Short Term Memory) (Hochreiter and Schmidhuber, 1997). To build the global model, they average scores of the deep learning and the feature-engineered models.

Wu *et al.* (2017) use sentence information content with WordNet (Wallace, 2007) and BNC word frequencies (Leech, Rayson and Wilson, 2001). One variant uses sentence information content exclusively. Another variant uses ensembles information content with Sultan, Bethard and Sumner (2015)'s alignment method. The third variant uses ensembles information content with a cosine similarity of summed word embeddings with an IDF (Inverse Document Frequency) weighting scheme (Jones, 1972).

Shao (2017) proposed a convolutional Deep Structured Semantic Model for the generation of sentence embeddings. The embeddings are compared using cosine similarity and element-wise difference with the resulting values fed to additional layers. This architecture is similar to Tian *et al.* (2017)'s deep learning models.

Henderson *et al.* (2017) proposed a feature engineering approach that they complete with deep learning. Ensembled components include alignment similarity; string similarity measures such as matching n-grams, summarization, Machine Translation (MT) metrics, an RNN (Recurrent Neural Networks), and RCNN (Recurrent Convolutional Neural Networks) over word alignments, and a BiLSTM (Bidirectional Long-Short Term Memory) networks.

4 System Overview

Some strategies can be considered to resolve the task 8 of SemEval-2022 :

- Building a model for each training dataset by language: English, French, Spanish, German, Polish, Arabic and Turkish.
- Building a unique model in English[3] and translating all texts into that language.
- Building a multilingual model that can handle two texts into different languages.

The advantage of the first and the third strategies is that they avoid translation to a pivot language. The second strategy has the advantage of enabling consistent training and being able to handle all languages using a single model. Our system is neither based on the translation of all texts in English, nor on the construction of a multilingual model. We have chosen to build learning models in some main languages, namely: English, French, Spanish, German, Arabic and Turkish. We abandoned Polish language because we did not find adequate pretrained models. When two texts to be compared are not in the same language, they are translated into the main language selected.

A fundamental point for the final score to be generated is the choice on the precision of the answer in terms of decimals. We have noticed that for some languages, the Overall score obtained is

[2]http://nlpprogress.com/english/semantic_textual_similarity.html

[3] The global reference language and therefore the one best managed by all techniques in natural language processing.

an integer. In English, for example, there is a decimal score and more annotators. The actual number of annotations cannot be determined in advance. We have therefore defined rules to complete our evaluation according to the score obtained and the language. We did not use the metadata of the news articles (authors, dates, newspaper, tags, etc.) because these data were incomplete. We simply retained both titles and text contents. We also noticed:

- Strong correlations between the Overall score and the scores of Entities, Narrative, and Geography. Table 2 describes the correlations between annotator scores.
- The training dataset is unbalanced, especially in English. We have more than four scores. The figure 1 describes more information about this dataset.

	Geography	Entities	Time	Narrative	Overall	Style	Tone
Geography	1	**0.63**	0.12	0.52	0.59	0.33	0.35
Entities		1	0.25	**0.74**	**0.80**	0.32	0.39
Time			1	0.4	0.43	0.10	0.18
Narrative				1	**0.88**	0.32	0.45
Overall					1	0.33	0.45
Style						1	0.57
Tone							1

Table 2: Correlations between annotator scores.

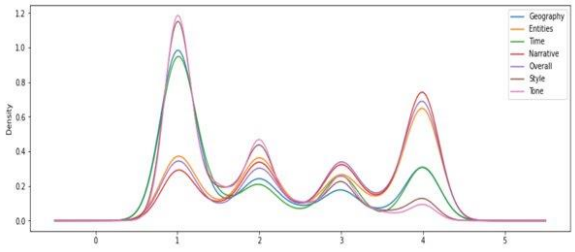

Figure 1 : Density of annotator scores.

Following the observations that are seen in table 2 and figure 1, we implemented a scoring system based on each language who will be features to feed a supervised system. We detail below our features:

1. A similarity score of titles based on sentence transformers with a pretrained encoder model (Reimers and Gurevych, 2019).
2. We used text summarization based on pretrained transformers in each language to measure a similarity score between these summaries. We tested several models on some examples and selected those which seemed to be the best. Based on the language, we were able to obtain one or two models of summaries and thus one or two scores. We have mainly turned to BART (Lewis *et al.*, 2019) or Sequence-to-Sequence models (Chen *et al.*, 2021; Zhang *et al.*, 2019; Shleifer and Rush, 2020; Eddine, Tixier and Vazirgiannis, 2021). The models are pretrained on different summarization corpus variants, for example MLSUM, the Multilingual Summarization Corpus (Scialom *et al.*, 2020).
3. We used identification and extraction of keywords/'key terms' in titles and content texts by using the PKE toolkit library (Boudin, 2016). The extracted tags are nouns, proper nouns, verbs, and adjectives only with stemming adding the ten semantically closest words by using Word2Vec (Mikolov *et al.*, 2013) with Gensim library (Řehůřek and Sojka, 2010), if a model exists in a specific language. We compute the number of common terms in both texts.
4. We used identification and extraction of common named entities between titles and content texts, namely: places, persons, organizations, and dates. We used two libraries: Spacy (Honnibal *et al.*, 2020) and Stanza (Qi *et al.*, 2020), and pretrained transformers models. As for keywords and key terms, we compute the number of identical entities in both texts.
5. For the various geographical detected entities (cities, regions, countries), we calculate a score for the proximity between places based on geocoding[4].
6. We used the zero-shot classification models (Yin, Hay and Roth, 2019) based on Press topics that we have defined manually: *politics*, *sport*, *health*, *economy* and *technology*. The similarity score obtained reflects the number of common topics between two texts.

[4] https://geopy.readthedocs.io/en/stable/#nominatim

7. Finally, sentiment analysis models are used to identify if the sentiment polarity is positive, negative, or neutral in both texts. The models used are proposed by Wolf *et al.* (2019), Guhr *et al.* (2020), Demange (2021) and Pérez, Giudici and Luque (2021). The similarity score obtained thus reflects the number of common points.

With all these features, it is possible to make a final rating based on the classification or regression techniques. For the classification, we have tested Random Forest classifier and Logistic Regression for the algorithms that achieve the best performance. For the regression, we have tested Linear Regression, Partial Least Squares (PLS) and Extra Trees Regressor. That said, we can obtain a final evaluation of the selected strategy. To optimize the Pearson correlation score (the measure chosen by the organizers for the evaluation), we use PyCaret library (Moez, 2020) to compare all possible models (by using cross-validation with 10 folds).

For our English model, the various elements found allowed to have a good performance quickly (approximately 0.85 of Pearson correlation). The only concern was the strong imbalance of the training dataset that we needed to rebalance. The French model had a poor training dataset (only 72 pair examples). Thus, we selected a more efficient NER (Named Entity Recognition) Transformer model [5] than Spacy; and supplemented and balanced it to 200 pair examples with Spanish train texts to obtain a correct performance. We also focused on optimising the Turkish model in the same way with an efficient NER transformer[6]. We have not worked to optimize the German model which ought to have been much better.

5 Experimental Setup

We applied our scoring models to every pair of titles and content texts. For Polish language and the new languages observed in the evaluation dataset like Chinese, Russian and Italian, all texts were translated into English with deep Translator library[7] (Google Translate model) and then applied the English model. When there are two different languages, they are translated into English (for DE_EN, SP_EN, PL_EN and ZH_EN pairs), French (for FR_PL pairs), Spanish (for SP_IT pairs) or German (for DE_PL and DE_FR pairs). Finally, a compromise strategy was used between the score obtained in classification (integers) and in regression (value between 1 and 4 with selected rounding). Table 3 describes the confusion matrix after a test on a subset of train corpus.

Annotation score	1	2	3	4
1	**136**	27	4	0
2	50	**84**	35	16
3	6	42	**53**	89
4	0	20	33	**363**

Table 3: Example of Confusion Matrix obtained in English (test on the training dataset) after a Classification Random Forest Model.

6 Results and Analysis

Our final Pearson correlation on the test corpus is 0.706. A closer look at the results shows inconsistent performance scores for different languages. Table 4 shows the obtained results on the test corpus for each language pair. There are very good results for English and French (0.82 as Pearson correlation), good for Spanish (0.75), rather interesting for Turkish (0.74), disappointing for German (0.60, poorly optimized and badly managed model), and weak for Arabic (0.64). The final result for a specific language is generally consistent with the evaluations made previously on the subset (cross-validation) of training dataset.

For the translation part, we observe a fast drift according to the languages: the ZH_EN examples translated into English remain very accurate (0.79) but the ZH_ZH examples only obtain an average score of 0.69. We thus have a significant performance reduction in IT_IT (0.74) and SP_IT (0.61) compared to cases when English is used.

We found that most of the large deviations in evaluation (45 greater than 2 and 98 greater than 1.5) were related to scraping errors (blank or inconsistent text) where 2 and 1.5 are “Overall scores”. This derivation also resulted in an evaluation of 3 to 4 on our part for an actual close to 1. We did not find a reverse case where our score was close to 1 and the actual close to 4.

[5] https://huggingface.co/Jean-Baptiste/camembert-ner

[6] https://huggingface.co/savasy/bert-base-turkish-ner-cased

[7] https://deep-translator.readthedocs.io/en/latest/

We observe in conclusion that 71% scores of pairs were excellent (below 0.1), 84% were good (below 0.5) and 96% below 1. We believe that our English, French, Turkish and Spanish models are correct and could have been further optimized cleanly. We had technical difficulties in making a correct Arabic model. As for the German model, we did not work on it enough and it should have obtained a performance close to 0.80.

Language pair	Pearson correlation
EN_EN	**0.82**
DE_DE	0.60
SP_SP	0.75
PL_PL	0.55
TR_TR	0.74
AR_AR	0.64
RU_RU	0.67
ZH_ZH	0.69
FR_FR	**0.82**
DE_EN	0.74
SP_EN	0.79
IT_IT	0.74
PL_EN	0.73
ZH_EN	0.79
SP_IT	0.61
DE_FR	0.61
DE_PL	0.4
FR_PL	0.74
Global	**0.706**

Table 4: System results on the test corpus.

7 Conclusion

In this paper, we described our supervised semantic textual similarity system developed for the SemEval-2022 task 8 and the result of the corresponding run we submitted. Our system uses different features reflecting the similarity that can be obtained, for example, between shared key terms and named entities, or even topics by using zero-shot learning for text classification systems. In addition, we use geolocation for location entities and measure the semantic similarity through the use of lexical embeddings at the sentence level (text title) and paragraph level (text summarizer obtained automatically by using transformers).

Beyond the use of a supervised system to measure the degree of similarity between two given texts, we are in a context of documents that can come from different languages by processing both pairs of monolingual documents and pairs of cross-lingual documents. Since the test corpus may contain documents written in natural languages not processed during learning phase, our system is able to perform an automatic translation into a pivot language in order to project new documents into already known spaces.

Acknowledgments

We would like to thank Berger-Levrault for giving us the opportunity to participate in the 2022 edition of SemEval. Our thanks also go to the research and technological innovation department (DRIT) which is an important part of Berger-Levrault.

References

Agirre, E. et al. (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*. USA: Association for Computational Linguistics (SemEval '12), pp. 385–393. Available at: https://aclanthology.org/S12-1051.pdf.

Agirre, E. et al. (2013) '*SEM 2013 shared task: Semantic Textual Similarity', in *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity.* Atlanta, Georgia, USA: Association for Computational Linguistics, pp. 32–43. Available at: https://aclanthology.org/S13-1004.

Agirre, E. et al. (2016) 'SemEval-2016 Task 1: Semantic Textual Similarity, Monolingual and Cross-Lingual Evaluation', in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016).* San Diego, California: Association for Computational Linguistics, pp. 497–511. doi:10.18653/v1/S16-1081.

Boudin, F. (2016) 'pke: an open-source python-based keyphrase extraction toolkit', in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations.* Osaka, Japan, pp. 69–73. Available at: http://aclweb.org/anthology/C16-2015.

Cer, D. et al. (2017) 'SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Crosslingual Focused Evaluation', in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017).* Vancouver, Canada: *Association*

for Computational Linguistics, pp. 1–14. doi:10.18653/v1/S17-2001.

Chen, C. et al. (2021) 'bert2BERT: Towards Reusable Pretrained Language Models', arXiv:2110.07143 [cs] [Preprint]. Available at: http://arxiv.org/abs/2110.07143 (Accessed: 25 February 2022).

Conneau, A. and Kiela, D. (2018) 'SentEval: An Evaluation Toolkit for Universal Sentence Representations', *CoRR*, abs/1803.05449. Available at: http://arxiv.org/abs/1803.05449.

Demange, J. (2021) 'Four sentiments with FlauBERT', Hugging Face repository: https://huggingface.co/DemangeJeremy/4-sentiments-with-flaubert. [Preprint].

Eddine, M.K., Tixier, A.J.-P. and Vazirgiannis, M. (2021) 'BARThez: a Skilled Pretrained French Sequence-to-Sequence Model', arXiv:2010.12321 [cs] [Preprint]. Available at: http://arxiv.org/abs/2010.12321 (Accessed: 25 February 2022).

Guhr, O. et al. (2020) 'Training a Broad-Coverage German Sentiment Classification Model for Dialog Systems', in *Proceedings of The 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 1620–1625. Available at: https://www.aclweb.org/anthology/2020.lrec-1.202.

Henderson, J. et al. (2017) 'MITRE at SemEval-2017 Task 1: Simple semantic similarity', in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 185–190.

Hochreiter, S. and Schmidhuber, J. (1997) 'Long Short-Term Memory', Neural Computation, 9(8), pp. 1735–1780. doi:10.1162/neco.1997.9.8.1735.

Honnibal, M. et al. (2020) 'spaCy: Industrial-strength Natural Language Processing in Python', Zenodo [Preprint]. doi:10.5281/zenodo.1212303.

Jones, K.S. (1972) 'A statistical interpretation of term specificity and its application in retrieval', Journal of Documentation, 28(1), pp. 11–21. doi:10.1108/eb026526.

Leech, G., Rayson, P. and Wilson, A. (2001) Word Frequencies in Written and Spoken English based on the British National Corpus.

Lewis, M. et al. (2019) 'BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension', arXiv:1910.13461 [cs, stat] [Preprint]. Available at: http://arxiv.org/abs/1910.13461 (Accessed: 25 February 2022).

Mikolov, T. et al. (2013) 'Efficient Estimation of Word Representations in Vector Space', in *Proceedings of the International Conference on Learning Representations (ICLR)*, pp. 1–12.

Moez, A. (2020) PyCaret: An open source, low-code machine learning library in Python. Available at: https://www.pycaret.org.

Pérez, J.M., Giudici, J.C. and Luque, F. (2021) 'pysentimiento: A Python Toolkit for Sentiment Analysis and Social NLP tasks', arXiv:2106.09462 [cs] [Preprint]. Available at: http://arxiv.org/abs/2106.09462 (Accessed: 25 February 2022).

Qi, P. et al. (2020) 'Stanza: A Python Natural Language Processing Toolkit for Many Human Languages', CoRR, abs/2003.07082. Available at: https://arxiv.org/abs/2003.07082.

Řehůřek, R. and Sojka, P. (2010) Software Framework for Topic Modelling with Large Corpora. University of Malta. Available at: https://is.muni.cz/publication/884893/en/Software-Framework-for-Topic-Modelling-with-Large-Corpora/Rehurek-Sojka (Accessed: 24 February 2022).

Reimers, N. and Gurevych, I. (2019) 'Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks', in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics.* Available at: https://arxiv.org/abs/1908.10084.

Scialom, T. et al. (2020) 'MLSUM: The Multilingual Summarization Corpus'. In the Computing Research Repository (CoRR). Available at https://arxiv.org/abs/2004.14900.

Shao, Y. (2017) 'Hcti at semeval-2017 task 1: Use convolutional neural network to evaluate semantic textual similarity', in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 130–133.

Shleifer, S. and Rush, A.M. (2020) 'Pre-trained Summarization Distillation', CoRR, abs/2010.13002. Available at: https://arxiv.org/abs/2010.13002.

Sultan, M.A., Bethard, S. and Sumner, T. (2015) 'Dls@ cu: Sentence similarity from word alignment and semantic vector composition', in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pp. 148–153.

Tian, J. et al. (2017) 'ECNU at SemEval-2017 Task 1: Leverage Kernel-based Traditional NLP features and Neural Networks to Build a Universal Model for Multilingual and Cross-lingual Semantic Textual Similarity', in *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pp. 191–197.

Wallace, M. (2007) Jawbone Java WordNet API. Available at: https://sites.google.com/site/mfwallace/jawbone.

Wolf, T. et al. (2019) 'HuggingFace's Transformers: State-of-the-art Natural Language Processing', ArXiv, abs/1910.03771.

Wu, H. et al. (2017) 'BIT at SemEval-2017 Task 1: Using semantic information space to evaluate semantic textual similarity', in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 77–84.

Yin, W., Hay, J. and Roth, D. (2019) 'Benchmarking Zero-shot Text Classification: Datasets, Evaluation and Entailment Approach', CoRR, abs/1909.00161. Available at: http://arxiv.org/abs/1909.00161.

Zhang, J. et al. (2019) 'PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization'. In the Computing Research Repository (CoRR). Available at https://arxiv.org/abs/1912.08777.