Nikkei at SemEval-2022 Task 8: Exploring BERT-based Bi-Encoder Approach for Pairwise Multilingual News Article Similarity

Shotaro Ishihara **Hono Shirai**
Nikkei, Inc. Chiyoda, Tokyo, Japan
{shotaro.ishihara,hono.shirai}@nex.nikkei.com

Abstract

This paper describes our system in SemEval-2022 Task 8, where participants were required to predict the similarity of two multilingual news articles. In the task of pairwise sentence and document scoring, there are two main approaches: Cross-Encoder, which inputs pairs of texts into a single encoder, and Bi-Encoder, which encodes each input independently. The former method often achieves higher performance, but the latter gave us a better result in SemEval-2022 Task 8. This paper presents our exploration of BERT-based Bi-Encoder approach for this task, and there are several findings such as pretrained models, pooling methods, translation, data separation, and the number of tokens. The weighted average ensemble of the four models achieved the competitive result and ranked in the top 12.

1 Introduction

Measuring sentence and document similarity is a task that has been studied for many years in the field of natural language processing. One of the applications is to identify whether news articles address the same subject. If news articles can be properly clustered, they can be used for a wide range of purposes, such as recommendation and displaying related articles. SemEval-2022 Task 8 attempts to tackle this task with multilingual news articles (Chen et al., 2022).

Nowadays, it is common for this kind of tasks to use transformer-based models like BERT (Devlin et al., 2019). There are several research directions, including post-processing (Li et al., 2020; Wang and Kuo, 2020), unsupervised learning (Zhang et al., 2020; Tiyajamorn et al., 2021), and supervised learning (Reimers and Gurevych, 2019; Thakur et al., 2021; Feng et al., 2020; Jiang et al., 2022). Here, since a labeled dataset was provided, we decided to use a supervised learning approach.

It is standard practice to combine two sentences as input when dealing with pairwise similarity of sentences in a supervised learning approach with BERT (Lin et al., 2021; Reimers and Gurevych, 2019). In contrast, we chose an approach that embeds each sentence separately. Figure 1 shows these two approaches, named Cross-Encoder and Bi-Encoder to follow the previous research (Thakur et al., 2021).

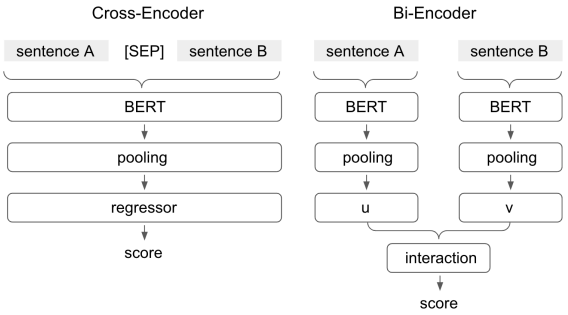

Figure 1: The architectures of Cross-Encoder and Bi-Encoder. The input of the Cross-Encoder architecture is two sentences joined by SEP, and through BERT and a pooling layer, the regressor outputs a score. In the Bi-Encoder architecture, each sentence is transformed by BERT and pooling layers, and the score is calculated through interaction of the two vectors.

This paper describes our system in SemEval-2022 Task 8. First, we explain the experimental results by adopting the Bi-Encoder architecture rather than Cross-Encoder. We also present the following research questions: 1) which pretrained model works well when dealing with multilingual news articles, 2) what kind of pooling method is proper for this task, 3) is it useful for translating the other language into English, and 4) is there some effect of data splitting and max length? Our code is available at https://github.com/upura/semeval2022-task8-multilingual-news-article-similarity.

2 Task Description

SemEval-2022 Task 8 provides a dataset that contains pairs of news articles. The dataset contains the information like the language and URL of each arti-

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1208 - 1214

Table 1: Language pairs in the training and evaluation dataset. There are eight language pairs in the training dataset, and an additional ten language pairs appear in the evaluation dataset.

language-language	traning	evaluation
English-English	1800	236
German-German	857	608
German-English	577	185
Spanish-Spanish	570	243
Turkish-Turkish	465	275
Polish-Polish	349	224
Arabic-Arabic	274	298
French-French	72	111
Russian-Russian		287
Chinese-Chinese		769
Spanish-English		496
Italian-Italian		411
Polish-English		64
Chinese-English		213
Spanish-Italian		320
German-French		116
German-Polish		35
French-Polish		11

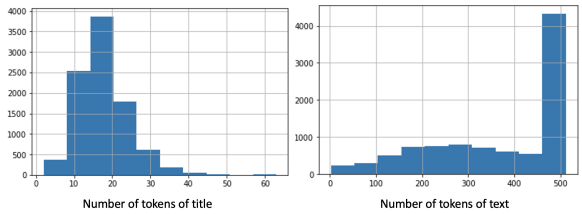

Figure 2: The histogram of the number of tokens. Left and right shows the number of tokens in the title and body text.

cle and the target score named `Overall`. Training dataset consists of 4,964 pairs of articles. Table 1 shows the number of each language pair. It is interesting that a large number of new language pairs appear in the evaluation dataset. It is inferred that building machine learning models using only the language pairs in the training dataset results in poor performance for these unknown language pairs. Therefore, it is essential to address multilingual datasets in some way. Information such as the title and body text of the article is available by scraping the data from the URL[1]. Figure 2 shows the histogram of the number of tokens[2]. Most titles are around 20-30 tokens, and the body text often has a maximum token length of 512. Each `Overall` score is calculated by averaging the annotators' scores.

The evaluation dataset, in which the labels are hidden from the participants, consists of 4,902 pairs. The ranking of the task is determined by Pearson's correlation between the labels in the evaluation dataset and the submitted predictions. Participants are allowed to submit their predictions five times per day, but none of the scores could be observed until the deadline. The leaderboard is also kept private until the end, so it is impossible to know the scores of the other teams.

[1] https://github.com/euagendas/semeval_8_2022_ia_downloader

[2] As a tokenizer, we used the pretrained BERT model named `bert-base-multilingual-uncased`.

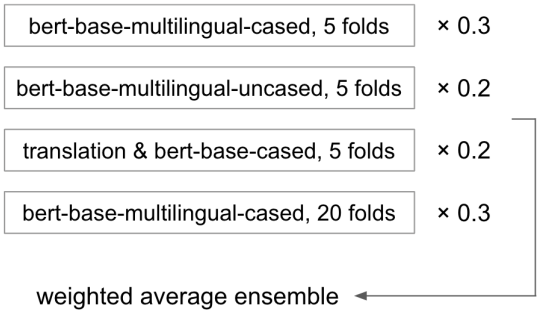

Figure 3: The overview of the developed system. Four neural networks output predictions and the final result is calculated by weighted average ensemble.

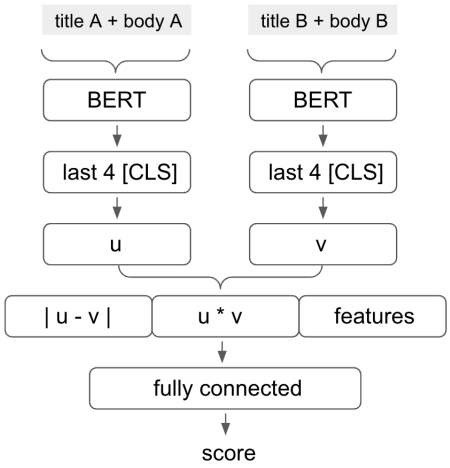

Figure 4: Base architecture of each neural network in the developed system.

3 System Overview

The developed system is outlined in Figure 3. The final prediction is calculated by a weighted average of the output of the four neural networks. Figure 4 illustrates the base architecture of each neural network. We use a Bi-Encoder approach where the two texts are entered into separate BERT. Each input is the combined title and body text of the article. As a pooling method, the representations of the last four `CLS` tokens are concatenated. `CLS` is the token to be attached to the beginning of an

input of BERT. At the end, the score is given based on the interaction of the two sentence vectors.

The rest of this section describes the key points of the system. First, we consider whether Cross-Encoder or Bi-Encoder should be used. Next, we address the perspectives of the research questions listed in Section 1: pretrained models, pooling methods, translation, data splitting, and maximum length. Finally, we explain the ensemble of models.

3.1 Cross-Encoder vs Bi-Encoder

In the Bi-Encoder architecture, each sentence is transformed into an embedding by BERT, and the sentence vectors are obtained by a pooling layer. Denote two vectors A and B, then their interactions are designed as `distance` and `angle` with reference to (Tai et al., 2015) in the following formula. Here, `distance` is calculated as an element-wise absolute error, and `angle` is calculated as an element-wise multiplication.

$$\texttt{distance} = |A - B|, \quad \texttt{angle} = A \otimes B$$

In addition to `distance` and `angle`, traditional features are also created and combined. We use Jaccard Index (Jaccard, 1912), Dice Index (Dice, 1945), and cosine similarity. These features may not work when dealing with pairs with different languages. However, when building a prediction model with LightGBM (Ke et al., 2017) using only these features, Pearson's correlation achieved 0.2989 in the evaluation data set. We believe that these features can contribute to some extent and combine them into the layer.

A comparison between Cross-Encoder and Bi-Encoder is reported in Section 4.1. The Cross-Encoder architecture for the comparison is shown in Figure 5. We use two BERT models, one with concatenated article headlines and one with concatenated body text. The pooling method and features are fixed to the same settings. The regressor is a simple fully connected layer.

3.2 Pretrained Models

We considered various pretrained BERT models of Hugging Face Transformers (Wolf et al., 2020). There are some multi-lingual pretrained models such as mBERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) available[3] and the following models are selected as candidates.

[3] https://huggingface.co/docs/transformers/multilingual

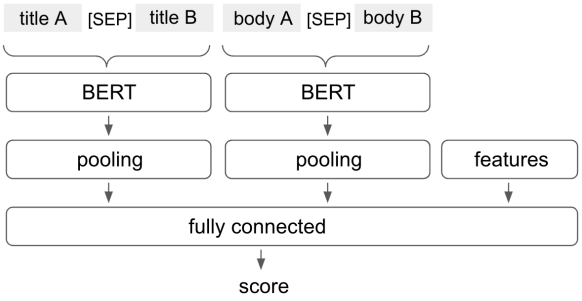

Figure 5: Cross-Encoder architecture for the comparison with Bi-Encoder architecture shown in Figure 4.

- `bert-base-multilingual-uncased`
- `bert-base-multilingual-cased`
- `xlm-roberta-base`

Since none of the models performed badly, we used all of them for the final submission as shown in Figure 3. A comparison among models is reported in Section 4.2.

3.3 Pooling Methods

There are several ways to extract the sentence vector from the output through BERT. One of the simplest ways is to use the embedding of the `CLS` token in the final layer, but some methods are proposed to use information from other layers as well. A number of experimental results have been reported (Reimers and Gurevych, 2019; Gao et al., 2021; Jiang et al., 2022; Conneau and Kiela, 2018), but we believe that the results depend largely on the individual task. Therefore, we consider the following four methods.

- `CLS`: Concatenate the last four representations of `CLS` token.
- `CNN`: Use the convolutional neural network (CNN) to extract sentence vectors.
- `LSTM`: Use the long short-term memory (LSTM) for extracting sentence vectors.
- `MAX`: Use max-pooling to extract sentence vectors.

On the basis of the results of our experiments, we adopted the first method. A comparison among the pooling methods is reported in Section 4.3.

3.4 Translation

One of the ways of dealing with multilingual datasets is the translation. Here we examine a method of translating all datasets into English and using pretrained models in English.

Table 2: Experimental results of Pearson's correlation for the validation and evaluation dataset. The columns named "pool" and "length" represent pooling methods and max length respectively.

id	architecture	model	pool	folds	length	validation	evaluation
0	Cross-Encoder	`bert-base-multilingual-cased`	`CLS`	5	512	0.7045	0.6188
1	Bi-Encoder	`bert-base-multilingual-cased`	`CLS`	5	512	0.7688	0.6922
2	Bi-Encoder	`bert-base-multilingual-uncased`	`CLS`	5	512	0.7627	0.6940
3	Bi-Encoder	`xlm-roberta-base`	`CLS`	5	512	0.7118	0.6153
4	Bi-Encoder	`bert-base-multilingual-cased`	`CNN`	5	512	0.4892	0.3269
5	Bi-Encoder	`bert-base-multilingual-cased`	`LSTM`	5	512	0.4979	0.3271
6	Bi-Encoder	`bert-base-multilingual-cased`	`MAX`	5	512	0.7221	0.6313
7	Bi-Encoder	translation with `bert-base-uncased`	`CLS`	5	512	0.7505	0.6748
8	Bi-Encoder	`bert-base-multilingual-cased`	`CLS`	20	512	0.7853	0.7107
9	Bi-Encoder	`bert-base-multilingual-cased`	`CLS`	5	256	0.7427	0.6616
10	Bi-Encoder	`bert-base-multilingual-cased`	`CLS`	5	128	0.7137	0.6355
11	Bi-Encoder	`bert-base-multilingual-cased`	`CLS`	5	64	0.6744	0.5782
12		weighted average ensemble				**0.7902**	**0.7425**

We use Googletrans[4] for the translation, and `bert-base-cased` as a pretrained model. In conclusion, the translation approach did not improve the performance of the multilingual models. A comparison result is shown in Section 4.4.

3.5 Data Splitting and Max Length

We also investigate the effect of data splitting and max length. The number of data partitions in cross validation (Blum et al., 1999) affects the number of available training samples. When the size of the training dataset is not very large, as in this task, it may affect the performance. The setting of the max length based on the distribution of the token size introduced in Section 2 is also an adjustable element. In general, news articles contain important information early in the article, so there is a possibility that a smaller max length works well.

For data splitting, it was suggested through the experiment that the larger the number of splits, that is, the larger the training dataset, the higher the performance. We set the max length to 512, because, in contrast to the hypothesis, the smaller max length led to the poor performance. Both comparison results are shown in Section 4.5.

3.6 Weighted Average Ensemble

In the weighted average of the models, the performance on the validation dataset was taken into account to determine the models to be used and their weights. In the final submission, we used the average of all models obtained in the cross validation process. In case of folds set 5, five models were generated. This means that $(5 + 5 + 5 + 20)$ models were used in Figure 3. The improvement through the ensemble is reported in Section 4.6.

[4] https://github.com/ssut/py-googletrans

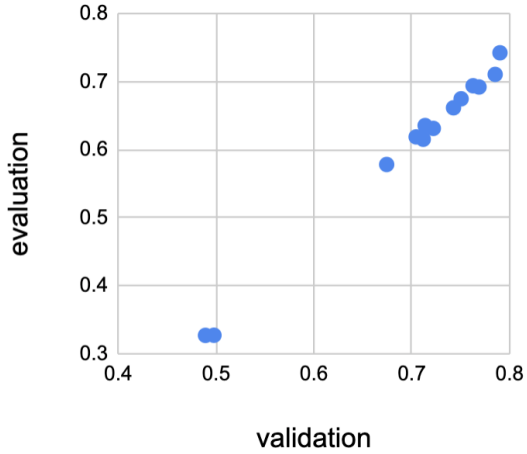

Figure 6: Scatter plot of Pearson's correlation for the validation and evaluation dataset. It can be observed that the scores are correlated.

4 Results

This section reports the experimental results that facilitated the design of the system described in the previous section. Table 2 lists the scores of Pearson's correlation for the validation and evaluation dataset.

The validation dataset is extracted from the training dataset. The column named "folds" shows the number of folds in cross validation. That is, in case of folds set 5, 20 % of the training dataset is removed for the validation. Each model is trained for seven epochs, and the scores with the best performance on the validation dataset are reported. In our experimental setting, the performance for the validation dataset converged after about five training epochs.

The scatter plots of the performance for the vali-

dation and evaluation datasets are shown in Figure 6, and it can be observed that they are correlated. It is suggested that the improved performance on the validation dataset is also useful on the evaluation dataset, and that the validation framework works well.

The rest of this section describes the result in detail from the same perspective as in the previous section. The models in Table 2 are referred to as experiments 0-12 respectively, and their performances are compared for the discussions.

4.1 Cross-Encoder vs Bi-Encoder

Comparing the results of experiments 0 and 1, we see that the architecture of Bi-Encoder worked well rather than Cross-Encoder. There is a large difference of more than 0.06 in Pearson's correlation. The results are only for this task and our experimental setup and should not be overly generalized. However, it is an interesting case study, as the results contrast with the use of Cross-Encoder in some of the previous studies presented in Section 1. The results suggest that it is important to try both Cross-Encoder and Bi-Encoder in the search for high performance.

4.2 Pretrained Models

The results of experiments 1-3 show the performance of each pretrained model. The model with `bert-base-multilingual-uncased` and `bert-base-multilingual-cased` performed better than `xlm-roberta-base`.

4.3 Pooling Methods

Seeing the results of experiments 1 and 4-6, we can say that the best pooling method in this task is `CLS`. It outperformed the other three methods.

4.4 Translation

The results of experiment 7 show that the translation approach did not improve the performance from experiment 1.

4.5 Data Splitting and Max Length

In experiment 8, the number of folds was changed from 5 to 20. This was not an exact comparison since the validation dataset was changed, but there was 0.03 improvement in Pearson's correlation from experiment 1. For max length, we examined values from 64 to 256 in experiments 9-11. It was observed that the performance was getting worse as the max length was decreased.

Table 3: Correlation between the experiments used in the final submission (1, 2, 7, and 8).

id	2	7	8
1	0.9156	0.8812	0.9348
2	-	0.8752	0.9105
7	-	-	0.8730

Table 4: Median of absolute error and the number of samples for the evaluation dataset for each language pair. The symbol ✓ means the language pair is included in the training dataset.

language-langage	median	samples	training
German-English	**0.2971**	185	✓
Chinese-English	0.4092	213	
French-French	0.4213	111	✓
Spanish-Italian	0.4251	320	
Polish-English	0.4286	64	
English-English	0.4662	236	✓
Spanish-Spanish	0.5223	243	✓
Spanish-English	0.5239	496	
Polish-Polish	0.5256	224	✓
Italian-Italian	0.5444	411	
Chinese-Chinese	0.5450	769	
Russian-Russian	0.5722	287	
Turkish-Turkish	0.5789	275	✓
French-Polish	0.5855	11	
German-German	0.6030	608	✓
Arabic-Arabic	0.6765	298	✓
German-French	0.6905	116	
German-Polish	**0.7751**	35	

4.6 Weighted Average Ensemble

The results of experiment 12 show that the weighted average of the models boosted the performance. It is important to highlight that when comparing experiment 8 and 12, the performance improved by only 0.005 on the validation dataset, but by more than 0.03 on the evaluation dataset. Table 3 describes the correlation between the experiments used in the final submission (1, 2, 7, and 8). We can see that there is a high similarity between experiments 1 and 8, where the only difference is the number of folds. It is also observed that the translation approach contributes to the diversity, because its correlation is low. This was our submission for SemEval-2022 task 8. The score 0.7425 for the evaluation dataset ranked 12th out of 32 teams.

4.7 Error Analysis

Here we describe the analysis results using labels of the evaluation dataset. First, a comparison of each pair of languages shows the performance differences observed in Table 4. The median absolute error for German-English is 0.2971, whereas German-Polish is 0.7751. German-German per-

forms relatively poorly, and their proportion in the evaluation dataset is high as shown in Table 1. Focusing on these language pairs may improve the overall performance of the system. It is noteworthy that even language pairs that are not part of the training dataset, such as Chinese-English, show excellent performance. It can be suggested that models pretrained in multilingual languages worked well.

Next, we identified the problem in obtaining the article body text by checking extremely incorrectly predicted samples. For example, consider the sample with `pair_id` equals `1512411298_1512618793` where the system predicted 3.555 and the correct answer was 1.000. We checked the body text of the article `1512618793` and found that the extracted text by the script was different from the actual body text for some reasons such as the page layout like advertisements or related articles.

5 Conclusion

This paper presented our exploration of BERT-based Bi-Encoder approach for SemEval-2022 task 8. The experiment showed that Bi-Encoder architecture worked better than Cross-Encoder. There are several findings, such as pretrained models, pooling methods, translation, data separation, and the number of tokens. The exploration of these different variants led to the creation of several diverse models. Finally, a weighted average ensemble of the four models achieved the competitive result.

Acknowledgements

We would like to thank Takanori Hayashi for the discussions on the Quora Question Pairs[5], which is a related competition held in Kaggle. We also thank anonymous reviewers for their many insightful comments and suggestions.

References

Avrim Blum, Adam Kalai, and John Langford. 1999. Beating the hold-out: bounds for k-fold and progressive cross-validation. In *Proceedings of the twelfth annual conference on Computational learning theory*, COLT '99, pages 203–208, New York, NY, USA. Association for Computing Machinery.

Xi Chen, Ali Zeynali, Chico Q. Camargo, Fabian Flöck, Devin Gaffney, Przemyslaw A. Grabowicz, Scott A. Hale, David Jurgens, and Mattia Samory. 2022. SemEval-2022 Task 8: Multilingual news article similarity. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Lee R Dice. 1945. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Paul Jaccard. 1912. The distribution of the flora in the alpine zone.1. *New Phytol.*, 11(2):37–50.

Ting Jiang, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Liangjie Zhang, and Qi Zhang. 2022. PromptBERT: Improving BERT sentence embeddings with prompts.

Guolin Ke, Qi Meng, Thomas Finely, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. 2017. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30 (NIP 2017)*.

[5] https://www.kaggle.com/c/quora-question-pairs

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130, Online. Association for Computational Linguistics.

Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2021. Pretrained transformers for text ranking: BERT and beyond. *Synthesis Lectures on Human Language Technologies*, 14(4):1–325.

Nils Reimers and Iryna Gurevych. 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.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from Tree-Structured long Short-Term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566, Beijing, China. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021. Augmented SBERT: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 296–310, Online. Association for Computational Linguistics.

Nattapong Tiyajamorn, Tomoyuki Kajiwara, Yuki Arase, and Makoto Onizuka. 2021. Language-agnostic representation from multilingual sentence encoders for cross-lingual similarity estimation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7764–7774, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bin Wang and C-C Jay Kuo. 2020. SBERT-WK: A sentence embedding method by dissecting BERT-Based word models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2146–2157.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yan Zhang, Ruidan He, Zuozhu Liu, Kwan Hui Lim, and Lidong Bing. 2020. An unsupervised sentence embedding method by mutual information maximization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1601–1610, Online. Association for Computational Linguistics.