

Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference “Dialogue 2019”

Moscow, May 29—June 1, 2019

CROSS-LANGUAGE TEXT ALIGNMENT FOR PLAGIARISM DETECTION BASED ON CONTEXTUAL AND CONTEXT-FREE MODELS

Zubarev D. V.¹, Sochenkov I. V.^{2,1}

¹Federal Research Center ‘Computer Science and Control’
of Russian Academy of Sciences, Moscow, Russia

²Skolkovo Institute of Science and Technology, Moscow, Russia

In this paper, we present a dataset for cross-language (Russian-English) text alignment subtask of plagiarism detection. We compare different models for detecting translated plagiarism. One is based on different textual similarity scores, which exploit word embeddings. Another model extends the previous one with the features obtained via neural machine translation. The last model is built on top of pre-trained language representation (Bert) via fine-tuning for our task. The Bert model shows great performance and outperforms other models. However, it requires much more computation resources than simpler models. Therefore, it seems reasonable to use both context-free models and contextual models together in modern plagiarism detection systems.

Keywords: cross-language text alignment, cross-language plagiarism detection, word embeddings, neural machine translation, Bert

ОБНАРУЖЕНИЕ ФРАГМЕНТОВ В ЗАДАЧЕ КРОСС-ЯЗЫКОВОГО ПОИСКА ЗАИМСТВОВАНИЙ С ИСПОЛЬЗОВАНИЕМ ЭМБЕДДИНГОВ СЛОВ

Зубарев Д. В.³, Соченков И. В.^{4,3}

³Федеральный исследовательский центр «Информатика
и управление» Российской академии наук, Москва, Россия

⁴Сколковский институт науки и технологий, Москва, Россия

1. Introduction

Plagiarism is a serious and known problem in education and research, especially in developing countries like Russia. The availability of the huge amount of texts on the Web and free machine translation services makes it easier to create an “original” study.

Systems for detecting plagiarism are very common now, and they are capable of detecting monolingual plagiarism even with obfuscations.

However, detecting translated plagiarism is a very challenging task, and there are no such tools on the Russian market of plagiarism detection systems. However, recent studies showed that cross-language text reuse is common in research and education [Bahteev et al., 2018]. Therefore, it is important for the state-of-the-art system to detect cross-language plagiarism too.

Commonly plagiarism detection is divided into two stages: source retrieval and text alignment.

- On the source retrieval stage for a given suspicious document, we need to find all sources of probable text reuse in a large collection of texts. For this task, a source is a whole text, without details of what parts of this document were plagiarized. Typically we get a large set of documents (around 1,000 or more) as a result of this stage. Those documents are called “candidates”.
- On the text alignment stage: we compare suspicious document to each candidate to detect all reused fragments, and identify its boundaries.

For a review of the state-of-the-art monolingual methods of plagiarism detection see [Stamatatos et al., 2015]. The same stages are valid for cross-language plagiarism detection too. In this work, we study only the second task. It means that we operate with pairs of a suspicious document and a candidate. To simplify this task further, we consider only pairs of sentences, as if we split the suspicious document and a source into sentences and combine all of them (Cartesian product of sentences in the source and target documents). Then our task is to identify the pairs of sentences that are translated from English to Russian.

2. Related work

The overview of different approaches for solving this task is presented in [Potthast et al., 2011]. Also, there made an evaluation and a detailed comparison of some featured methods. In [Ferrero et al., 2017] is described a method for cross-language similarity detection based on the distributed representation of words (word embeddings). Experiments were conducted on English-French corpus. In [Kutuzov et al., 2016] is described a training of word embeddings on monolingual comparable corpora and learning the optimal linear transformation of vectors from one language to another (there were used Russian and Ukrainian academic texts). Also there were discussed usage of those embeddings in source retrieval and text alignment subtasks.

SemEval 2017 [Cer et al., 2017] was focused on multilingual and cross-lingual semantic textual similarity (STS) of sentence pairs. Unfortunately, the Russian language was not presented in any task. Most participants used common machine translating systems to transform the task of cross-lingual STS to monolingual task.

However, the reverse translation of the suspicious document is not enough (and the translation of potential source as well), since machine translation generates multiple variants, and unscrupulous authors modify reused fragments.

Another task similar to plagiarism detection is the cross-lingual language understanding (XLU) [Conneau et al., 2018] that is derived from Multi-Genre Natural Language Inference (MNLI) Corpus. Only the development and test sets of MNLI were translated, and that corpus is mostly dedicated for the zero-shot models.

3. Cross-language sentence similarity scores

In this section, we describe how we measure the similarity between two sentences. For this, we trained cross-language word embeddings based on a large parallel corpus. We used these embeddings to estimate two different similarity scores: one is based on sentence embeddings, and other is calculated after the substitution of all words with the most similar ones in the other language. We trained the neural machine translation system to obtain an additional similarity score by comparing various N-grams of sentences. Finally, we used these similarity scores to construct a dataset for cross-language text alignment task.

3.1. Preprocessing

On a preprocessing stage, we split each sentence into tokens and lemmatize tokens of texts using methods described in [Osipov et al., 2013]. In addition, we removed words with non-important part of speech: conjunction, pronoun, preposition, etc., and common stop-words (be, являться). The result of preprocessing is the source to learn embeddings.

3.2. Cross-language word embeddings

We train cross-language word embeddings for a Russian-English pair on parallel corpora available on the Opus site¹, namely:

- OpenSubtitles2018
- News Commentary
- TED Talks 2013
- MultiUN
- ParaCrawl
- Wikipedia

We extended this dataset with sentences from the Yandex Parallel corpus² [Antonova et al., 2011]. We used texts of various genres since we wanted to have a large vocabulary. The final goal of training word embeddings was to be able to find most common translations of Russian word/phrases to English. Generally, we were

¹ Opus: the open parallel corpus, <http://opus.nlpl.eu/>

² Англо-русский параллельный корпус: <https://translate.yandex.ru/corpus?lang=en>

interested in multiple translations from different genres since we do not know in advance what text will be checked (e.g., for word “chick” there were translations *девчонка* and *цыпленок*). All parallel sentences were preprocessed. After that, all pairs that had a difference in the size of more than five words were filtered out.

We extended our model for cross-language word embeddings adding phrases representing some concepts/terms. Thus, we used parallel concepts from Wikidata³, which consist of multiple words, as a single phrase (e.g., *military_law*) when training embeddings. Thus we can learn the similarity between words and phrases (sustainable word combinations) that are represented differently in two languages: such as “зубная щетка” and toothbrush. We excluded concepts that are too rare for our corpus (<10 occurrences) and those concepts that had an irrelevant category (film/song/books titles) for our purposes.

Finally, we assembled a corpus of more than 44 million sentences (for each language). The dictionary size was around 507,000 words/phrases.

We applied the method proposed in [Vulić et al., 2015], designed for learning bilingual word embeddings from non-parallel document-aligned data, but it can be used for learning on parallel corpora too. According to the method, two comparable texts in different languages are combined into one pseudo text. In our case, we interleaved two parallel sentences, e.g. if we were given two sentences: “Мама мыла раму” and “Mother washed the frame”, the result of their merging is: “мама mother мыла washed раму the frame”. Since we removed auxiliary words from sentences, we assumed that corresponding Russian and English words were in the same context window. It would not be the case if there is a different words order, and it can be somehow fixed with larger context window, but in our experiments and in [Upadhyay et al., 2016] the context window of 5 words was enough. After that, the word2vec skip-gram model [Mikolov et al., 2013] is used on the resulting corpus of merged texts. We used gensim word2vec implementation with those parameters: dimensionality of embeddings was 300, a window size of 5 words, the minimal corpus frequency of 10, negative sampling with 10 samples, no down-sampling, 15 iterations over the corpus.

3.3. Sentence embeddings

Sentence embedding in our approach is defined by averaging embeddings of its words and phrases.

$$Emb(s) = \frac{1}{|s|} \sum_{w \in s} Emb(w)$$

We tried various approaches to obtain sentence embeddings from the word embeddings [Rücklé et al., 2018], but their performance was slightly worse than an average of word vectors.

We chose cosine similarity as a sentences similarity score.

³ Wikidata: <https://www.wikidata.org/wiki/Q321>

3.4. Words substitution

The natural approach to measure the cross-language similarity between sentences is to substitute Russian words from the sentence with the N_s most similar English words. The similarity is determined based on cosine similarity between embeddings of words/phrases. On the first step, we tried to replace the whole phrases in the sentence if they were found in the dictionary. On the second step, we replaced single words with their most similar English analogues. We precalculated top N_s of similar words for each Russian word/phrase in our dictionary to increase computational performance.

Let S_r and S_e denote the Russian and English sentences respectively. For the simplicity, we will consider the case when we map words from Russian sentence S_r to English words.

$Si(w_1, w_2)$ is a cosine similarity score between two words w_1 and w_2 , $Top(w, l)$ is a function that for word w returns N_s most similar words of language l .

We then define for each English word w_e a set of Russian words that have w_e in their top of similar words. $(w_e) = \{w_r \mid w_r \in S_r \wedge w_e \in Top(w_r, 'en')\}$. Then non-normalised similarity score is calculated by the following formula.

$$NSubst(S_r, S_e) = \sum_{w \in S_e} \frac{\sum_{w_r \in M(w)} Sim(w, w_r) \cdot (v(w) + v(w_r))}{|M(w)|}$$

where (w) is the number of words in a phrase if w is a phrase and 1 otherwise.

A normalized variant is presented below:

$$N_r = \sum_{w \in S} v(w) \quad N_e = \sum_{w \in S} v(w)$$

$$Subst(S_r, S_e) = \frac{NSubst(S_r, S_e)}{N_r + N_e}$$

3.5. Neural machine translation (NMT)

We used OpenNMT-py⁴ library to train a machine translation (MT) system as an additional criterion to estimate the pairwise similarity between the sentences. For this purpose, we employed a subset of the parallel corpus, which was used for learning embeddings. We adjusted corpus in the following ways:

- sentences from OpenSubtitles2018 were removed since they are generally too short after stop-words removing;
- the maximum difference in length between sentences should have been no more than 2;
- some sentences from ParaCrawl and MultiUN were dropped to reduce the time of training.

Thus we got about 7 millions of parallel sentences.

⁴ An open source neural machine translation system: <http://opennmt.net/>. Pytorch implementation <https://github.com/OpenNMT/OpenNMT-py> v0.5

We did not use pre-trained embeddings for training MT system. We used default architecture RNN encoder-decoder with attention and mostly default settings provided by OpenNMT, with Russian and English dictionary sizes: 95k and 80k respectively, with 64 batch size and with 600k train steps.

We used the trained model for translating Russian sentences and measuring Jaccard similarity between sentence pairs without any further preprocessing.

$$NMT(S_{rt}, S_e) = \frac{|S_{rt} \cap S_e|}{|S_{rt} \cup S_e|},$$

where S_{rt} is a translated Russian sentence, and S_e is an English sentence. We measured similarity on 1-grams (NMT) and on 2-grams (NMT2).

4. Dataset

To address our cross-language text alignment task we created a dataset for Russian-English plagiarism detection. 16k sentence pairs were taken from Yandex parallel corpus (those sentences were not used for learning word embeddings), and 4k sentences were manually written by students. Students should have searched sources in English using common Web search engines. After that, they must have translated them. They were allowed to use common translation tools like Google Translate or Yandex Translate, but the adjustment of the translated text was required to produce correct Russian text. Some percent of sentences should have been translated without any automatic tools. Those pairs are positive examples of plagiarized pairs of sentences.

Modern methods for training language models and learning word embeddings need negative sampling to get meaningful results. To obtain negative samples, we compared each Russian sentence to all English sentences (except one that is a translation) using various sentence similarity scores (described in the previous sections). The most similar sentences were selected for each Russian sentence as negative examples. We controlled that for each Russian sentence there should be distinct English translations, because the same English translation may be the most similar one by various scores. However, the same English translation could have been selected for different Russian sentences. The rationale behind this approach is that random negative sampling would not have any reasonable effect due to involving quite different sentences into the training set. However, the desired behavior of the model is to separate plagiarized sentences from others, which contain some similar lexis but have different semantics. To achieve this, we generated two corpora with different amount of negative samples per each positive sentence pair (containing source and plagiarized sentences):

- Negative-1: One negative example was selected randomly from the most similar sentences. According to [Belyy et al., 2018] one negative sample is enough for the text alignment task. We used this dataset for training and tuning models.
- Negative-4: 4 negative examples were selected (one most similar sentence for each used similarity score). By its nature, this task requires a comparison of many pairs of sentences. Moreover, most of these pairs are negative examples. Therefore, we used this dataset for testing purposes, to check how models handle a larger amount of negative examples.

The obtained corpora characteristics are presented in the table (sizes are in sentences).

Table 1. Corpora characteristics

	training set size	hold-out set size	test set size
Negative-1	28,320	4,000	7,998
Negative-4	65,962	9,266	18,613

These corpora are freely available⁵.

5. Models

We used previously described sentence similarity scores as simple baselines. If a score for some pair was greater than some threshold, this pair was considered to be a case of plagiarism. The process of tuning thresholds is described in the next section. As a more complex model, we used a classifier with similarity scores as features. In addition, we tried to use some pre-trained language representations for this task.

5.1. Similarity scores

We tuned thresholds and parameters for each score independently. We selected parameters using grid search to maximize F1 on the hold-out Negative-1 dataset. The obtained values of parameters are presented in the table.

Table 2. Thresholds and parameters values

	T_{cos}	T_{subst}	N_s	T_{nmt}	T_{nmt2}
Negative-1	0.7	0.25	3	0.2	0.06

5.2. Logistic Regression Classifier

We trained two Logistic Regression (LR) classifiers (with L2 regularization and $C=1.0$) that used the previously described sentence similarity scores as features. The first classifier (LR-1) used all features, whereas the second (LR-2) used only sentence embeddings similarity score and words substitution similarity score as features.

⁵ <http://nlp.isa.ru/ru-en-text-align-corp>

5.3. Bert

We fine-tuned Bert [Devlin et al., 2018] Multilingual model⁶ for our classification task. We considered a simple linear layer for the sentence pair classification on top of the pooled output of Bert. Pooling is done by simply taking the hidden state corresponding to the first token ('CLS' in this case) in the input sequence. The same pooling was used to train Bert originally to perform Next-Sentence-Prediction task. The input sequence consists of two sentences separated with the special token 'SEP'. The task is the same as paraphrase detection MRPC⁷. The main difference that the multilingual model is used and the two sentences are in a different language. Bert was trained on Negative-1 training data with the following parameters: `max_seq_length - 128`, `train_batch_size - 32`.

5.4. Laser sentence embeddings

Another approach to get sentence embeddings is called Language-Agnostic Sentence Representations (LASER). LASER [Artetxe et al., 2018] provides a BiLSTM encoder, which was trained on 93 languages. The encoder was coupled with an auxiliary decoder and trained on publicly available parallel corpora.

We obtained sentence embeddings from the encoder via max pooling of the last layer outputs. We apply cosine similarity on corresponding sentence embeddings of each sentence pair to determine whether this is a plagiarism case. The threshold was tuned on Negative-1 hold-out set to maximize F1 score. The value of the tuned threshold was 0.72.

6. Evaluation Results

In this section, we present evaluation results, obtained on test sets of two corpora negative-1 and negative-4, for each score independently and for the classifiers.

The results are in the **table 3**.

Bert outperforms the classifiers on both datasets. Bert's performance drops while testing on the larger corpus, but this decrease was lesser than for the classifiers.

The classifiers outperform all standalone similarity measures. The best F1 score has the classifier trained on all features (LR-1). Its performance is decreased on the 0.09 when moving to the larger data set (Negative-4), whereas the performance of the best result among similarity scores (NMT) dropped on 0.13 when comparing F1 scores. It is clear that the higher number of negative examples is the lesser precision, although recall stays the same.

⁶ BERT-Base, Multilingual Cased: https://storage.googleapis.com/bert_models/2018_11_23/multi_cased_L-12_H-768_A-12.zip.

⁷ Microsoft Research Paraphrase Corpus: <https://www.microsoft.com/en-us/download/details.aspx?id=52398>.

Table 3. Results on the test set

	Negative-1			Negative-4		
	Precision	Recall	F1	Precision	Recall	F1
Sentence embeddings	0.75	0.77	0.76	0.45	0.77	0.57
Words substitution	0.84	0.76	0.80	0.6	0.76	0.66
NMT	0.85	0.8	0.82	0.61	0.8	0.69
NMT2	0.83	0.64	0.72	0.54	0.64	0.58
LR-2 (2 features)	0.87	0.8	0.83	0.64	0.8	0.71
LR-1 (all features)	<u>0.91</u>	0.8	0.85	<u>0.73</u>	0.8	0.76
Laser	0.9	<u>0.89</u>	<u>0.89</u>	0.7	<u>0.89</u>	<u>0.78</u>
Bert	0.96	0.93	0.95	0.88	0.93	0.9

It is debatable what score should be used for evaluation of the PD system. There are multiple scenarios of usage for such systems, which requires different qualities of the system. Precision is more important when searching for literal plagiarism. The PD system may be used as a source of evidence during some legal procedures. Few plagiarized fragments are enough in this case, and it is also important to minimize manual checking of not plagiarized fragments. High recall is vital for researchers that scrupulously study origins of some piece of work or for checking students' essays. One may argue that it is easy to obtain high precision and recall scores independently: for the former, it is enough to find one literal plagiarism case, for the latter one can return all fragments marked as plagiarism. We think that the balance of precision and recall is quite important for this task and the PD system. It is required to find all plagiarism cases, and in the same time not to clutter the results with many false positives, which can make the inspection of a final report quite challenging for a human expert. We think that the real world PD system should optimize F1 score if there are no specific requirements by default, and should be tunable for various specific use-cases, which require either higher precision or higher recall.

Also, the PD system should be quite efficient to be able to cope with a large amount of checks in a small amount of time (typical use-case scenario during exam session). The computation times of all models are presented in the table.

Table 4. Computation time of models in seconds

	Negative-1	Negative-4
Sentence embeddings	2.89	4.02
Words substitution	2.63	3.3
NMT on GPU	34.15	34.31
NMT on CPU	240.13	240.29
LR-2 (2 features)	5.53	7.34
LR-1 (all features)	39.68	41.65
LASER	7.63	11.04
Bert	91.95	197.45

Bert is quite slow: it requires about 90 seconds to classify all pairs in Negative-1 test set, using one GPU GTX 1080. Bert is more than two times slower than the classifier and orders of magnitude slower than the simple classifier (LR-2). The LASER embeddings show good balance between F1 and computational cost. Considering that we did not pre-learn English embeddings, its computation time may be reduced further.

As a side note, the LR-1 classifier trained on Negative-1 and Negative-4 showed roughly the same performance, when were tuned on Negative-4 hold-out set.

Table 5. Results on test set

	Recall	Precision	F1
LR-1 trained on Negative-1	0.75	0.81	0.779
LR-1 trained on Negative-4	0.75	0.82	0.782

To achieve these results, we tuned classifier's margin b on the hold-out Negative-4 set $S = (x, y)$ [Belyy et al., 2018].

$$b = \operatorname{argmax}_b F1(y, [cls(x) - b > 0])$$

Results of tuning:

Table 6. Classifier's margin values after tuning

	b
LR-1 trained on Negative-1	0.6
LR-1 trained on Negative-4	0.33

The tuning of the classifier on the Negative-1 hold-out set yielded $b == 0.5$, which is not a surprise since the set was balanced.

7. Conclusion

In this paper, we presented a dataset for cross-language (Russian-English) text alignment task as an alternative to existing datasets. We compared different models for detecting translated plagiarism. One is based on various textual similarity scores that exploit word embeddings and neural machine translation. Another model is built on top of pre-trained language representation via fine-tuning for our task. The Bert model showed great performance and outperformed our custom model. However, in the production usage, it is common to process a hundred millions of sentence pairs only for one suspicious document. It is not practical to employ Bert model for such a computationally expensive task. It is reasonable to filter out many negative pairs with a more efficient method. As such a method, it is possible to use our classifier with reduced feature space: only with sentence embeddings and word substitution measures. This classifier, tuned to maximize recall, can significantly decrease the load on the more complex processing downstream. Also, it seems promising to employ cross-language sentence embeddings (LASER) for the preprocessing step. Since the

embeddings for the source and suspicious sentences could be built only once. After that, it is possible to use efficient nearest-neighbor search algorithms [Johnson et al., 2017] to find similar vectors.

Extending vocabulary is another important issue, which should be considered for any real cross-language plagiarism detection system. Most available parallel corpora contain common lexis. However, plagiarism detection should also work for scientific papers, patents, etc. containing a lot of special lexis and terms. One of the possible solutions is to create parallel corpora from comparable corpora [Zweigenbaum et al., 2018] using the system for translated plagiarism detection and extend vocabulary with new parallel data. It requires additional study.

This work was dedicated to text alignment subtask of plagiarism detection task. The source retrieval subtask is the first and crucial step in plagiarism detection. We plan to address this problem in future research, using cross-language word embeddings described in this paper.

8. Acknowledgments

The reported study was funded by RFBR according to the research projects № 18-37-20017 & № 18-29-03187.

References

1. Antonova, A., & Misyurev, A. (2011, June). Building a web-based parallel corpus and filtering out machine-translated text. In Proceedings of the 4th Workshop on Building and Using Comparable Corpora: Comparable Corpora and the Web (pp. 136–144). Association for Computational Linguistics.
2. Artetxe, M., & Schwenk, H. (2018). Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. arXiv preprint arXiv:1812.10464.
3. Bahteev, O. Y. (2018). Cross-language plagiarism detection in large collections of scientific documents: http://www.machinelearning.ru/wiki/images/f/ff/Sample_final.pdf.
4. Belyy A. V., Dubova M. A. (2018). Framework for Russian plagiarism detection using sentence embedding similarity and negative sampling. In Proceedings of the International Conference “Dialogue 2018” (pp. 96–110).
5. Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., & Specia, L. (2017). Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. arXiv preprint arXiv:1708.00055.
6. Conneau, A., Lample, G., Rinott, R., Williams, A., Bowman, S. R., Schwenk, H., & Stoyanov, V. (2018). Xnli: Evaluating cross-lingual sentence representations. arXiv preprint arXiv:1809.05053.
7. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

8. Ferrero, J., Agnes, F., Besacier, L., & Schwab, D. (2017). Using Word Embedding for Cross-Language Plagiarism Detection. arXiv preprint arXiv:1702.03082.
9. Johnson, J., Douze, M., & Jégou, H. (2017). Billion-scale similarity search with GPUs. arXiv preprint arXiv:1702.08734.
10. Kutuzov et al. (2016). Clustering Comparable Corpora of Russian and Ukrainian Academic Texts: Word Embeddings and Semantic Fingerprints. Proceedings of the Ninth Workshop on Building and Using Comparable Corpora (LREC 2016, Portorož, Slovenia, May 23, 2016). pp. 3–10
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111–3119).
12. Osipov G., Smirnov I., Tikhomirov I., Shelmanov A. (2013), Relational-situational method for intelligent search and analysis of scientific publications, Proceedings of the Integrating IR Technologies for Professional Search Workshop, pp. 57–64.
13. Potthast, M., Barrón-Cedeño, A., Stein, B., & Rosso, P. (2011). Cross-language plagiarism detection. Language Resources and Evaluation, 45(1), 45–62.
14. Rüdclé, A., Eger, S., Peyrard, M., & Gurevych, I. (2018). Concatenated p -mean Word Embeddings as Universal Cross-Lingual Sentence Representations. arXiv preprint arXiv:1803.01400.
15. Stamatatos, E., Potthast, M., Rangel, F., Rosso, P., & Stein, B. (2015, September). Overview of the pan/clef 2015 evaluation lab. In International Conference of the Cross-Language Evaluation Forum for European Languages (pp. 518–538). Springer, Cham.
16. Upadhyay, Shyam, et al. (2016). Cross-lingual models of word embeddings: An empirical comparison. arXiv preprint arXiv:1604.00425.
17. Vulić, I., & Moens, M. F. (2015). Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers) (Vol. 2, pp. 719–725).
18. Zweigenbaum, P., Sharoff, S., & Rapp, R. (2018, May). Overview of the third BUCC shared task: Spotting parallel sentences in comparable corpora. In Proceedings of 11th Workshop on Building and Using Comparable Corpora (pp. 39–42).