CrossLang: the system of cross-lingual plagiarism detection

Oleg Bakhteev
Moscow Institute of Physics and Technology
Moscow, Russia
bakhteev@phystech.edu

Alexandr Ogaltsov
Higher School of Economics
Moscow, Russia
aogalcov@hse.ru

Andrey Khazov
Antiplagiat Company
Moscow, Russia
hazov@ap-team.ru

Kamil Safin
Moscow Institute of Physics and Technology
Moscow, Russia
kamil.safin@phystech.edu

Rita Kuznetsova
IBM Research
Rueschlikon, Switzerland
kuz@zurich.ibm.com

Abstract

Plagiarism and text reuse become more available with the Internet development. Therefore it is important to check scientific papers for the fact of cheating, especially in Academia. Existing systems of plagiarism detection show the good performance and have a huge source databases. Thus now it is not enough just to copy the text "as is" from the source document to get the "original" work. Therefore, another type of plagiarism become popular — cross-lingual plagiarism. We present a CrossLang system for such kind of plagiarism detection for English-Russian language pair.

1 Introduction

Plagiarism detection and originality checking has become a major problem in Academia. There are several plagiarism detection systems (Turnitin, Antiplagiat.ru, Plagiarism.org, URKUND) that show good performance on verbatim plagiarism detection task. Possessing huge indexed collections of sources they detect copy-and-paste text reuse with high recall. Because of it another type of plagiarism becomes popular — when reused text was translated from another language Barrón-Cedeno et al. [2010], Bakhteev et al. [2015]. The translation can be both manual or automatic — modern machine translation systems could provide high quality text. Thus it is a very simple way to obtain "original" text without making any effort. There exist some articles described the problem of cross-lingual plagiarism detection for some language pairs Franco-Salvador et al. [2016a,b, 2013]. Unfortunately none of described approaches are production-ready. On the other side, the existing industrial tools are also unable to detect such kind of plagiarism. Thus there is a need for such tool that allows us to solve this problem at industrial scale with high quality.

In this paper [1], we focus entirely on the case when unauthorized text reuse comes from English to Russian language. The problem is formulated as follows: given a suspicious Russian document and English reference collection. Suspicious document could possibly contain passages translated from some documents from the collection. The problem is to find all translated passages in the suspicious

[1] This work was supported by RFBR project No.18-07-01441 and FASIE project No.44116.

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

document and their corresponding source passages in the documents from the collection. In general case, the language pair could be any. CrossLang is the new extension of the existing system for plagiarism detection — Antiplagiat [2], which is the most known system at Russia and CIS.

2 Related work

Based on the fact that we did not find the tools for cross-lingual plagiarism detection task (none of plagiarism detection systems announced about that), we provide the research papers that are dedicated to this problem (or considering the related topics).

Current state-of-the-art Franco-Salvador et al. [2013, 2016a], propose to construct semantic graph for each document. Text similarity evaluation is based on the similarity of the structures of these graphs. The main drawback of this approach is the resources requirement: the approach requires using multilingual ontologies, such as BabelNet Navigli and Ponzetto [2010], which cannot be used for commercial products.

Another class of papers similar to our approach is devoted to the document retrieval. In Ning et al. [2015], Vashchilin and Kushnir [2017] various methods of the document retrieval are compared. A number of works Le and Mikolov [2014], Dai et al. [2015] proposes to use paragraph or document vectors for this problem. One of the challenges of such methods is its computational expensiveness. In Boytsov et al. [2016] authors propose to use approximate nearest neighbors method for fast document retrieval, which allows to retrieve documents faster at the cost of significant memory usage. A number of works Ratna et al. [2016, 2017], Potthast et al. [2008] use methods for determining the text similarity like to latent semantic indexing Landauer and Dumais [2008], using decomposition of word-document matrix. This approach focuses on a significant text reuse, while our main task is to develop a system which can work with small-size text reuse cases.

As we use the monolingual approach the problem is very close to paraphrase detection task. Many approaches Tai et al. [2015], Yih et al. [2011], Kiros et al. [2015] have been developed for the paraphrase detection with neural sequence embedding. In Tai et al. [2015], Socher et al. [2014] authors propose to use recursive neural networks with dependency or constituency grammars. In Kiros et al. [2015] authors propose to use long short-term memory (LSTM) and gated recurrent unit (GRU). For the cross-lingual paraphrase detection one can employ deep learning methods based on bilingual autoencoders Chandar et al. [2014], Zhang et al. [2017] or on siamese neural networks Yih et al. [2011]. Opposing to works Kiros et al. [2015], Socher et al. [2011], He et al. [2015] we consider neural network outputs as embeddings in vector space for further approximate nearest neighbor search Wang et al. [2014].

3 CrossLang design

CrossLang is a service, consists of a set of microservices organized in five main components. Each of these microservices interacts with others via gRPC protocol. The microservise paradigm allows us to build more complex system — we can easily embed other microservices into the current solution if required. The key idea for CrossLang system is that we use the monolingual approach. We have suspicious Russian document and English reference collection. We reduce the task to the one language — we translate the suspicious document into English, because the reference collection is in English. After this step we perform the subsequent document analysis. Due to this fact the main challenge with the CrossLang design is that the algorithms for the plagiarism detection task should be stable to the translation ambiguity.

The main stages of CrossLang service is depicted in Figure 1. CrossLang receives the suspicious document from Antiplagiat system, when user send it for originality checking. Then it goes to *Entry point* — main service, that routes tha data between following stages:

1. *Machine Translation system* — microservice, that translates suspicious document into English. For these purposes we use Transformer Vaswani et al., open-source neural machine translation framework. For the details see section (3.1).

[2] https://www.antiplagiat.ru

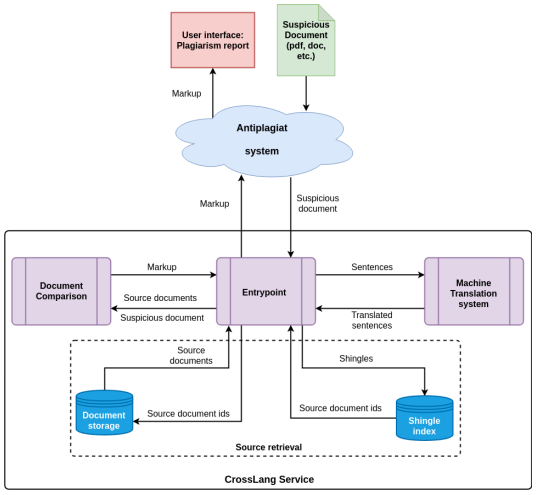

Figure 1: CrossLang service design.

2. *Source retrieval* — this stage unites two microservices: *Shingle index* and *Document storage*. Entry point receives the translated suspicious document's shingles (n -grams) and Shingle index returns to it the documents ids from the reference English collection. To deal with the translation ambiguity we use modified shingle-based approach. Document storage returns the Source texts from the collection by these ids. For the details see section (3.2).
3. *Document comparison* — this microservice performs the comparison between translated suspicious document and source documents. We compare not the texts themselves, but the vectors corresponding to the phrases of these texts. Thus we deal with the translation ambiguity problem. For the details see section (3.3).

Also in this section we would like to highlight the main differences of our work:

- The best of our knowledge it is the first system for cross-lingual plagiarism detection for English-Russian language pair. It is deployed on production and we could analyze the results. We could not find another examples of such system (even for other language pairs).
- The Source retrieval stage is often employed using rather simple heuristical algorithms such as shingle-based search Osman et al. [2012], Vashchilin and Kushnir [2017] or keyword extraction Ning et al. [2015], Dutta and Bhattacharjee [2014] because of simplicity of such methods and their computational efficiency. However, these methods can significantly suffer from word replacements and usually detect only near-duplicate paraphrase. We employ modified shingle-based method for this stage. In order to handle translation ambiguity we clusterize the words using word embedding model. We use semantic classes instead of words during this stage.
- Many articles on the cross-lingual plagiarism detection topic investigate the solutions based on bilingual or monolingual word embeddings Franco-Salvador et al. [2016a], Ferrero et al. [2017] for documents comparison, but almost none of them uses the phrase embeddings for this problem solution.

In the next sections we introduce how the main stages work.

3.1 Machine Translation system

We create machine translation system using state-of-the-art Transformer algorithm Vaswani et al.. We utilize Tensorflow realization [3] of it. Training dataset consists of approximately 30M parallel sentences. They were obtained from open-source parallel OPUS Tiedemann [2012] corpora, but also

[3] https://tensorflow.github.io/tensor2tensor/

we mine parallel sentences from Common Crawl.[4] Algorithm was trained for 5 epochs with batch size equals to 128 on Amazon p2.xlarge instance[5] We evaluate BLEU score Papineni et al. [2002] for *Russian* $\rightarrow$ *English* translation on news test 2018 dataset [6] and compare it with Google translator via API [7]. Results are in Table 1.

Table 1: BLEU of different systems

System	BLEU
Google	31.34
CrossLang Transformer	28.18

The CrossLang BLEU score lower than Google's BLEU score — this was to be expected. But it is very important to notice that we are not interested in ideal translation. Our main goal is to translate with sufficient quality for the next stages: Source retrieval and Document comparison.

3.2 Source retrieval

The method of source retrieval in the case of verbatim plagiarism is inverted index construction,where a document from the reference collection is represented as a set of its shingles, i.e. overlapping word n -grams, and a suspicious document's shingles are checked for matches with the indexed documents. There is one major problem with using the standard shingles — in our case the machine translation stage generates texts that differ too much from the sources of plagiarism. We argue that the source retrieval task can be solved with the help of a similar method that performs better than the method mentioned above; this improvement is achieved by moving from word shingles to word-class shingles, where each word is substituted by the label of the class it belongs to:

$$\{\text{word}_1, \ldots, \text{word}_n\} \rightarrow \{\text{class}(\text{word}_1), \ldots, \text{class}(\text{word}_n)\}.$$

Clustering the word vectors is a convenient and relatively fast way of obtaining semantic word classes. For the word embedding model we used `fastText` Bojanowski et al. [2016] trained on English Wikipedia. The dimension for word embedding model was set to 100. For the semantic word classes construction we applied agglomerative clustering on word embeddings with the cosine similarity measure to group words into word classes. We got 777K words clustered into 30K classes.

3.3 Document Comparison

For the comparison between retrieved documents and translated suspicious documents we introduce the phrase embedding model. Since in the final plagiarism report we must highlight phrases, we need to compare separate text fragments. We split documents (retrieved and suspicious) into phrases s and compare its vectors. Our goal is to learn representations for variable-sized phrases. For this purpose we learn a mapping: $s \rightarrow \mathbf{s}$, where $s = (\text{word}_1, \ldots, \text{word}_n)$. We learn this mapping both in unsupervised and semi-supervised training regimes. For mapping the word sequence into low dimensional space we use the encoder-decoder scheme. An encoder learns a vector representation of the input phrase and the decoder uses this representation to reconstruct the phrase in reverse order. During the training error between input phrase and reconstructed output phrase is minimized.

$$E_{rec} = \| \mathbf{s} - \hat{\mathbf{s}} \|^2 . \tag{1}$$

Encoder-decoder model is semi-supervised and consists of unsupervised part and supervised signal (see below). We train Seq2Seq model with attention Bahdanau et al. [2014]. As initial word vector representations for word_i we used word vectors from `fastText` model. For reconstruction error minimization E_{rec} (1) 10M sentences from Wikipedia articles were used.

In order to use information about phrase similarity we extend the objective function. We employ the margin-base loss from Wieting et al. [2015] with the limited number of similar phrase pairs

[4] http://commoncrawl.org/

[5] https://aws.amazon.com/ec2/instance-types/p2/

[6] http://www.statmt.org/wmt18/translation-task.html

[7] https://cloud.google.com/translate/docs/

$\mathcal{S} = \{(s_i, s_j)\}$:

$$E_{me} = \frac{1}{|\mathcal{S}|}\left(\sum_{(s_i,s_j)\in\mathcal{S}} \max(0, \delta - c_-) + \max(0, \delta - c_+)\right), \quad (2)$$

where $c_- = \cos(\mathbf{s}_i, \mathbf{s}_j) - \cos(\mathbf{s}_i, \mathbf{s}_{i'})$, $c_+ = \cos(\mathbf{s}_i, \mathbf{s}_j) + \cos(\mathbf{s}_j, \mathbf{s}_{j'})$, δ is the margin, $\mathbf{s}_{i'} = \arg\max_{\mathbf{s}_{i'}\in\mathcal{S}_b\setminus(\mathbf{s}_i,\mathbf{s}_j)} \cos(\mathbf{s}_i, \mathbf{s}_{i'})$, $\mathcal{S}_b \in \mathcal{S}$ — current mini-batch.

The sampling of so named "false neighbour" $\mathbf{s}_{i'}$ during training helps to improve the final quality without strict limitations on what phrases we should use at dissimilar.

This part of objective requires a dataset of similar sentences $\mathcal{S} = \{(s_i, s_j)\}$. We used double translation method as a method of similar sentences generation comparable to paraphrase. Consider a parallel corpus with pairs of Russian and English sentences. We translate Russian sentences back to English. This method of generation allows us to obtain pairs of sentences we process in CrossLang: both during training and during system usage we process a pairs of English sentences sentences translated from Russian into English by our translation system. We believe that this method gives us the opportunity to make phrase embedding model robust to our translation system errors since the machine translation errors can significantly influence the total performance of our framework. We used 100K pairs of sentences from OpenSubtitles Tiedemann [2012] corpus. The final objective function is:

$$\alpha E_{rec} + (1 - \alpha)E_{me}, \quad (3)$$

where α is a tunable hyperparameter that weights both of errors. [8]

For each phrase embedding from the suspicious document find M nearest vectors by cosine similarity from source documents using *Annoy*[9] library. The main idea of this function is to reduce the number of fragments pairs with a simple decision rule: for phrase embeddings pairs $(\mathbf{s}_i, \mathbf{s}_j)$ we consider that it is the plagiarism case if $\cos(\mathbf{s}_i, \mathbf{s}_j) > t_1$, where t_1 is a cosine measure threshold[10].

4 Experiments

4.1 Experiment on synthesized collection

There are no results and datasets for cross-lingual plagiarism detection task for language pair English-Russian. We create dataset for the problem and make it available. Visit [11] to get dataset and for details on dataset generation. For the whole framework we got Precision $= 0.83$, Recall $= 0.79$ and $F1 = 0.80$.

4.2 Monolingual plagiarism detection

Since our system translates the suspicious document into the language of the document collection it's quite natural to analyze the performance of our system not only for cross-lingual plagiarism detection problem but also for monolingual problem. For such experiment we do not use the machine translation service. In order to check performance of monolingual paraphrased plagiarism detection we exploit PAN'11 contest dataset and quality metrics Potthast et al.. Since we use PAN'11 corpora, it is naturally to compare algorithm performance with PAN'11 participants and other works that were tested on this corpora. Results of CrossLang and top-5 known previous methods are in Table 2.

5 Architecture

In this section we briefly describe how the microservices are deployed in our system. Our main technical requirement for the system is the document check speed and an ability to scale with the number of simultaneous document checks. Our microservices are stateless, i.e. they treat all the

[8] The objective (3) had the following value: $\alpha = 0.1$.. In the objective (2) $\delta = 0.3$

[9] https://github.com/spotify/annoy

[10] We set $t_1 = 0.6$

[11] http://tiny.cc/cl_ru_en

Table 2: PAN'11 performance comparison

Model	P	R	F	Plagdet
CrossLang	**0.94**	**0.76**	**0.84**	**0.83**
PDLK Abdi et al. [2015]	0.90	0.70	0.79	0.79
Sys-1 Wang et al. [2013]	0.86	0.69	0.76	0.75
Sys-2 Grozea et al. [2009]	0.75	0.66	0.7	0.69
Sys-3 Suchomel et al. [2012]	0.89	0.55	0.68	0.68
Sys-4Oberreuter et al. [2010]	0.87	0.56	0.68	0.67

operations as independent. This allows us to easily replace microservice backends and make the architecture more flexible. Currently we use RocksDB [12] for the Shingle index and Document storage services and Tensorflow Abadi et al. [2015] for the Machine translation and Document comparison services.

Our service is deployable on an 8-GPU cluster with Tesla-K100 GPUs, 128GB RAM and 64 CPU Cores. Depending on the requirements, the service is able to scale horizontally. For the fast rescaling we use Docker containerization and Consul and Consul-template for the service discovery and automatic load balancing.

The stress testing of our system showed that the system is able to check up to 100 documents in a minute. Despite the fact the average loading on our service is much lower, this characteristic of our service is important for withstanding peak loads.

6 Production performance

Our service was successfully deployed and connected to Antiplagiat system. We analyzed the performance of the service from May to July 2018. During this period students in Russia take exams and the average load on the system increases. For the production version of our service we indexed 30M documents from the Internet in addition to Wikipedia and arxiv we used earlier.

There were about 1.5M text reuse check in this period. We analyzed the statistics of document checks and found that 467K documents were detected as documents containing text reuse, which is about one of a third of all checked documents. However only a small part of documents contained significant reuse: we had about 70K document checks that contained more than 5% of cross-lingual text reuse. The median of text reuse level is 8.94 for such documents.

7 Conclusion

We introduced CrossLang — a framework for cross-lingual plagiarism detection for English Russian language pair. We decomposed the problem of cross-lingual plagiarism detection into several stages and provide a service, consists of a set of microservices. The CrossLang use a monolingual approach — reducing the problem to the one language. For this purpose we trained the neural machine translation system. Another two main algoithmic components are Source Retrieval and Document Comparison stages. For the Source Retrieval problem we used a modification of shingling method that allow us to deal with ambiguity after translation. For the Document Comparison stage we used phrase embeddings that were trained with slight supervision. We proposed method to make documents comparison efficient using approximate nearest neighbors method. We evaluated the effectiveness of our approach on several datasets. We also provided our own dataset. We integrated CrossLang in Antiplagiat system — the most popular and well-known system for plagiarism detection in Russia and CIS.

In future, we are going to develop the approach in several directions — use the documents vectors instead of shingles in source retrieval stage and modify our phrase embedding model. Also we will monitor system performance and analyze real users documents. We would like to conduct more experiments on the samples of real-world cases as long as corresponding data is available. Since our

[12] https://github.com/facebook/rocksdb

approach is rather general and does not use any language-specific features we believe that it can be applied to other language pairs. Therefore one of our plans is to implement our approach for language pairs other than English-Russian.

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

A. Abdi, N. Idris, R. Alguliyev, and R. Aliguliyev. Pdlk: Plagiarism detection using linguistic knowledge. *Expert Systems with Applications*, 07 2015.

D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

O. Bakhteev, R. Kuznetsova, A. Romanov, and A. Khritankov. A monolingual approach to detection of text reuse in russian-english collection. In *Artificial Intelligence and Natural Language and Information Extraction, Social Media and Web Search FRUCT Conference (AINL-ISMW FRUCT), 2015*, pages 3–10. IEEE, 2015.

A. Barrón-Cedeno, P. Rosso, E. Agirre, and G. Labaka. Plagiarism detection across distant language pairs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 37–45. Association for Computational Linguistics, 2010.

P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

L. Boytsov, D. Novak, Y. Malkov, and E. Nyberg. Off the beaten path: Let's replace term-based retrieval with k-nn search. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1099–1108. ACM, 2016.

S. Chandar, S. Lauly, H. Larochelle, M. Khapra, B. Ravindran, V. C. Raykar, and A. Saha. An autoencoder approach to learning bilingual word representations. In *Advances in Neural Information Processing Systems*, pages 1853–1861, 2014.

A. M. Dai, C. Olah, and Q. V. Le. Document embedding with paragraph vectors. *arXiv preprint arXiv:1507.07998*, 2015.

S. Dutta and D. Bhattacharjee. Plagiarism detection by identifying the keywords. In *Computational Intelligence and Communication Networks (CICN), 2014 International Conference on*, pages 703–707. IEEE, 2014.

J. Ferrero, F. Agnès, L. Besacier, and D. Schwab. Using word embedding for cross-language plagiarism detection. In *EACL 2017*, volume 2, pages 415–421, 2017.

M. Franco-Salvador, P. Gupta, and P. Rosso. Cross-language plagiarism detection using a multilingual semantic network. In *European Conference on Information Retrieval*, pages 710–713. Springer, 2013.

M. Franco-Salvador, P. Gupta, P. Rosso, and R. E. Banchs. Cross-language plagiarism detection over continuous-space-and knowledge graph-based representations of language. *Knowledge-Based Systems*, 111:87–99, 2016a.

M. Franco-Salvador, P. Rosso, and M. Montes-y Gómez. A systematic study of knowledge graph analysis for cross-language plagiarism detection. *Information Processing & Management*, 52(4): 550–570, 2016b.

C. Grozea, C. Gehl, and M. Popescu. Encoplot: Pairwise sequence matching in linear time applied to plagiarism detection. *3rd PAN Workshop. Uncovering Plagiarism, Authorship and Social Software Misuse*, 502:10, 01 2009.

H. He, K. Gimpel, and J. J. Lin. Multi-perspective sentence similarity modeling with convolutional neural networks. In L. Màrquez, C. Callison-Burch, J. Su, D. Pighin, and Y. Marton, editors, *EMNLP*, pages 1576–1586. The Association for Computational Linguistics, 2015.

R. Kiros, Y. Zhu, R. R. Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler. Skip-thought vectors. In *Advances in neural information processing systems*, pages 3294–3302, 2015.

T. K. Landauer and S. Dumais. Latent semantic analysis. *Scholarpedia*, 3(11):4356, 2008.

Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1188–1196, 2014.

R. Navigli and S. P. Ponzetto. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 216–225. Association for Computational Linguistics, 2010.

H. Ning, L. Kong, M. Wang, C. Du, and H. Qi. Comparisons of keyphrase extraction methods in source retrieval of plagiarism detection. In *Computer Science and Network Technology (ICCSNT), 2015 4th International Conference on*, volume 1, pages 661–664. IEEE, 2015.

G. Oberreuter, S. A. Ríos, and J. D. Velásquez. Fastdocode: Finding approximated segments of n-grams for document copy detection lab report for pan at clef 2010, 2010.

A. H. Osman, N. Salim, Y. J. Kumar, and A. Abuobieda. Fuzzy semantic plagiarism detection. In *International Conference on Advanced Machine Learning Technologies and Applications*, pages 543–553. Springer, 2012.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318, 2002.

M. Potthast, A. Eiselt, A. Barrón-cedeño, B. Stein, and P. Rosso. Overview of the 3rd international competition on plagiarism detection. In *In Working Notes Papers of the CLEF 2011 Evaluation*.

M. Potthast, B. Stein, and M. Anderka. A wikipedia-based multilingual retrieval model. *Advances in Information Retrieval*, pages 522–530, 2008.

A. A. P. Ratna, F. A. Ekadiyanto, Mardiyah, P. D. Purnamasari, and M. Salman. Analysis on the effect of term-document's matrix to the accuracy of latent-semantic-analysis-based cross-language plagiarism detection. In *Proceedings of the Fifth International Conference on Network, Communication and Computing*, ICNCC '16, pages 78–82, New York, NY, USA, 2016. ACM.

A. A. P. Ratna, P. D. Purnamasari, B. A. Adhi, F. A. Ekadiyanto, M. Salman, M. Mardiyah, and D. J. Winata. Cross-language plagiarism detection system using latent semantic analysis and learning vector quantization. *Algorithms*, 10(2):69, 2017.

R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 151–161, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

R. Socher, A. Karpathy, Q. V. Le, C. D. Manning, and A. Y. Ng. Grounded compositional semantics for finding and describing images with sentences. *TACL*, 2:207–218, 2014.

Suchomel, J. Kasprzak, and M. Brandejs. Three way search engine queries with multi-feature document comparison for plagiarism detection. 09 2012.

K. S. Tai, R. Socher, and C. D. Manning. Improved semantic representations from tree-structured long short-term memory networks. *CoRR*, abs/1503.00075, 2015.

J. Tiedemann. Parallel data, tools and interfaces in opus. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, 2012.

S. Vashchilin and H. Kushnir. Comparison plagiarism search algorithms implementations. In *Advanced Information and Communication Technologies (AICT), 2017 2nd International Conference on*, pages 97–100. IEEE, 2017.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30*.

J. Wang, H. T. Shen, J. Song, and J. Ji. Hashing for similarity search: A survey. *arXiv preprint arXiv:1408.2927*, 2014.

S. Wang, H. Qi, L. Kong, and C. Nu. Combination of vsm and jaccard coefficient for external plagiarism detection. In *2013 International Conference on Machine Learning and Cybernetics*, volume 04, pages 1880–1885, 2013.

J. Wieting, M. Bansal, K. Gimpel, and K. Livescu. Towards universal paraphrastic sentence embeddings. *CoRR*, abs/1511.08198, 2015.

W.-t. Yih, K. Toutanova, J. C. Platt, and C. Meek. Learning discriminative projections for text similarity measures. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, CoNLL '11, pages 247–256, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-92-3. URL `http://dl.acm.org/citation.cfm?id=2018936.2018965`.

B. Zhang, D. Xiong, and J. Su. Battrae: Bidimensional attention-based recursive autoencoders for learning bilingual phrase embeddings. In *Proc. of AAAI*, 2017.