Siamese Twins with Bert Encoder to Detect Paraphrasing

Siddhi Bajracharya
Applied AI Research Lab, Department of Computer Science University of South Dakota, Vermillion, SD 57069
siddhi.bajracharya@coyotes.usd.edu

***Abstract*— Plagiarism is an evident problem that is widespread in academia [1]. According to a study [2], faculty 68 percentage of the faculty reported plagiarism committed by students. Another study [3], from plagiarism.org reports that 38 percentage of students admit to copying or paraphrasing from different sources. It is however unclear the number of students who plagiarize from fellow students. There are tools such as Jplag, MOSS, Plaggie, SIM and Turnitin [1] that can detect plagiarism by comparing the work to a database of large number of works. Although they can successfully detect paraphrased sentences, they can be expensive for smaller institutions. In this work, I propose using Siamese Twin network along with BERT encoder to retrieve the most similar work from a pool of works submitted by the students. Instructors, teachers, and professors can facilitate this work to check the most common assignments submitted by their students. For repeatability, the code is hosted in github repository: https://github.com/siddhi47/siamese-bert.**

***Index Terms*—Paraphrasing, plagiarism, Siamese twin network, Bert encoder**

I. Introduction

Paraphrase detection is one of the challenging tasks in machine learning. Studies [1]–[3] show that plagiarism is a widespread problem in academia. This problem has exacerbated because of introduction of artificial intelligence (AI) tools such as ChatGPT, Google BARD, BingChat, and so on. The popularity of these large language model tools have made it easy to paraphrase work. In the context of academia, professors have a difficult time grading assignments because most student copy assignments by paraphrasing using the aforementioned tools. Not only it harms academic integrity, but also leaves a remarkable effect on student's creativity. The paraphrased text can be identified through vigorous analysis, but it diverts the professor's focus to paraphrasing instead of examining the purpose of the assignment. In this project done as a part of Information Storage and Retrieval course, I propose a deep learning architecture to measure the similarity between student's assignments using Siamese twin network. To generate textual features, I have proposed to use BERT encoder. Our contributions are listed as follows:

- Propose a siamese twin architecture using BERT to demonstrate use of siamese network retrieve most similar documents (assignments) from a pool of documents.
- Creation of a new dataset using abstract and their paraphrased text using ChatGPT.

II. Data collection

At the time of this research project, we could not find any relevant dataset related to paraphrasing. Thus, a challenge was to collect dataset for original and paraphrased texts. we did not have any access to assignments, and hence deliberated to use abstracts from conference papers as original text files. Most datasets such as microsoft research paraphrase corpus (MPRC) [4] are short and are paraphrased by humans and not state of the art tools like ChatGPT, hence we collect our own dataset for this work. We scraped abstract of 100 conference papers from 2021 IEEE International Conference on Autonomous Systems (ICAS) using Pythhon's request and BeautifulSoup packages. For paraphrased dataset, we used ChatGPT to paraphrase the 100 text files. The ChatGPT API is not free, thus we used the web version of ChatGPT to paraphrase 100 abstract. Due to slight different HTML responses from these 100 abstracts, we had to filter out 18 text files, leaving me with only 82 text files to work with.

The paraphrased and original texts were placed into two separate directory with same name. The files were named as 0.txt, 1.txt to 99.txt for both paraphrased and original text files. To make it more challenging for the model to distinguish between paraphrased and original text, we take abstracts from the same conference. This ensures that each abstracts are somewhat related to each other. Since average character counts for the extracted abstracts is around 200 characters, we trim the abstracts to 200 and employ padding when the abstracts are less than 200. We then create BERT encoding and corresponding mask. All this is done on the fly using PyTorch's Dataset class. We perform 80-20 split to create traning and validation split. No further preprocessing is required for the texts as BERT works well without pretrainig.

III. Related works

Paraphrase detection is a well known researched problem in NLP and information retrieval. In recent years, the advent of neural language models, exemplified by architectures such as BERT [5] (Bidirectional Encoder

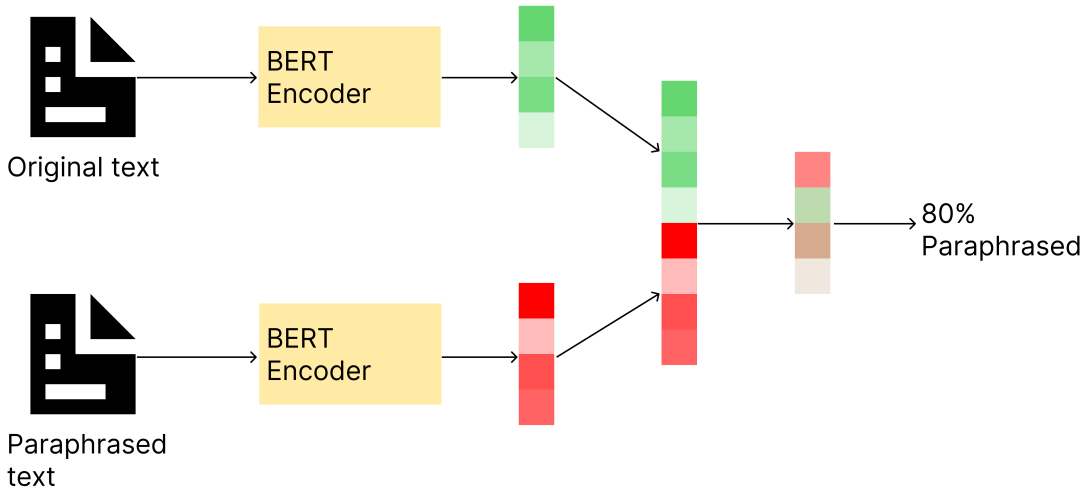

Fig. 1: Proposed Siamese network with BERT encoding. The encoding from original and paraphrased texts are extracted using BERT encoder. The embedding are then concatenated, and passed through a fully connected feed forward network. The output is a value between 0 and 1, where 1 is the most similar and 0 is the most dissimilar.

Representations from Transformers), has significantly advanced natural language understanding and generation capabilities. However, this progress has raised concerns about the potential threat to academic integrity, particularly in the context of text paraphrasing. The ability of neural language models to perform human-like paraphrasing poses a challenge for identifying machine-obfuscated plagiarism.

Addressing this concern, a notable contribution [6] has been made in the form of a large-scale dataset created by [Author(s)] to facilitate research on detecting machine-generated paraphrases. This dataset comprises documents paraphrased using Transformer-based models such as BERT, RoBERTa, and Longformer. It encompasses a diverse range of sources, including paragraphs from scientific papers on arXiv, theses, and Wikipedia articles, along with their corresponding paraphrased counterparts, totaling 1.5 million paragraphs. Notably, the research demonstrates that the paraphrased text maintains the semantics of the original source, underscoring the efficacy of these neural models in preserving contextual meaning.

Another proposed model [7] adopts the shortest path between synsets in WordNet as a foundational element for assessing the relatedness between two sentences. WordNet, a lexical database of the English language, provides a rich network of interlinked synsets based on conceptual-semantic relationships. By utilizing the shortest path between synsets, the study aims to capture the semantic nuances between sentences, offering a nuanced perspective on paraphrase detection.

Another research [8] introduces a novel approach for paraphrase sentence detection, employing Siamese Similarity with Word2vec embedding. The study underscores the crucial role of training data quantity, revealing a dominant impact on new data accuracy. With 800,000 pairs, the model achieves an impressive 99
accuracy on training data and 82.4% on new data, surpassing alternative methods. Notably, the research highlights the nuanced relationship between training data quantity and its effect on generalization. These findings contribute valuable insights into optimizing paraphrase detection systems within NLP applications.

Another article published in 2023 [9] proposes a novel Siamese architecture-based model for English–Hindi language pairs, combining convolutional neural network (CNN) and bidirectional long short-term memory (Bi-LSTM) components. The model outperforms existing methods on the Microsoft Paraphrase Corpus adapted to English–Hindi, achieving 67%, 72%, and 67% weighted average precision, recall, and F1-measure scores. This highlights the model's efficiency in representing cross-lingual text and its effectiveness in CLPD compared to baseline models.

IV. Proposed methodology

The proposed methodology is shown in figure 1. We propose the use of Siamese twin network to detect similarity between the texts. Siamese networks are popular mostly for one-shot learning for image classification problems [10]–[12]. A typical usage of Siamese network is shown in figure 2. The network is called Siamese twin network because the same architecture is used twice. Siamese networks are utilized to measure similarity by learning embeddings that capture the relationships between pairs of input samples. They excel in tasks requir-

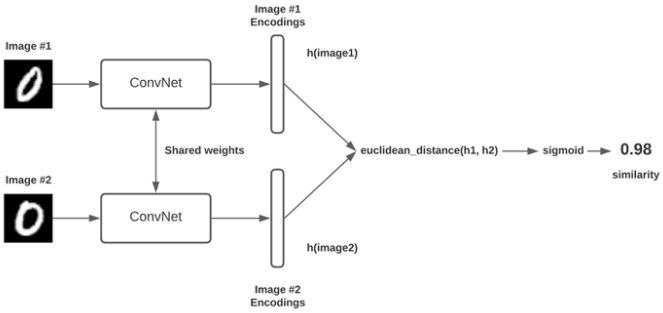

Fig. 2: A typical architecture of Siamese network showing two identical network structure for image processing. The embeddings are concatenated and passed through a feed forward network.

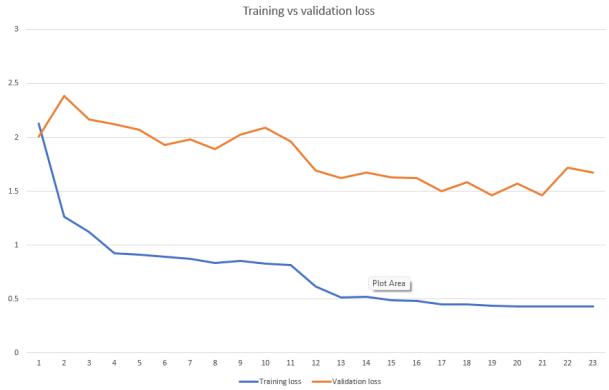

Fig. 3: Training and validation loss for 24 epoch. Blue line is training and orange line is validation loss.

ing contrastive learning, pairwise comparison, and robustness to input variations. Their versatility, efficiency, and adaptability make them valuable for various applications, especially in scenarios with limited labeled data and complex relationships between instances.

A key concept in Siamese network is contrastive learning. Contrastive learning is a machine learning paradigm that aims to teach a model to distinguish between similar and dissimilar pairs of data. The fundamental idea is to map similar samples closer together and dissimilar samples farther apart in the learned feature space. This approach is particularly useful when labeled data is scarce or expensive to obtain. For our propsosed architecture, we use contrastive learner as a loss function. The contrastive loss for a pair of instances x_i and x_j is given by:

$$L_{i,j} = -\frac{1}{2} y_{i,j} \log(\hat{y}_{i,j}) - \frac{1}{2}(1 - y_{i,j}) \log(1 - \hat{y}_{i,j}),$$

where:

- $L_{i,j}$ is the contrastive loss for the pair of instances x_i and x_j.
- $y_{i,j}$ is a binary indicator, where $y_{i,j} = 1$ if x_i and x_j are similar, and $y_{i,j} = 0$ if they are dissimilar.
- $\hat{y}_{i,j}$ is the predicted similarity score between $f(x_i)$ and $f(x_j)$ produced by a similarity function. This function typically involves measuring the distance or similarity between the embeddings (e.g., Euclidean distance, cosine similarity).

The overall contrastive loss for a batch of pairs is often calculated by averaging the individual losses:

$$L = \frac{1}{N} \sum_{i=1}^{N} L_{i,j},$$

where:

- N is the total number of pairs in the batch.

Our final architecture consists of 109 million parapeters with only 394 trainable parameters. The twin BERT encoders extract the embedding from the paraphrased and original text. Each embedding is reduced to 256 linear units, which is then passed through a feed forward network to get a single output followed by Sigmoid activation. Sigmoid activation are used for binary classification, which is the type of problem we are trying to solve.

V. Experimental setup

For our research, we employ PyTorch Lightning as our deep learning framework. It is a comprehensive library with minimal boilerplate codes. We perform our training on University's Lawrence HPC[1] cluster with Ampere GPU. The model was trained for 50 epochs with a batch size of 2. Adamdelta optimizer was used with a learning rate of 0.0001. Since we care about both the quality and quantity of the documents retrieved, we use f1-socre, which is the harmonic mean of precision and recall for evaluation.

VI. Results and analysis

The training and validation loss of 24 epoch is shown in figure 3. The loss beyond 24 epoch started to degrade and thus the model was only trained for 24 epochs. This suggests that the model's parameters are not being trained well. The results suggest that the model is not performing well. The cause can be attributed to very small dataset size. Although the Siamese network should work for smaller datasets, the model performs poorly for the collected dataset. Due to small dataset, the validation split only had 16 dataset, which may not be good enough to provide significant evaluation, thus, we only report the training f1 score. The model achieves 0.56 in the training loop, which is below the results observed in the literature review.

[1]Computation was performed on Lawrence Supercomputer at the University of South Dakota awarded by NSF.1626516

VII. Conclusion and future works

In this study, we have leveraged the Siamese twin architecture to match the text document with paraphrased text. We employ BERT encoder for extracting text encoding and a fully connected neural network architecture to perform a binary classification. We retrieve the documents that are at least 50% similar to the original text. Although the architecture performs poorly, we successfully demonstrate that siamese network with BERT encoders show a promising result to detect paraphrasing in text documents. The research project can be extended and improved in following ways:

- Improve the model performance by collecting a larger dataset by utilizing API for paraphrasing instead of performing it manually.
- Experimentation with different language models may improve performance of the model.

References

[1] Vandana, "A comparative study of plagiarism detection software," *IEEE 5th International Symposium on Emerging Trends and Technologies in Libraries and Information Services, ETTLIS 2018*, pp. 344–347, 10 2018.

[2] "Plagiarism in college." [Online]. Available: https://www.affordablecollegesonline.org/college-resource-center/plagiarism-prevention-and-awareness/

[3] "Plagiarism: Facts stats - plagiarism.org." [Online]. Available: https://www.plagiarism.org/article/plagiarism-facts-and-stats

[4] J. Wahle, T. Ruas, T. Foltýnek, N. Meuschke, and B. Gipp, "Identifying machine-paraphrased plagiarism," 03 2021.

[5] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "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)*, J. Burstein, C. Doran, and T. Solorio, Eds. Association for Computational Linguistics, 2019, pp. 4171–4186. [Online]. Available: https://doi.org/10.18653/v1/n19-1423

[6] J. P. Wahle, T. Ruas, N. Meuschke, and B. Gipp, "Are neural language models good plagiarists? a benchmark for neural paraphrase detection," in *2021 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, 2021, pp. 226–229.

[7] J. C. Lee and Y.-N. Cheah, "Paraphrase detection using semantic relatedness based on synset shortest path in wordnet," in *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, 2016, pp. 1–5.

[8] A. A. Aziz, E. C. Diamal, and R. Ilyas, "Paraphrase detection using manhattan's recurrent neural networks and long short-term memory," in *2019 6th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 2019, pp. 432–437.

[9] B. Agarwal, M. K. Gupta, H. Sharma, and R. C. Poonia, "Siamese-based architecture for cross-lingual plagiarism detection in english–hindi language pairs," *Big Data*, vol. 11, no. 1, pp. 48–58, 2023, pMID: 36260373. [Online]. Available: https://doi.org/10.1089/big.2020.0243

[10] L. Liying and W. Muqing, "Visual tracking based on siamese neural network with non-local attention network," in *2022 IEEE 8th International Conference on Computer and Communications (ICCC)*, 2022, pp. 1905–1909.

[11] M. Wu and J. Zhao, "Siamese network object tracking algorithm combined with attention mechanism," in *2023 International Conference on Intelligent Media, Big Data and Knowledge Mining (IMBDKM)*, 2023, pp. 20–24.

[12] Y. Haoran, "Face detection using the siamese network with reconstruction supervision," in *2023 3rd International Conference on Information Communication and Software Engineering (ICICSE)*, 2023, pp. 44–50.