A Paraphrase Identification Approach in Paragraph length texts

Arwa Al saqaabi
Dr. Craig Stewart
Dr. Eleni Akrida
Prof. Alexandra Cristea

ABSTRACT

How to measure the semantic similarity of natural language is a fundamental issue in many tasks, such as paraphrase identification (PI) and plagiarism detection (PD) which are intended to solve major issues in education. There are many approaches that have been suggested, such as machine learning (ML) and deep learning (DL) methods. Unlike in prior research, where detecting paraphrases in short and sentence-level texts has been done, we focus on the not yet explored area of paraphrase detection in paragraphs. We consider that the meaning of a piece of text can be broken into more than one sentence, this is over and above the sentences as extracted from two benchmark datasets (Webis-CPC-11 and MSRP). TF-IDF, Bleu metric, N-gram overlap, and Word2vec are used as features, then SVM is invoked as a classifier. The contribution of this paper clearly indicates that, on a commonly used evaluation set, text at the length of a paragraph is more appropriate to consider than short or long text for ML and DL approaches. Additionally, our method outperforms the existing work done on the Webis-CPC-11 dataset.

Keywords

Natural Language Processing, Machine Learning, Deep Learning, Paraphrase Identification, Paragraph Length

1. INTRODUCTION

When students submit their work, institutions have to verify if the work is free of plagiarism. To overcome the limit of human abilities in terms of scalability (e.g. time to check and consistency) machine and deep learning techniques are applied for plagiarism and paraphrase detection tasks. Plagiarism is defined as using someone's written work without giving reference to the original source, or claiming the ideas taken from the work of others [22]. In some instances, the copying of many words from the original source, regardless of the provision of a reference, is also considered an act of plagiarism [4]. The modification of sentences in such a way that the original structure of the sentences, without acknowledgment, is used by the author, also falls in the category of plagiarism. According to Ehsan et al. [14], plagiarism detection methods are divided into two main categories, which are *intrinsic* plagiarism detection and *external* plagiarism detection methods. Intrinsic methods are implemented to detect the parts of the text that are inconsistent, while external methods can match suspicious passages in a text to the source(s), detecting exact verbatim copying and paraphrased text [23].

Bhagat and Hovy [6] define paraphrasing as a means of conveying the same meaning, but with different sentence structure and wording. This definition clearly does not include exact verbatim reproduction as a case of paraphrasing. For our purposes, let us have two different texts, A and B. If the information, ϕ, which can be derived from A, can also be inferred from B, and vice versa, then A is a paraphrase of B (Equation 1): α represents a given domain or background knowledge [8].

$$(\mathbf{A} \wedge \boldsymbol{\alpha}| = \boldsymbol{\varphi}) \Leftrightarrow (\mathbf{B} \wedge \boldsymbol{\alpha}| = \boldsymbol{\varphi}) \text{ , where } \boldsymbol{A} \neq \boldsymbol{B} \quad (1)$$

From this definition, it is obvious that paraphrase identification (PI) is implicitly part of plagiarism detection. (PD) Both PI and plagiarism detection have assumed a tremendous importance for academic institutions, researchers, and publishers concerned for the preservation of academic integrity [3]. PI is a method that aims to measure the degree of similarity between two given texts [11, 15]. PI also helps determine whether the two texts share the same meaning, which plays a vital role in natural language applications, such as plagiarism detection, summarisation of textual material, and machine translation. Semantic similarity is also used in several other activities, such as to retrieve information [21], answer questions [5] [31,1] and clustering [7].

Attempts to solve the problem of paraphrase identification in past studies were mainly focused on comparing words in sentences [28,29], phrases in sentences [2], or sentence to a sentence [12, 24]. These studies achieved robust results. However, comparing each sentence in the suspicious document (i.e., a students' assignment), to all sentences in the source documents, is not an efficient approach for long texts. Additionally, existing studies are ignoring the fact that sentence semantics could be distributed in a paragraph as a passage–level paraphrase type, which is more complex to recognise. Thus, we aim to develop a method for recognizing paraphrasing in paragraphs, henceforth called *passage-level paraphrasing*. This approach considers a paragraph as a basic unit, avoiding comparing all sentences of the documents as separate entities.

Existing work investigated paraphrasing that is accrued at sentence level [16, 30]. Additionally, prior works count exact and quasi-exact sentences as a paraphrased text [2, 25]. To the best of our knowledge, this study is the first to allow for passage-level paraphrasing (beyond sentence-only) text length. We have chosen to focus on passage-level paraphrasing, as we argue that it is a common and naturally occurring way to consider paraphrasing, more so than the previously studied sentence-level paraphrasing. In addition, we analyse how the text length affects machine learning (ML) model accuracy. For this purpose, we compare ML approaches that are mainly based on handcrafted features, as well as state-of-the-art deep learning sentence representation models, such as word2vec.

A. A. Saqaabi, C. Stewart, E. Akrida, and A. Cristea. A paraphrase identification approach in paragraph length texts. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining*, pages 782–788, Durham, United Kingdom, July 2022. International Educational Data Mining Society.

https://doi.org/10.5281/zenodo.6852990

For further research, constricting a new dataset is required to consider the text length and paraphrasing type. For purpose of this research, we provide the following definitions:

a. *Sentence level paraphrasing*: the meaning of one sentence is paraphrased into exactly one other sentence (example: data in the MSRP dataset);
b. *Passage level paraphrasing*: the semantics of a piece of text of multiple sentences are paraphrased into a potentially different number of sentences without a one-to-one mapping between sentence semantics (example: data in the Webis-CPC-11 dataset);
c. *Sentence-length level*: represents a **short text** length, which is less than 50 words;
d. *Paragraph-length level*: represents a **mid-text** length, which consists of about 100 words, this is the average length of a paragraph [20].
e. *Passage-length level*: **long text** which consists from more than a paragraph (containing 150+ words).

Hence, our main research questions (RQ) are:

RQ1: How does the length of a piece of text affect the efficiency of the paraphrase identification approach used?

RQ2: What type of features are more effective for the problem of paraphrase identification on sentence - and passage level of paraphrasing, respectively?

RQ3: How effective are current state-of-the-art paraphrase identification methods for the problem of paraphrase identification in long text (paragraph) and (passage-level paraphrase)?

2. RELATED WORK

Machine learning (ML) and deep neural networks (DNN) attract the PI researchers and a lot of efforts have been devoted in this area.In these works, features are extracted by considering N-gram overlap features, metrics like Bleu, syntactic features, and semantic features from external knowledge such as WordNet or pretrained word embedding.

Cordeiro et al. [9, 10] implemented experiments that extract features from text, by applying a variety of metrics such as Bleu, edit distance which calculates how many character or word insertions, deletions, and replacements are required to change one string into the other, and Sim, Word Simple N-gram Overlap. They evaluated the experiments on the MSRP and the Knight and Marcu Corpus (KMC), where the paraphrased sentence is a shortened or summarized version of the original one. The Sim metric present the highest accuracy after removing the equal and quasi-equal samples from the dataset. For more investigations on the metrics' efficiency, they defined two types of paraphrasing, which are *symmetrical paraphrasing* (SP) and *asymmetrical paraphrasing* (AS). Symmetrical sentence pairs contain the same information, while in the asymmetrical paraphrasing, at least one sentence has more information. The result shows that the Sim metric is efficient for AS, while the Logisim metrics, based on the theory of exclusive lexical links between pairs of short text, is better for SP. Rather than implementing a specific threshold value to do binary classification, these metrics also were fed as text features extracted from the Webis-CPC-11 dataset, which has longer text samples, to a classifier such as SVM and k-nearest neighbours [8]. Regardless of the accuracy of these models, the Sim metric is suitable only for short texts, because of its demands on computing time.

Ferreira et al. [16] evaluated different ML algorithms, namely RBFNetwork, BayesNet , C4.5, and SMO on short text. They measured the lexical features from a Bag of Words (BOW), which breaks a text into all of its unique words and counts the number of times each word appears, syntactic and semantic features from the Resource Description Framework (RDF) based on dependence tree graphs. It mainly tackled two specific issues: sequences with the same meaning, but different terms, and the word-order problem. The results of the RBFNetwork and BayesNet algorithms outperform others, with accuracies of 75.13 and 74.08, respectively, both of which are measures of performance discussed in more detail in Section 3.3. Despite the fact that it did not improve overall outcomes, it significantly recognized the meaning of sentences that shared the same words but in a different order.

Wan et al. [29] designed an approach that considered 17 syntactic dependency features, to examine their effects on the accuracy of different machine learning algorithms, namely Naive Bayes learner, C4.5 decision tree, support vector machine, and K-nearest neighbour, to indicate dissimilarity between a pair of sentences. They claimed that dependency and N-gram features enhanced the classifier to recognise falsely paraphrased cases. In addition, avoiding lemmatisation was shown to keep the signs of differences in meaning and focus between sentences. However, more of the correctly paraphrased cases were identified as a negative, decreasing the overall accuracy of the approach. Moreover, they evaluated their experiment on a partial MSPC, because some cases led to stopping the parsing script. To leverage the limitations of this study, Ji and Eisenstein [18] considered the same features of Wan et al. [29] and implemented them on the whole MSRP corpus. Additionally, they developed a metric that computed the discriminability of features between sentences, called Term Frequency Kullback Leibler Divergence (TF-KLD). It counted the probabilities that appeared on paraphrased and non-paraphrased sentences, to re-weight features before factorisation, to obtain latent representations of the text. It clearly outperformed TF-IDF by 4% in accuracy and 1% in F1 score on MSRP. Moreover, they combined other features, such as unigram and bigram, overlapping fine-grained features, which raised the accuracy from 72.75 to 80.41. TF-KLD improved discriminatively distributional features while reducing others.

From another perspective, Vrublevskyi and Marchenko [28] extracted dependency tree, IDF and Bleu features from natural language. They concatenated word embedding with dependency tree features, to show that this combination can be useful to detecting paraphrase. However, this model did not outperform the state-of-the-art in that area [18].

Ji and Eisenstein [18] and Wan et al. [29] noted the need for more investigating on another dataset, also considering long text, such as a paragraph, where these studies examined only the MSPC corpus, where the maximum length of a sentence is 36 words [13]. In addition, using parse trees to solve problems restricted an approach to single sentences [19]. Also, BOW was unable to consider word order which is a vital textual feature in PI [30].

Nguyen et al. [24] developed an algorithm based on external knowledge and word embedding. It takes name entities (e.g., US) and rewrites them in words (e.g., United States). Additionally, they have applied the continuous bag of words CBOW and Skip-Gram models to extract interdependent features based on pre-trained word embedding. CBOW predicts a target word based on its context and Skip-Gram does the opposite, predicting context words according to the target word. As a part of the methodology, more features were also included that help to measure semantic relatedness based on external knowledge resources such as WordNet.

Features extracted from sentences with and without pre-processing. Then Support Vector Machine SVM is involved for classification task. It examined on MSRP, SemEval and P4PIN datasets achieving high accuracy 84.17, 83,73, and 95,22 respectively by considering features based on the external-knowledge resource. Unfortunately, they didn't report other evaluation metrics like F1 score, precision, and recall.

The above-mentioned work considered words or sentences as a mean unit of calculating semantic similarity in text segments. In addition, methods like SVM and BOW fall short of delivering a high-quality solution for extracting the semantics meaning of natural language. Particularly, SVM works better when integrated with deep learning models for recognizing paraphrased sentences [32].

Several academics have recently used deep neural networks to model sentence pairs to leverage the shortness of non-deep learning methods. Vrbanec and Meštrović [26, 27] conducted two studies on different models of sentence semantic representation such as word2vec, Glove and Fast-Text. The experiments were done on three datasets namely MSRP, Webic-CPC-11 and C&S. Because of the pairs of sentences that are semantically unrelated and very similar lexically, no specific model outperforms others on all datasets, however, USE provides high accuracy and F1 score. They also compare similarity metrics with two types of word embedding deep learning models: word2vec and Fast-Text [27].

Kenter and De Rijke [19] focused on word2vec and Glove semantic sentence representation. They extract word level and short sentence level features by word alignment and word embedding beside the saliency weighted semantic graph. This approach is corpus-based hence no use of external knowledge source is needed. It measures the semantic similarity on sentence level ignoring the importance of the word order in PI downstream task. Although applying word alignment to extract syntactic and semantic relations between words of the sentence and feeding them into SVM model as features showed the significance result on PI task, it needs to be examined on paragraph level. The authors show that classifiers which trained to predict semantic similarities between short texts can benefit from saliency-weighted semantic networks. In addition, concatenating of pre-trained word embedding models obtain better scoring than WordNet-based approaches.

Although these studies were done on more than one of public paraphrase dataset, they did not take into account the verity of numbers of words on each sample, nor the type of paraphrase have been done on different datasets. More importantly, they consider typical samples as paraphrased cases that do not state the paraphrase definition.

To overcome the limitation with sentence representation models that are based on a unidirectional encoder, Devlin et al. [12] proposeed Bidirectional Encoder Representation from Transformers (BERT), which uses a masked language model and next sentence prediction, and is fine-tuned with one additional output layer. BERT has been demonstrated to achieve state-of-the-art outcomes on a wider array of sentence-level and token-level NLP tasks. Specifically in the PI task, it evaluated on MSRP with 89.30 percent of accuracy. This high accuracy raises the machine prediction accuracy to be closer to human performance.

As we see, BERT as a transformer learning model outperforms other sentence representation models because it generates a context vector representation that can discriminate word meaning in different contexts rather than giving the same weight for each word wherever it occurs [12].

Deep learning techniques have attracted a lot of attention in different research fields regarding to its impressive performance. In the PI field, researchers employ deep learning models to detect semantic similarity mainly in short text. In a way to show the efficiency of deep learning models over machine learning on PI task. Hunt et al. [17] compared the accuracy of two machine learning models with three different deep neural network models. Results illustrate that all DL models' accuracy outperforms LR and SVM models. The lowest accuracy is obtained by Siamese NN (~62) while the best accuracy is (~82) from LSTM RNN on the PI task.

3. MOTHODOLOGY

Prior research has investigated the efficiency of pre-processing techniques such as removing stop words and word lemmatizing [29], similarity metrics such as cosine, soft cosine and Euclidean [27] and using pre-trained word embedding models [26] on the PI downstream task. Here we examine how the length of text affects the model's accuracy in determining the appropriate number of words that could provide enough semantic information for machine models. Specifically, do short texts (sentences), mid length texts (paragraphs), or long texts (passages and paragraphs), provide sufficient semantic detail for the models? What type of features are most appropriate to use when attempting to identify paraphrases in sentence or paragraph texts? More importantly goal is that how to measure semantic meaning in paragraph length level to enhance other fields such as plagiarism detection, summarization, and text matching. Additionally, how the state-of-the-art autoregressive method advance the PI task?

3.1 Dataset

3.1.1 Microsoft Paraphrase Corpus (MSRP)

Dolan et.al. [13] presented separated sets of sentence pairs for training and evaluation. MSRP contains 4076 pairs of short text for train and 1725 for evaluation, taken from news sources on the internet. Human reviewers have then determined if each pair has a semantic equivalence. Each sentence pair is then labelled by 0 or 1, which represent negative and positive labels respectively.

3.1.2 Webis Crowd Paraphrase Corpus 2011 (Webis-CPC-11)

Burrows et.al. [8] provided 7859 possible text paraphrases pairs by Mechanical Turk crowdsourcing. The corpus consists of 4067 acceptable paraphrased pairs (meaning that one piece of text is a paraphrase of the other) and 3792 non-paraphrased pairs.

Most of the existing PI experiments are done on MSRP, which contains sentence-level paraphrases, so the existing results measure sentence similarity, whereas the most common and natural paraphrase is at the passage-level as seen in Webis-CPC-11. Another point of comparison between these corpora is the length of text which is vital for our study. The maximum sentence length in MSRP is 36 words while in the Webis-CPC-11 it is about a thousand words. This variety of document lengths in the Webis-CPC-11 dataset enables us to study how the text length can affect the ML model's results and to determine the best length of text that could be applied for ML and DL models.

3.2 Features Extracted

Based on the purpose of this study, we select the most important features in the PI task that represent text into numeric values. Each work discussed in Section 2 has implemented at least one feature of TF-IDF, Bleu, dependency tree, N-gram overlap or Word2vec [9, 10, 17, 18, 19, 24, 26, 27, 29]. However, most of these works

are done on the MSRP dataset which represents sentence level paraphrase, whereas we present a study on passage level paraphrase as seen in the Webis-CPC- 11 dataset.

4. EXPERIMENT

As we aim to study how the length of a text can affect the accuracy of using a specific or a combination of features in ML and DL models, we use MSRP as it focuses on short text length and Webis-CPC-11 as it contains a variety of text length samples. Therefore, we clean, then divide the samples from Webis-CPC-11 into three subsets based on the text length after removing the empty samples containing no text. Firstly, we remove identical sample texts from the Webis-CPC-11 dataset to satisfy the requirements of the paraphrase definition mentioned in the introduction, see equation (1). So, given a text pair (text 1, text 2), text 1 must be different from text 2 but carrying the same meaning. We call the resulting new dataset Webis-CPC-21, following the trend of the original dataset naming where 11 refers to 2011, the year of creation. However, we perform our experiments on the Webis corpus both with and without these identical samples to compare our results to state-of-the-art literature where possible. Secondly, we split the Webis-CPC-21 into three sub corpora: short text, where the maximum length of samples is 50 words; mid text represents the paragraph length (51-150 words) as the average length of a paragraph in English consists of 100 words [20]; long text containing samples of 151-500 words. We keep MSRP in its original form, as its samples consist of short text length (less than 40 words). Each dataset has numbers of negative and positive labelled samples. Positive and negative label refer to pair of text that are paraphrased and non-paraphrased respectively.

4.1 Pre-Processing

The data cleaning process includes removing irrelevant punctuation and stop words, which are commonly used words such as 'a', 'in' and 'the'. There isn't a single list of stop words that applies to every NLP task; we use the stop words list constructed by NLTK (Natural Language Toolkit) in python. Additionally, the process involves converting all letters to a lower case, then lemmatizing each word.

4.2 Feature sets

Since we're interested in how different features perform on different categories of text length, we carry out experiments per feature (TF_IDF, Bleu metric, sen2vec and N-gram overlap) and with combinations on the original dataset, modified dataset, and the sub-datasets that consist of different text sample lengths.

4.3 Baselines

The ground truth on Webis-CPC-11 dataset is determined by [8], where the Precision (P) is 81, Accuracy (Acc) is 84, and Recall (R) is 90. Although F1 is not reported, we calculate it by equation (2), which results in a value of 85 for F1. These results are obtained in [8] by feeding 10 different metrics as features to the k-nearest neighbour machine learning algorithm.

$$F1 = \frac{2PR}{P+R} \qquad (2)$$

Table 1. The experiment's result, bold font represents the highest accuracy and f1 in each feature

Dataset	Bleu		TF-IDF		Sen2vec		Ngram_overlap		All Features	
evaluation	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Webis-CPC-11	57	72	87	82	64	63	56	68	66	65
Webis-CPC-21	77	87	78	87	80	88	83	90	79	88
Short text	77	82	78	87	79	88	77	82	79	88
Mid length	**81**	**89**	**83**	**90**	**84**	**91**	**85**	**91**	**85**	**91**
Long text	73	84	73	84	75	89	78	86	75	85
MSRP	67	80	71	80	72	82	69	81	71	81

4.4 Result and Discussion

Our result in table 1 outperforms the baseline system of Webis-CPC-11 dataset which is the main dataset for this study, thanks to its variety of text length and the paraphrase type that is applied on its samples. Our accuracy and F1 are 80 and 88, respectively, which outperform baseline results by 3% in terms of F1 score on Webis-CPC-21 for Sen2Vec feature set. While the result on WebisCPC-11 is more by 3% for accuracy and less by 3% F1 with implementing one feature which is TF-IDF, rather than using 10 different metrics as the baseline system did.In addition, the efficiency of the ML and DL models could be improved when the length of the text is neither short nor long. Thus, the features engineers have to consider also must include the text length when extracting these features from text segments.

Our results on each feature on MSRP are worse than the ones achieved on the short text category, which potentially indicates that the good results we have achieved on Webis-CPC-21 are because the length of the samples in MSRP is even shorter than the short text category of Webis-CPC-21. Various previous studies applied a pre-trained word2vec with cosine similarity or soft cosine on MSRP [27]. In this work, we convert each piece of text into one vector by summing up all word vectors in the text. This means we consider the semantic substance of the text to represent the overall text meaning. This clearly brings high accuracy and F1 results on Webis-CPC-21. In general, the Sen2vec yields the highest F1 score on most of the categories and surprisingly N-gram overlap performs much better than expected.

5. CONCLUSION

In this study, we answered RQ1 and RQ2 by investigating how the length of a text effects the model results in terms of measuring the semantic similarity of two different texts and which features work better with sentence, paragraph, and passage length levels. From the present experiment's results, we show that paragraph length level can convey the semantic meaning of natural language text better than sentence or passage length levels.

Based on the results, we plan to build a new passage-level- paraphrasing dataset that consist of a paragraph-length level to

achieve our contribution. Then involve the state-of-the-art transformer models to detect paraphrasing.

6. REFERENCES

[1] Aouicha, M. B., Taieb, M. A. H., and Hamadou, A. B. (2018). SISR: System for integrating semantic relatedness and similarity measures. *Soft Computing*, 22(6), 1855-1879

[2] Arase, Y., and Tsujii, J. (2021). Transfer fine-tuning of BERT with phrasal paraphrases. *Computer Speech & Language*, *66*, 101164.

[3] Bach, N. X., Le Minh, N., and Shimazu, A. (2014). Exploiting discourse information to identify paraphrases. *Expert Systems with Applications*, 41(6), 2832-2841

[4] Bär, D., Zesch, T., and Gurevych, I. (2012). Text reuse detection using a composition of text similarity measures. *Proceedings of COLING 2012*, 167-184.

[5] Barrón-Cedeño, A., Vila, M., Martí, M. A., and Rosso, P. (2013). Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4), 917-947.

[6] Bhagat, R., and Hovy, E. (2013). What is a paraphrase?. *Computational Linguistics*, 39(3), 463472.

[7] Bollegala, D., Matsuo, Y., and Ishizuka, M. (2007, April). An integrated approach to measuring semantic similarity between words using information available on the web. In Human Language Technologies 2007: *The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference* (pp. 340-347).

[8] Burrows, S., Potthast, M., and Stein, B. (2013). Paraphrase acquisition via crowdsourcing and machine learning. *ACM Transactions on Intelligent Systems and Technology (TIST),* 4(3), 1-21.

[9] Cordeiro, J., Dias, G., and Brazdil, P. (2007, March) a. A metric for paraphrase detection. In *2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07)* (pp. 7-7). IEEE.

[10] Cordeiro, J., Dias, G., and Brazdil, P. (2007) b. New functions for unsupervised asymmetrical paraphrase detection. *Journal of Software*, *2*(4), 12-23.

[11] Das, D., and Smith, N. A. (2009, August). Paraphrase identification as probabilistic quasi-synchronous recognition. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP (pp. 468-476

[12] Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[13] Dolan, W., Quirk, C., Brockett, C., and Dolan, B. (2004). Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources.

[14] Ehsan, N., Shakery, A., and Tompa, F. W. (2019). Cross-lingual text alignment for fine-grained plagiarism detection. *Journal of Information Science*, *45*(4), 443-459

[15] Fernando, S., and Stevenson, M. (2008, March). A semantic similarity approach to paraphrase detection. In *Proceedings of the 11th annual research colloquium of the UK special interest group for computational linguistics* (pp. 45-52).

[16] Ferreira, R., Cavalcanti, G. D., Freitas, F., Lins, R. D., Simske, S. J., and Riss, M. (2018). Combining sentence similarities measures to identify paraphrases. *Computer Speech & Language*, 47, 59-73.

[17] Hunt, E., Janamsetty, R., Kinares, C., Koh, C., Sanchez, A., Zhan, F., ... and Oh, P. (2019, November). Machine learning models for paraphrase identification and its applications on plagiarism detection. In *2019 IEEE International Conference on Big Knowledge (ICBK)* (pp. 97-104). IEEE

[18] Ji, Y., and Eisenstein, J. (2013, October). Discriminative improvements to distributional sentence similarity. In *Proceedings of the 2013 conference on empirical methods in natural language processing* (pp. 891-896).

[19] Kenter, T., and De Rijke, M. (2015, October). Short text similarity with word embeddings. In *Proceedings of the 24th ACM international on conference on information and knowledge management* (pp. 1411-1420).

[20] Larock MH, Tressler JC, and Lewis CE (1980) Mastering effective English. Copp Clark Pitman, Mississauga

[21] Li, H., and Xu, J. (2014). Semantic matching in search. *Foundations and Trends in Information retrieval*, *7*(5), 343-469.

[22] Maurer, H. A., Kappe, F., and Zaka, B. (2006). Plagiarism-A survey. *J. Univers. Comput. Sci.*, *12*(8), 1050-1084.

[23] Muangprathub, J., Kajornkasirat, S., and Wanichsombat, A. (2021). Document plagiarism detection using a new concept similarity in formal concept analysis. *Journal of Applied Mathematics*, *2021*.

[24] Nguyen, H. T., Duong, P. H., and Cambria, E. (2019). Learning short-text semantic similarity with word embeddings and external knowledge sources. *Knowledge-Based Systems*, *182*, 104842.

[25] Ul-Qayyum, Z., and Altaf, W. (2012). Paraphrase identification using semantic heuristic features. *Research Journal of Applied Sciences, Engineering and Technology*, *4*(22), 4894-4904

[26] Vrbanec, T., and Meštrović, A. (2020). Corpus-based paraphrase detection experiments and review. *Information*, *11*(5), 241

[27] Vrbanec, T., and Meštrović, A. (2021). Relevance of Similarity Measures Usage for Paraphrase Detection

[28] Vrublevskyi, V., and Marchenko, O. (2020, November). Paraphrase Identification Using Dependency Tree and Word Embeddings. In *2020 IEEE 2nd International Conference on Advanced Trends in Information Theory (ATIT)* (pp. 372-375). IEEE.

[29] Wan, S., Dras, M., Dale, R., and Paris, C. (2006, November). Using dependency-based features to take the'para-farce'out of paraphrase. In *Proceedings of the Australasian language technology workshop 2006* (pp. 131-138)

[30] Wang, X., Li, C., Zheng, Z., and Xu, B. (2018, July). Paraphrase recognition via combination of neural classifier and keywords. In *2018 International Joint Conference on Neural Networks (IJCNN)*(pp. 1-8). IEEE.

[31] Yang, Y., Yih, W. T., and Meek, C. (2015, September). Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 conference on empirical methods in natural language processing* (pp. 2013-2018).

[32] Yin, W., Schütze, H., Xiang, B., and Zhou, B. (2016). Abcnn: Attention-based convolutional neural network for modeling sentence pairs. *Transactions of the Association for Computational Linguistics*, *4*, 259-272

PRESENTATION LETTER

Arwa Al saqaabi
1st year PhD student at
Durham university
arwa.alsaqaabi@durham.ac.uk

PhD supervisor(s):

1- Dr Craig Stewart
https://www.durham.ac.uk/staff/craig-d-stewart/

2- Dr Eleni Akrida
www.durham.ac.uk/staff/eleni-akrida/

3- Professor Alexandra Cristea
https://www.durham.ac.uk/staff/alexandra-i-cristea/

Title: A Paraphrase Identification Approach in Paragraph length texts

I mainly work on Paraphrase identification downstream task as it important for other tasks such as plagiarism detection. These tasks could be solved by machine learning and deep learning approaches. The existing work done on sentence-length level and sentence-level paraphrasing while I focused on paragraph text length and passage-level paraphrasing.

In this year (1st year of my PhD), I implemented a study on how the length of text impact the model's results and submitted as a short paper for this conference. The experiment's results clearly show that the paragraph length level provides semantic meaning better than sentence or passage length levels. To expand my research and contribution, I plan to construct a dataset considering the text length and paraphrasing type to be in paragraph length level and passage-level paraphrasing as the existing dataset represent sentence-level paraphrasing.

I am seeking behind your recommendation, comment, opinion, and advice such on how to expand my research and What tool or approach I have to consider.