Intelligent System for Detecting Plagiarism by Obfuscation in Academic Works (ISDP).

Abstract:
Distance learning, particularly through the new generations of Open Digital Universities (DOUNG) [1], faces the samechallenges in terms of the integrity of university documents (articles, dissertations, etc.). Traditional methods, based on textual comparison, are unable to detect intelligent reformulations. This paper evaluates the contribution of a domain adaptation of a Sentence-BERT (Bidirectional Encoder Representations from Transformers) model, by contrastive fine-tuning on plagiarized source/passage pairs from the PAN (Plagiarism Analysis, Authorship Identification, and Near-Duplicate Detection) corpus, within a hybrid pipeline combining Winnowing fingerprints and semantic similarity. Through a controlled ablation study, we isolate the contribution of three levers, fine-tuning, similarity threshold and segment fusion, and show that enriched fine-tuning (multi-level and hard negative pairs) improves, with identical configuration and test set, the recall on paraphrases, and almost doubles the overall accuracy. Our approach is based on a deep semantic analysis rather than a simple textual match, thus allowing us to respond effectively to sophisticated forms of plagiarism. Domain adaptation by contrastive fine-tuning of Sentence-BERT

***Key words**:- Plagiarism detection, DOUNG, contrastive fine-tuning, Sentence-BERT, Obfuscation*

…….....

Introduction:-

Plagiarism can be defined as the use of an original work without mentioning its author or as the act of attributing another person's work to oneself. This phenomenon, amplified by the high digitization of content and the easier access to online resources, threatens scientific integrity and compromises the fundamental values of learning. Therefore, the implementation of a plagiarism detection system appears to be a necessity to preserve the originality of students' work and to maintain the credibility of the diplomas awarded. Lexical fingerprint-based methods, such as Winnowing [4], effectively detect literal copying and light obfuscation, but fail on strong paraphrasing, where vocabulary and word order are profoundly reworked. Semantic representations from sentence models such as Sentence-BERT [3] offer a complementary lead: they bring together two formulations with similar meanings regardless of their lexical surface. However, a generic model is not specialized in the characteristics of plagiarized texts. We study the hypothesis that domain adaptation by contrastive fine-tuning, trained on pairs of passages linked by plagiarism, improves the detection of obfuscated cases. Our contribution is twofold: (i) a rigorous experimental evaluation, by evaluating, by means of comparative experiments, the contribution of fine-tuning as well as its contribution on performance according to the different parameters of the processing chain; (ii) an analysis of the factors influencing the ability of the system to detect truly plagiarized passages, which can be observed thanks to an evaluation metric (the recall), resulting in a prioritized progression roadmap. The measurements relate to the PAN-PC-11 corpus, which is a corpus of data developed and used as part of the PAN 2011 scientific campaign, organized during the CLEF (Conference and Labs of the Evaluation Forum) conference, and the PlagDet metric (Plagiarism Detection Score), which is a metric that measures the overall performance of an automatic plagiarism detection system [1].

The remainder of this paper is organized as follows. Section II presents a review of the literature on plagiarism detection. Section III provides an overview of similarity detection methods and the various similarity measures employed in the field. Section IV describes Transformer architecture, while Section V introduces SBERT technology. Section VI details the proposed architecture and algorithm developed for this study. Section VII

UNDER PEER REVIEW IJAR

presents and discusses the experimental results, and Section VIII concludes the paper and outlines potential directions for future research.

Related works: -

The authors of [5] and [8] present two studies on the evolution of plagiarism detection methods, based respectively on the analysis of 41 and 189 scientific publications. They examine lexical, syntactic, semantic and machine learning and deep learning approaches. Their analyses highlight the limitations of traditional methods in the face of reformulated or translated texts and highlight the interest of hybrid approaches to improve plagiarism detection.

In [10], the authors propose T-SRE (*Transformer-based Semantic Relation Extraction*), a method based on Transformers to detect plagiarism by reformulation. It combines syntactic analysis, named entity recognition and contextual semantic representations. Evaluated on the Udacity dataset, it obtained an accuracy of 92%, a recall of 89%, an F1-score of 90.5% and a granularity of 1.28, showing the interest of combining syntactic and semantic information.

In [11], the authors develop a system combining lexical (Jaccard and TF-IDF) and semantic (Doc2Vec and SBERT) methods. The source documents constituting the corpus of data are automatically retrieved from Google, Bing and Yahoo search engines. The experiments carried out on ten (10) suspect documents show better performance of lexical methods than of semantic methods in the context studied. This work nevertheless underlines SBERT's interest in the analysis of semantic similarity.

In the paper [2], the authors propose a method combining the vector model (SVM) and the Jaccard coefficient to improve the detection of plagiarism on the PAN corpus evaluated in the framework of the CLEF. The results show that this combination improves performance compared to the individual use of these methods, particularly in terms of recall, precision and granularity. This approach highlights the interest of combining several similarity measures to improve the detection of plagiarized passages.

Materials and Methods: -

In this study, we propose a hybrid approach to plagiarism detection based on a combination of the Winnowing algorithm and a fine-tuned SBERT model. This approach utilizes a pipeline architecture consisting of interconnected, complementary modules that jointly leverage both the lexical and semantic characteristics of texts. It aims to improve plagiarism detection performance, addressing both direct textual similarities and more complex forms of reformulation and semantic paraphrasing.

Plagiarism detection methods

In this part we will discuss the different methods of plagiarism detection. Similarity between sentences occupies a very important place in Natural Language Processing (NLP) applications. We can mention two types of similarity between sentences: lexical similarity and semantic similarity.

Lexical similarity: Lexical similarity consists of comparing each word or each phase character by character. The expected result is within the range of [0,1] or can be expressed as a percentage.

Semantic similarity: Semantic similarity consists of comparing sentences and detecting whether they have the same meaning, i.e. evaluating the (semantic) proximity of their meaning. In [9], the authors present three different types of semantic similarities. They are essentially based on corpus, knowledge and strings.

Existing similarity measure: There are several equations that can be used to measure the similarity between sentences. We will mention a few:

UNDER PEER REVIEW IN IJAR

- ✓ **Jaccard similarity** given by the formula below:

Jaccard's similarity measures the proportion of common words in relation to the set of distinct words in two sentences. For instance, for the two preprocessed sentences (Tokenization):

A = {Student, submits, his, dissertation, for, a, plagiarism, check}

B = {Student, submits, his, document, for, plagiarism, check}

$$J(A, B) = \frac{| A \cap B |}{| A \cup B |}$$

With: A ∩ B = 6 (The set of words common to the two sentences).A ∩ B

A ∪ B = 9 (The set of all the distinct words of the two sentences)

$$J(A, B) = \frac{6}{9} = 0{,}6667$$

66, 67 % similarity

Dice coefficient: The Dice coefficient is identical to the Jaccard similarity, but it gives double weight to the intersection of the two sets.

$$\text{Dice}(A, B) = \frac{2 * | A \cap B |}{(|A| + |B|)} = \frac{12}{15} = 0{,}8$$

80 % similarity

- ✓ **Cosine similarity:** Unlike the two previous methods, cosine similarity compares two numerical vectors by measuring the angle between them.

Its formula is: $$Sim = \frac{(A.B)}{(||A|| \times ||B||)}$$

Suppose that A and B are represented by the two vectors: A= (1,1,1,1,0,1,1,1,1) and B = (1,1,1,0,1,1,0,1,1)

$A.B = 6$; $||A|| = \sqrt{8}$ And $||B|| = \sqrt{7}$

So,

$$Sim = \frac{6}{\sqrt{56}} = 0{,}8018$$

80, 18 %Similarity

Transformer architecture

In [12] the authors proposed the transform architecture. The transformer processes all the elements of the sequence simultaneously thanks to the attention mechanism.

For example, suppose the sentence: **The student submits his dissertation for plagiarism checking.**

UNDER PEER REVIEW IN IJAR

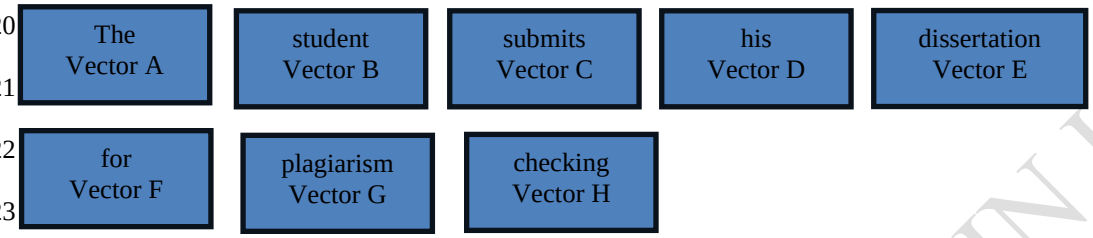
✓ **Step 1: Tokenization**

Thanks to Tokenization, this sentence will be cut into Tokens (words).

The - student – submits – his/her – dissertation – for – plagiarism – checking

✓ **Step 2: Initial Vectorization**

Each token receives a digital representation



✓ **Step 3: Parallel Calculations in the Transformer Layer**

From the initial representations, the Transformer performs all the necessary calculations for each token simultaneously. The self-attention mechanism comes into play at this stage. The self-attention mechanism allows each word to consider the other words in the sentence to better understand its context.

For example, to interpret the word "**dissertation**", the Transformer can consider:

- "**Student":** indicates that the person concerned is a student.
- "**Submits**": indicates that the work is being submitted or uploaded.
- "**his**": establishes a possessive relationship with the dissertation.
- "**plagiarism checking**": provides the context in which the dissertation is being used.

This will make it possible to represent this word taking into account the context in which it appears. Unlike recurrent networks (RNN, LSTM, GRU) which process sequences sequentially.

Taking the same sentence, here is the fundamental difference:

1. Treats the "**student**"
2. Milks "**submits**" taking into account the previous state
3. Treats "**his**"
4. Treatise "**dissertation**"
5. Continue until "**checking**"

Each step depends on the state conveyed by the previous step.

Architecture and Proposed Algorithm

Hybrid Discovery Pipeline Architecture

UNDER PEER REVIEW IN IJAR

The proposed architecture is based on a hybrid approach consisting of a set of interconnected modules that work in the form of a pipeline for better plagiarism detection, including forms of reformulation and semantic paraphrases. This architecture iscomposed of :

Document acquisition module

This module receives the documents to be compared (source document and suspect document).Let be a corpus of N source documents. N = | D|. Each document is seen as an ordered sequence of sentences.

$$D = \{d_1, \ldots\ldots d_N\}$$

Let us consider a suspicious document whose sentences are segmented into m sentences. d_S

$$\boldsymbol{d_S = (S_1, \ldots . S_m)} \quad \textbf{(1)}$$

Each source document for (1 ≤ i ≤ N) is segmented into sentences:$d_i n_i$

$$\boldsymbol{d_i = \left(t_{i,1}, \ldots\ldots t_{i,n_i}\right)} \quad \textbf{(2)}$$

We gather all the source sentences in a single set T:

$$\boldsymbol{\mathcal{T} = \left\{t_{i,j} : 1 \leq i \leq N, 1 \leq J \leq n_i\right\}} \quad \textbf{(3)}$$

Word preprocessing module

After receiving the documents, a pre-processing follows through the Spacy tool by applying the following operations: cleaning, and segmentation by sentence.

Candidate Source Selection Module

Winnowing identifies candidate sources for each suspect document that share at least a minimum number of k-gram fingerprints and delineates areas that may contain plagiarism.

(Common seeds). For the source document d_i and the suspect document, we consider the set of seeds as being the set of pairs of positions where the same imprint appears in both documents:d_s

$$\boldsymbol{g_i = \{(p, q) : \exists h, h \in H(d_s) \text{ à } la\ position\ p, h \in H(d_i) \text{ à } la\ position\ q\}.} \quad \textbf{(4)}$$

(Candidate area). From the common seeds, we propose a spatial grouping rule to construct candidate areas. We then define the coverage of a sentence by a zone by means of the overlap of their intervals.

A zone for source i is a pair of intervals

$$\boldsymbol{z = (I_z^s, I_z^{src}), \text{ with and } I_z^s \subseteq [0, |d_s|) I_z^{src} \subseteq [0, |d_i|)}, \text{ such that the region} \quad \textbf{(5)}$$

concentrates at least $\boldsymbol{(I_z^s, I_z^{src})}$**i** spatially contiguous seeds (deviation bounded by a margin). Wenote:$\boldsymbol{g_i}$

$$\boldsymbol{Z(d_s) = \bigcup_{i=1}^{N} \{z : z\ is\ an\ area\ of\ the\ source\ i\}} \quad \textbf{(6)}$$

The set of all the fields of the suspect document. A source i is a candidate if it has at least one zone [4].

UNDER PEER REVIEW IN IJAR

(Coverage of a suspicious sentence). A suspicious sentence is covered by a z-zone S_k if its interval overlaps with the suspicious region of *z:*

$$\boldsymbol{couvre}(\boldsymbol{z}, \boldsymbol{S_k}) \Leftrightarrow [\boldsymbol{a_k}, \boldsymbol{b_k}) \cap \boldsymbol{I_z^s} \neq \emptyset. \quad \textbf{(7)}$$

Winnowing locates the alleged plagiarism as zones, both in the suspect document and in the source. The semantics are then calculated only for the suspect sentences covered by an area, which severely restricts the space for comparison.

Semantic Threshold Verification

(Pairing and detection in an area). So be it S_k a sentence covered by a zone *z* de source i = σ(*z*). We retain the most similar source sentence and its score:

$$\boldsymbol{j_{k,i}^*} = \arg\max \boldsymbol{sim}(\boldsymbol{S_k}, \boldsymbol{t_{i,j}}), \qquad \boldsymbol{.\rho_{k,i}} = \boldsymbol{sim}\,(\boldsymbol{S_k}, \boldsymbol{t_{i,j_{k,i}^*}}) \quad \textbf{(8)}$$

The couple (S_k, i) produces elementary detection if $\rho_{k,i} \geq \mathrm{T}$, where T is the similarity threshold. The set of elementary detections is:

$$\boldsymbol{\Delta = \{(k, i, ,[a_k, b_k)\rho_{k,i}) : \exists z \in Z(d_s), couvre(z, S_k), \sigma(z) = i, \rho_{k,i} \geq \mathrm{T}\}.} \quad \textbf{(9)}$$

Only sentences covered by a zone are submitted to the semantic judge. The decision is based solely on the cosine similarity and the threshold T.

Semanticrepresentation module

Sentences in candidate areas are transformed into embeddings using the SBERT model.

(Semantic embedding): a function φ: X $\rightarrow R^d$ associates to any sentence x a normalized vector φ(x),$\|\phi(x)\|_2 = 1$.

Similarity Calculation Module

The vectors of the suspect sentences are compared to those of the candidate source sentences by the cosine similarity, to identify the closest matches. The similarity of two sentences is the cosine:

$$\mathbf{sim}(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{\phi}(\boldsymbol{x}), \boldsymbol{\phi}(\boldsymbol{y}) \rangle \, \boldsymbol{\epsilon}[-\mathbf{1}, \mathbf{1}]$$

The cosine similarity is a dot product because we have to make one of the normalized vectors. φ is the Sentence-Bert model. [3]

Decision Module

At this level, we use an *S threshold*to determine whether a sentence is plagiarized or not.

We set the threshold *S* = 0.75.

Yes:

$$\mathbf{sim}(\boldsymbol{x}, \boldsymbol{y}) > \mathbf{0.75}$$

Then the sentence is considered plagiarized.$\boldsymbol{x}$

Blending and Filtering Module

UNDER PEER REVIEW IN IJAR

Neighboring detections pointing to the same source are repasted into coherent passages (fusion), reducing fragmentation. Anti-isolated filtering then discards short and isolated segments with a moderate score, which typically constitute false alarms.

Resultsgeneration module

The detected passages are formalized in localized annotations, specifying for each suspicious passage the corresponding source and their positions, allowing the evaluation of the detection.

This architecture allows us to better maintain our system and add new methods or modules. The base model is all-mpnet-base-v2, derived from the MPNet architecture.

Domain adaptation by contrastive fine-tuning

Fine-tuning does not modify the architecture: it moves the representations in the vector space in such a way as to bring the pairs closer together (source, plagiarized passage) and to distance the unrelated pairs. Our enhanced modelproduces pairs with two levels granularities (integer passage and aligned sentences) and adds to each anchor a hard negative — a source sentence thematically close but not related — obtained by searching in the source space. The training is based on 7,370 pairs of sentences and uses the MultipleNegativesRankingLoss [13], which exploits both the explicit and implicit negatives of the batch.

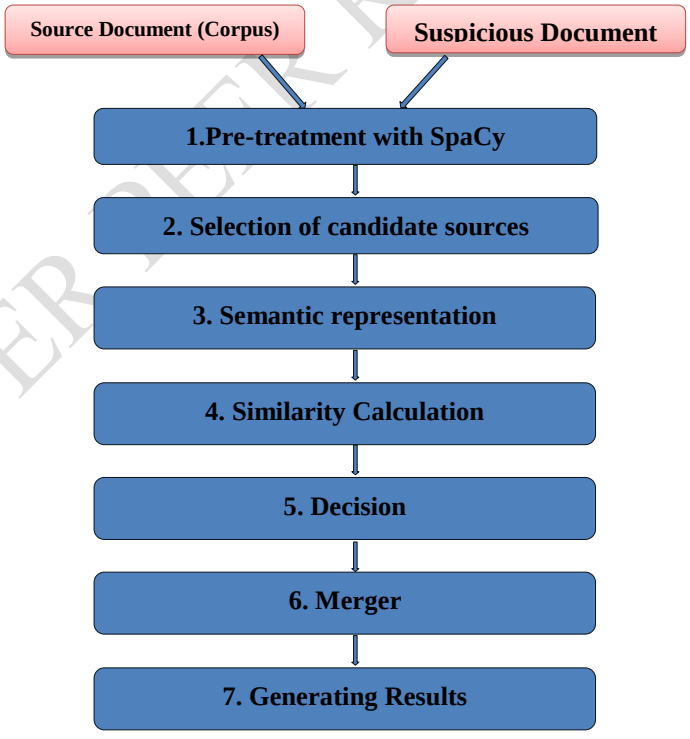


Figure 1 : General architecture of the proposed architecture

UNDER PEER REVIEW IN IJAR

Results and Discussion: -

The experiments are based on the PAN Text Alignment corpus. The training partition and the test partitions are strictly disjointed at the level of the suspect document, guaranteeing the absence of leakage: no document used for the evaluation contributed to the fine-tuning. The metrics are those of PAN [1]: precision and recall at the character level, granularity (average number of detections per case), and plagdet = F1 / $\log_2(1 + \text{granularity})$. The results are broken down by type of obfuscation; the analysis focuses on the artificial type (monolingual obfuscation), which is the majority and statistically significant.

- **Precision**

VP: True Positive
FP: False Positive

$$Precision = \frac{VP}{VP + FP}$$

Accuracy reflects our system's ability to detect cases of plagiarism accurately and unambiguously.

- **Recall**

VP: True Positive
FN: False Negative

$$Recall = \frac{VP}{VP + FN}$$

After implementing our algorithm using the Python language, we tested our model on a corpus composed of 11093 source documents and 200 suspect documents. To optimize the computation time for similarity detection, we processed the 11093 source documents using the Spacy word processing tool and then we calculated and stored for each source document and for each sentence its embedding in a folder that we named document_source_embedding. We loaded the folder containing the 200 suspicious files, each file being processed with Spacy and segmented into sentences. Each sentence is transformed into a vector using fine-tuned SBERT for a comparison with the vectors of all the source files (11093 vectorized source files). The central result is established on a test set of 200 documents (987 cases, including 708 artificial and 158 translations), aligned and strictly disjointed from the training. The basic model and the fine-tuned model are compared in a rigorously identical configuration (threshold 0.75); only the model/embeddings pair changes. Any difference is therefore attributable to fine-tuning alone.

After calculating the following metrics — precision and recall — we obtained the following overall averages:

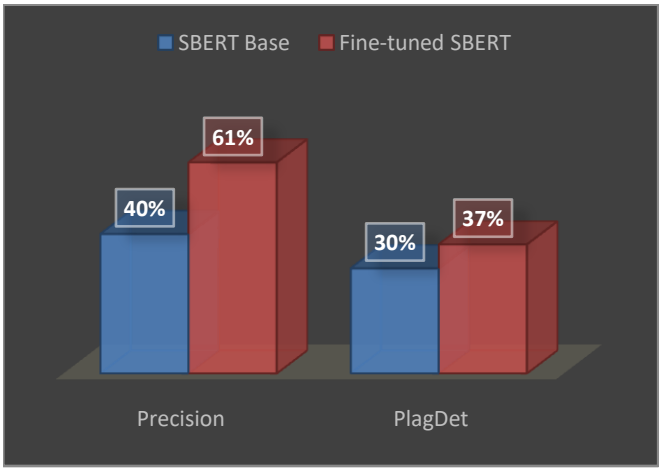


Figure 3 : Comparison between the Base model and the fine-tuned model at the global level

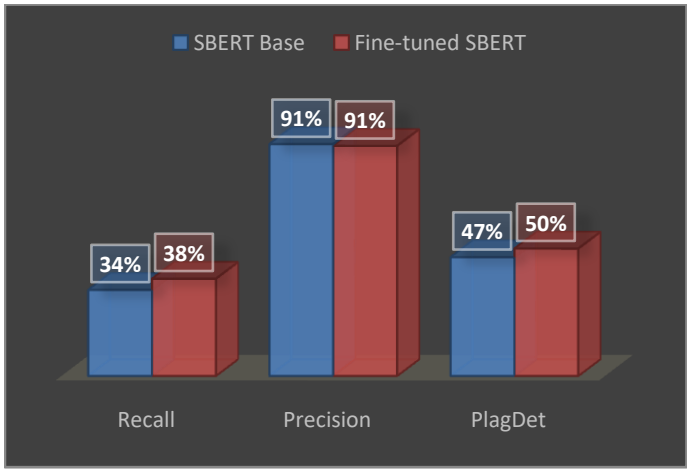


Figure 2 : Comparison between the Base model and the fine-tuned model at the artificial level

PEER REVIEW IN IJAR

Fine-tuning raises the recall on cases of artificial plagiarism from 0.339 to 0.383 at constant precision. On a global scale, the fine-tuned model produces fewer detections (562 vs. 775) but these are more reliable, as shown by the increase in precision from 0.401 to 0.607. The overall PlagDet score also increases from 0.302 to 0.371. These results show that the model's adaptation improves its ability to identify artificial reformulations while reducing erroneous detections. It thus makes it possible to obtain a better compromise between recall and precision, despite a decrease in the total number of passes detected.

Conclusion: -

This study allowed us to highlight the limitations of the basic SBERT model in the field of plagiarism. The proposal of a model adapted to plagiarism by contrastive fine-tuning improves the detection of plagiarism by monolingual obfuscation. The artificial recall goes from 0.339 to 0.383 at constant precision and brings the overall result from 0.401 to 0.607, an increase of 0.206 by reducing false alarms: adaptation improves both sensitivity and specificity.

References: -

[1]. B. T. Abdoul Aziz, M. I. Tiado, N. A. Hassane, M. Harouna, " Simulation Performance Evaluation of the Synchronous Course Transcription System in the DOUNG (Digital Open Universities - New Generation) Ad Hoc Network ", Proceedings of the International Conference on Electrical, Computer and Energy Technologies (ICECET 20-22 July 2022) Prague-Czech Republic, INSPECT membership number:22028564, DOI**:** 10.1109/ICECET55527.2022.9873049,https://ieeexplore.ieee.org/document/9873049/

[2]. Shuai Wang, Haoliang Qi, Leilei Kong, Cuixia Du: "Combining VSM and Jaccard Coefficient for External Plagiarism Detection," Proceedings of the 2013 International Conference on Machine Learning and Cybernetics, Tianjin, July 14-17, 2013.

[3]. Nils Reimers, Iryna Gurevych, "Sentence-BERT: Embeddings of Sentences Using BERT Siamese Networks," Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th Joint International Conference on Natural Language Processing, pages 3982–3992, Hong Kong, China, November 3–7, 2019. 2019 Association for Computational Linguistics.

[4] Schleimer, S., Wilkerson, D. S., Aiken, A. (2003). Selection: local algorithms for document fingerprinting. Proceedings of the ACM SIGMOD 2003.

[5] Altynbek Amirzhanov, Cemil Turan and Alfira Makhmutova, Types of Plagiarism and Detection Methods: A Systematic Study of Algorithms in Text Analysis, Frontiers in Computer Science, vol. 7, article 1504725, 2025. DOI: 10.3389/fcomp.2025.1504725

[6]. CNRS Ethics Committee (COMETS). "Ethical reflection on plagiarism in scientific research", Opinion n°2017-34, 27/06/2017.

[7]. Ferric C. Fang, R. Grant Steen, and Arturo Casadevall, "Misconduct accounts for the majority of retired scientific publications," edited by Thomas Shenk, Princeton University, Princeton, NJ, and approved September 6, 2012 (received for review July 18, 2012).

[8] Muhammad Sajid, Muhammad Sanaullah, Muhammad Fuzail, Tauqeer Safdar Malik, and Shuhaida Mohamed Shuhidan, Comparative Analysis of Text-Based Plagiarism Detection Techniques, PLOS ONE, Vol. 20, No. 4, Item E0319551, 2025. DOI: 10.1371/journal.pone.0319551

UNDER PEER REVIEW IN IJAR

[9]. Nora Raju T, Rahana P A, Raichel Moncy, Sreedarsana Ajay, Sindhya K Nambiar, "Sentence Similarity – A State-of-the-Art Approach," 2022 International Conference on Computing, Communication, Security and Intelligent Systems (IC3SIS) | 978-1-6654-6883-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/IC3SIS54991.2022.9885721.

[10] Pon Abisheka, C. Deisy, and P. Sharmila, T-SRE: Transformer-Based Semantic Relationship Extraction for Contextual Paraphrased Plagiarism Detection, Journal of King Saud University – Computer and Information Sciences, Vol. 36, No. 10, Article 102257, December 2024. DOI: 10.1016/j.jksuci.2024.102257

[11] Ayoub Ali M. Saeed and Alaa Yaseen Taqa, Using Retrieved Sources for Semantic and Lexical Plagiarism Detection, Iraqi Journal of Science, Vol. 64, No. 6, pp. 3136–3152, 2023. DOI: 10.24996/ijs.2023.64.6.41.

[12]. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, Illia Polosukhin, "Attention is All You Need," 31st Neural Information Processing Systems Conference (NIPS 2017), Long Beach, CA, USA.

[13]. Henderson, M., Al-Rfou, R., Strope, B., Sung, Y.-H., Lukács, L., Guo, R., Kumar, S., Miklos, B., & Kurzweil, R. (2017). Effective naturallanguageresponse suggestion for intelligent response. arXiv: 1705.00652. DOI:10.48550/arXiv.1705.00652

UNDER PEER REVIEW IN IJAR