Journal of Informatics and Web Engineering
Vol. 4 No. 3 (October 2025) eISSN: 2821-370X

Lightweight String Similarity Approaches for Duplicate Detection in Academic Titles

Fahrudin Mukti Wibowo[1*], Muhammad Zidny Nafan[2], Muhamad Azrino Gustalika[3], Harinda Fernando[4], Muhammad Hussain[5], Nur Afiqah Binti Sahadun[6]

[1,2,3]Faculty of Informatics, Telkom University, Jalan D.I Panjaitan no. 128 Purwokerto, Indonesia.
[4]Faculty of Computing, Sri Lanka Institute of Information Technology, Malabe, Sri Lanka.
[5]Faculty of Engineering & Technology, University of Sindh, Hosho Rd, Jamshoro, Pakistan.
[6]Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Jalan Delta 1/6, 86400 Parit Raja, Johor, Malaysia.
**corresponding author: (fahrudinw@telkomuniversity.ac.id; ORCiD: 0000-0001-7681-5255)*

Abstract - This study addresses the critical challenge of detecting duplicate final year project (FYP) titles in academic institutions, where minor variations like reordering, synonyms, and paraphrasing often obscure plagiarism. We systematically evaluate four string similarity algorithms - Jaro-Winkler, Levenshtein Edit Distance, TF-IDF with Cosine Similarity, and Jaccard Similarity - using a synthetic dataset of 250 title pairs representing common duplication patterns. Our experiments reveal that character-based methods (Jaro-Winkler and Edit Distance) achieve perfect detection (F1-score=1.0) for literal matches, including typographical variations and phrase reordering. At the same time, TF-IDF demonstrates strong semantic capability (F1-score=0.95), albeit with some false positives. Jaccard Similarity performs poorly (Recall=0.40) due to its inability to handle paraphrased content. The analysis of score distributions show a clear separation between duplicates and non-duplicates for character-based approaches, compared to significant overlap in set-based methods. Based on these findings, we propose a practical two-stage screening framework: initial high-confidence filtering using Jaro-Winkler (threshold>0.9) followed by semantic validation with TF-IDF (threshold>0.8). This hybrid approach offers institutions an effective balance between accuracy and computational efficiency for title screening. This study contributes by demonstrating how existing string similarity techniques can be orchestrated into a lightweight, two-stage screening framework tailored for academic title duplication, balancing accuracy with deployment feasibility in institutional settings. Future work should explore multilingual extensions and validation with real-world title datasets to further enhance the practical applicability of these findings.

Keywords— Duplicate Detection, String Similarity, TF-IDF, Lightweight NLP, Hybrid Models

Received: 1 May 2025; Accepted:12 August 2025; Published: 16 October 2025

Journal of Informatics and Web Engineering
https://doi.org/10.33093/jiwe.2025.4.3.25

Published by MMU Press. URL: https://journals.mmupress.com/jiwe

1. INTRODUCTION

The rise of digital communication and widespread access to academic content have increased the likelihood of both intentional and unintentional duplication in scholarly work, including FYP titles. While full-text plagiarism detection has been extensively studied—particularly for essays and source code—the challenge of identifying semantic similarity in short academic texts, such as titles, remains underexplored [1], [2], [3]. Titles often undergo minor modifications such as reordering, synonym substitution, or paraphrasing, which may obscure underlying similarity [4], [5].

This issue poses practical challenges for institutions aiming to uphold academic integrity and efficiently manage project supervision. Manual screening is often time-consuming and prone to inconsistency. Automated tools that rely solely on surface-level comparisons are limited in detecting semantically equivalent but lexically different titles [6]. For example, consider the two titles: "A Mobile App for Mental Health Monitoring" and "Mental Health Tracker Using Smartphone Application." Though the wording differs significantly, the core idea remains the same. Traditional string-matching techniques may fail to identify such titles as duplicates due to superficial lexical differences.

The most recent short text duplication detection methods have tried to combine the character-based methods with semantic analysis. For instance, methods like Levenshtein Distance [7], [8], Jaro-Winkler [9], [10], and statistical models such as TF-IDF [11], [12], as, well as topic modelling techniques like LDA [13], [14], [15]. [16] proposed an approach that uses FastText embeddings along with TF-IDF weighting for identifying subtle paraphrasing in academic writing—paving the way for hybrid techniques to effectively deal with reworded or semantically equivalent expressions in both low-resource settings. However, no existing study has validated lightweight string similarity models specifically on short academic titles such as FYP titles

While the individual methods explored are well-established, their integrated use for short-title screening in an academic quality assurance context remains underexplored. Our contribution lies not in algorithmic novelty but in applying a calibrated hybrid design—optimized for semantic robustness, runtime efficiency, and practical deployment. In this context, this paper looks into several string similarity methods including Jaro-Winkler, Edit Distance, TF-IDF, and Jaccardin order to understand their performance in the context of detecting semantically similar FYP titles. We compare them across detection accuracy, semantic coverage, and computational efficiency, aiming to propose lightweight solutions suitable for real-time academic screening. On the larger scale, we aim to find operational solutions that compromise between performance and efficiency of these complex methods and can be used in academia in real-time.

2. LITERATURE REVIEW

2.1 Traditional and Hybrid String Similarity Methods

String similarity algorithms have been widely applied in text mining, record linkage, and plagiarism detection. Classical techniques such as Levenshtein Distance, Jaccard Similarity, and cosine similarity with TF-IDF representations are well-known for their simplicity and computational efficiency. [17] proposed a hybrid model that integrates approximate string-matching techniques with TF-IDF and cosine similarity to enhance plagiarism detection. Their approach demonstrated increased Precision and Recall compared to standalone methods. In a related study, [18] developed a web-based plagiarism detection system using TF-IDF and cosine similarity, which outperformed basic substring algorithms like Rabin-Karp. Their tool showed high consistency between automated and manual evaluations, effectively capturing nuanced term-level similarities in short academic texts.

Despite the popularity of models based on the vector space or bag-of-words, they often fall short in capturing semantic equivalence, particularly in cases involving rewording or paraphrasing. [19] addressed this limitation by employing Word2Vec embeddings to cluster words into semantic groups, improving the detection of disguised plagiarism. However, the added semantic sensitivity comes with computational costs, making such approaches less suitable for lightweight applications such as FYP title comparison.

2.2 Semantic Embedding and Lightweight Models

Hybrid strategies are particularly valuable in academic environments where title duplication may occur through subtle variations such as word reordering or synonym replacement. [16] proposed two hybrid plagiarism detection models combining FastText word embeddings with TF-IDF. Their system employed a two-stage filtering process—Bag-of-

Words for document-level filtering and semantic similarity scoring— achieving a reported Precision of 95.1%. However, Recall and F1-score were not reported, and the method's semantic embedding layer likely introduced moderate computational overhead—raising questions about suitability for lightweight applications.

[20] also underscored the applicability of string-based metrics by using Jaccard and overlap coefficients to cluster malware binaries. While their domain was cybersecurity, the methodological approach offers transferable insights for detecting structural similarity in short texts.

[21] advanced the field further by introducing short-text similarity models that integrate semantic and syntactic information. Their approaches—KEBERT-GCN and CPT-TK—combined BERT embeddings with graph convolutional networks and tree kernels. Although these models outperformed standard BERT on benchmark datasets, their reliance on external linguistic resources and computational complexity make them impractical for lightweight academic screening. Moreover, most studies emphasize Precision alone, with limited discussion of Recall, F1-score, or runtime—metrics critical for real-time academic screening tools.

2.3 Research Gap and Motivation

The reviewed literature suggests that while deep learning models offer high semantic accuracy, they often require substantial computational resources. Conversely, traditional string similarity methods, though efficient, struggle with paraphrased or semantically altered texts. A middle ground lies in hybrid lightweight models that combine character-level metrics, statistical representations, and semantic embeddings. These models promise scalability, interpretability, and practical relevance—especially in academic contexts like FYP title screening, where phrase variation is often subtle yet meaningful.

This study builds on these insights to evaluate selected string similarity algorithms for short academic texts. The goal is to identify lightweight, accurate methods suitable for real-time deployment in academic environments. While previous studies have proposed various methods, their results vary in scope, metrics, and practical feasibility. Table 1 below summarizes these methods based on key dimensions including evaluation metrics, dataset context, and suitability for real-time academic screening.

Table 1. Summary of Related Methods with Reported Metrics

Study	Method(s) Used	Dataset / Context	Metrics Reported	Lightweight Suitability
[11]	TF-IDF + Edit Distance	Student essay plagiarism	Precision, Recall	Partial – moderate cost
[12]	TF-IDF + Cosine	Web-based title detection	Accuracy	Good – web-optimized
[10]	FastText + TF-IDF	PAN-PC-11 dataset	Precision (95.1%)	Limited – semantic layer costly
[14]	Jaccard, Overlap	IoT Malware (non-academic)	Custom score clustering	Yes – transferable method
[15]	BERT + GCN/Tree kernels	Benchmark NLP datasets	F1-score, Accuracy	No – high computational load

3. RESEARCH METHODOLOGY

3.1 Dataset Construction

To compare the performance of string similarity algorithms in identifying duplicated academic project titles, a synthetic sample consisting of 250 FYP title pairs was created. This method allowed for controlled experimentation and also prevented privacy issues on real student input. These artificial titles were generated to follow the most common copying and pasting phenomena in academia, including titles with lexical deformation (e.g., synonym replacement, such as "AI-Based" and "Machine Learning-Based"), structural deformation (e.g., phrase-level reordering, like "Chatbot for Education" and "Educational Chatbot"), and semantic deformation (e.g., titles with close portrayal of meaning but different wording such as "IoT Home Automation" and "Smart Home System Using IoT"). The data set contained various areas of computing, such as artificial intelligence, web systems, software engineering,

so that the evaluation was representative of real diversity in academic project titles. The dataset has been completely synthesized; however, its synthetic nature constrained its use. For example, although the test questions were designed to be analogous to real student work, they may not be as inclusive of the range of linguistic variation and creativity found in student work. The study also only considered titles in English between 5–15 words, so the findings cannot be generalized to longer or indeed multilingual text.

While synthetic datasets offer control and allow systematic variation testing, we acknowledge that they lack the linguistic diversity, originality, and edge-case phrasing often found in authentic student submissions. To strengthen the applicability and credibility of our findings, future work will focus on validating the proposed methods against real FYP title data collected from academic repositories, subject to institutional privacy policies and data-sharing agreements. Such validation would enable a more comprehensive understanding of model robustness in practical academic settings.

A preliminary analysis suggested that some algorithms showed a bias depending on the type of variation; e.g., although Jaro-Winkler worked well for typographical changes, it performed less well for semantic paraphrasing whereas TF-IDF + Cosine similarity would be less affected by word-order changes but less robust to minor character-level mismatches

3.2 Preprocessing Pipeline

For comparison, all titles were subjected to a light text normalization process to make them more uniform as well as to reduce noise before applying similarity methods. First, titles were lower-cased to eliminate case-sensitivity. Tokenization then divided titles into word units, which were further filtered to exclude common stop words (e.g., “for”, “the”, “and”) that are devoid of content. Finally, the Porter stemming algorithm was applied to reduce the lemmatized words to their stem (e.g., “automation” changed to “automat”). This step partially normalizes across morphological variability (e.g., “detection” vs. “detecting”) on similarity scores. Though this pipeline made comparisons across titles easier, once we start to bring in these various levels of processing, drawbacks become possible as well: aggressive stemming could risk conflating semantically distinct terms (e.g., “universe” and “university” both stem to “univers”), stop word removal might obscure valuable context in some edge cases (e.g., “The Effect of AI” vs. “Effect of AI” might erroneously be declared the same). The Porter Stemmer was chosen because it is lightweight, widely used, and performs well on general English text. It strikes a good balance between reducing words to their root forms without being too aggressive, unlike Lancaster. Snowball is more refined but heavier, which is less ideal for lightweight applications. To illustrate the preprocessing pipeline, Table 2 shows how a sample project title is transformed through lowercasing, stop word removal, and stemming.

Table 2: Example of Title Transformation Across Preprocessing Steps

Preprocessing Step	**Example Title**
Original	“An IoT-Based Smart Farming System for Rice Monitoring”
Lowercased	“an iot-based smart farming system for rice monitoring”
Stop word Removal	“iot-based smart farming system rice monitoring”
Porter Stemming	“iot-bas smart farm system rice monitor”

3.3 Algorithm Selection and Implementation

We chose four string similarity algorithms to represent the different computational approaches, from a character-level similarity to a semantic-aware method:

- TF-IDF + Cosine Similarity: A term-based approach adapted from information retrieval (IR) that summarizes word importance using inverse document frequency (IDF) and calculates similarity by the cosine distance between title vectors. It is good at detecting similarity in topic but does not consider word order and syntactic structure.
- Jaro-Winkler Distance: A character-based metric that gives credit for common characters at the start, in addition, it weighs second characters higher than the first common characters (useful in detecting names with

a smaller number of character changes but relocation of characters). But it fares poorly on semantically equivalent and lexically dissimilar titles.

- Levenshtein Edit Distance: Another character-based approach that measures the minimum number of insertion, deletion, or substitution operations required to transform a title into another. Though efficient for small-scale variations (e.g., misspellings), it is computationally costly for long texts and lacks effectiveness in capturing semantic similarity.
- Jaccard Similarity: A set-based similarity measure that estimates the intersection of word sets of two titles. It is good for similarity at the surface level, but not good for fine-grained semantic relationships, especially if synonyms or paraphrasing exist.

All algorithms were implemented by python libraries (e.g., scikit-learn for TF-IDF, jellyfish for Jaro-Winkler) via default parameters for their reproducibility.

3.4 Proposed Two-Stage Detection Framework

To operationalize the string similarity methods into a practical screening tool, we designed a hybrid two-stage detection framework that leverages the strengths of both character-based and semantic approaches. In the first stage, Jaro-Winkler rapidly identifies high-confidence duplicates based on surface-level string similarity. Pairs exceeding a similarity score of 0.90 are flagged immediately. In the second stage, pairs with lower scores undergo deeper semantic comparison using TF-IDF with cosine similarity, which captures paraphrased or reworded titles more effectively. This tiered design balances speed with semantic sensitivity, reducing false positives while keeping computational overhead minimal. The complete structure of the hybrid detection process is illustrated in Figure 1.

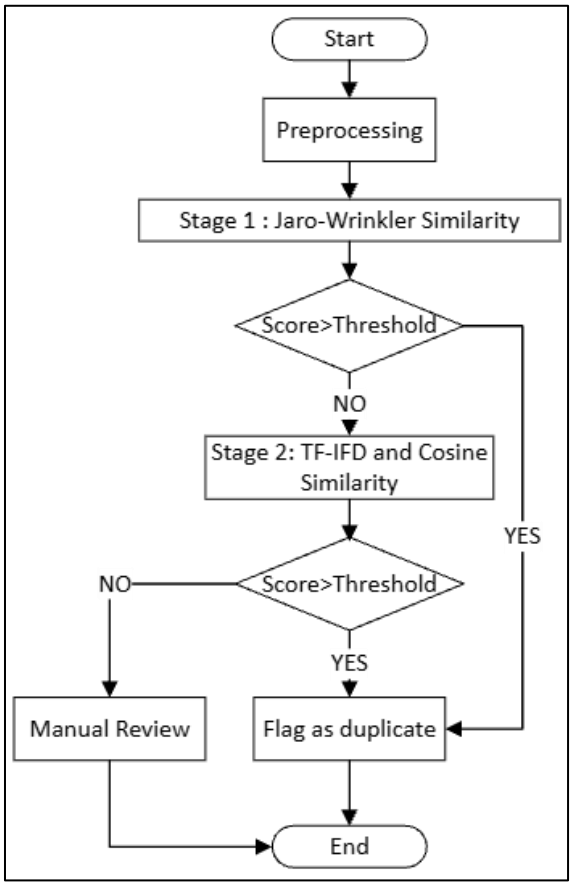

Figure 1. Two-stage Hybrid Duplicate Detection Framework Combining Jaro-Winkler and TF-IDF

4. RESULTS AND DISCUSSIONS

Comparison of the four string similarity algorithms is conducted systematically from three aspects, namely, classification metrics, score distributions and algorithmic bias analysis. Combined, these visualizations provide a multi-view perception of duplicate detection performance for academic project titles.

4.1 Classification Performance

As shown in Figure 2, both Jaro-Winkler and Edit Distance achieved perfect classification results (Precision = Recall = F1-score = 1.0) at the 0.70 similarity threshold. This demonstrates the effectiveness of character-level models in handling:

- Variant spellings (e.g., “Optimization” vs. “Optimisation”)
- Phrase reordering (e.g., “AI for Healthcare” vs. “Healthcare AI Applications”)

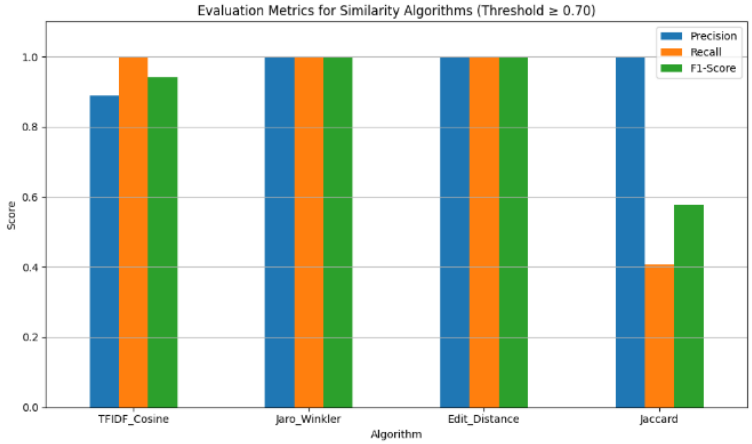

Figure 2. Evaluation Metrics (Precision, Recall, and F1-score) for Four String Similarity Algorithms

TF-IDF + Cosine produced strong results (F1-score = 0.95), with slightly lower Precision (0.90) due to false positives where titles share keywords but differ in concept. Jaccard Similarity, on the other hand, struggled with semantic variation—its Recall dropped to 0.40 despite high Precision, confirming its weakness in handling reworded or paraphrased content.

TF-IDF + Cosine achieved competitive but imperfect results (F1-score = 0.95), with Precision (0.90) constrained by false positives in titles sharing technical terminology but differing conceptually - a limitation visually corroborated by Figure 2's score distribution. Jaccard Similarity's poor Recall (0.40) in Figure 2 aligns with its known sensitivity to token overlap rather than semantic meaning, as further evidenced in Figure 3's overlapping quartile ranges.

4.2 Threshold Validation

The boxplot in Figure 3 illustrates the distribution of similarity scores for both duplicate (1) and unique (0) title pairs across all algorithms. Two notable patterns emerge:

- Jaro-Winkler and Edit Distance show a clear separation between duplicates and unique titles, validating their suitability for high-confidence classification at the 0.70 threshold. The threshold of 0.70 was heuristically determined based on the observed separation in score distributions of Figure 3 and preliminary F1-score maximization trials across a range of values.
- TF-IDF + Cosine displays partial overlap, especially in the 0.4–0.6 range for unique titles, which explains its occasional false positives and moderate Precision.

This distribution helps clarify the trade-off observed in Figure 2: while TF-IDF + Cosine captures thematic similarity, it benefits from secondary filtering to handle borderline cases.

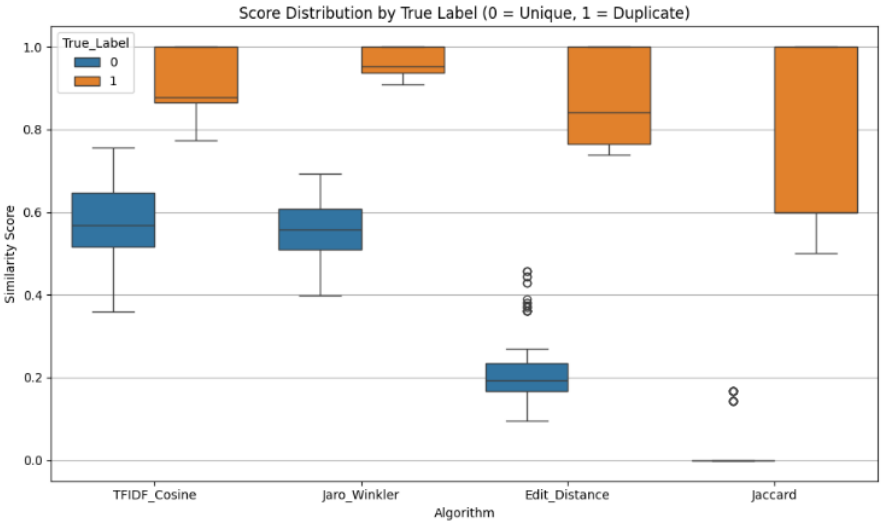

Figure 3. Score Distributions By Algorithm and Label (0 = Unique, 1 = Duplicate)

This distribution explains the Precision-Recall trade-off visible in Figure 2, particularly for TF-IDF + Cosine. The histogram's right skew among duplicates confirms the algorithm's strength in detecting thematic similarity, while the left-tail false positives underscore the need for supplemental character-based verification.

4.3 Algorithmic Bias Analysis

The boxplot in Figure 4 illustrates the similarity score distributions across algorithms, grouped by true labels (duplicate = 1, unique = 0).

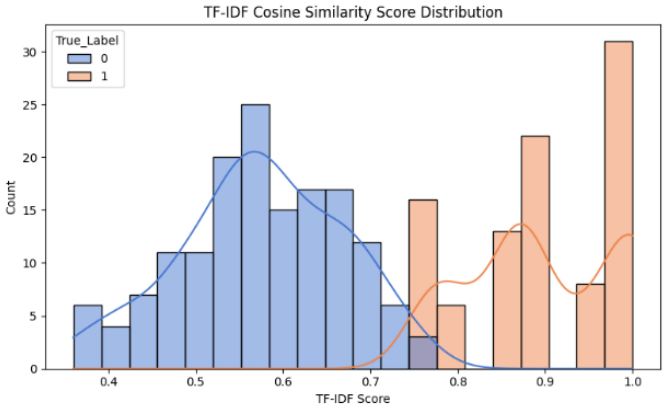

Figure 4. Boxplot of Similarity Score Distributions by Algorithm and True Label (0 = unique, 1 = duplicate)

This visualization highlights how different algorithms behave in distinguishing between semantically similar and dissimilar title pairs. A tabular interpretation of the interquartile ranges (IQR) from the plot supports three main conclusions derived from earlier classification metrics:

- Character-based methods demonstrate strong binary separation.
  Both Jaro-Winkler and Edit Distance exhibit tightly clustered scores for duplicate pairs (mostly between 0.95–1.0) and distinctly lower scores for unique pairs (below 0.10). The lack of overlap between these ranges confirms their reliability in exact or near-exact match detection.

- TF-IDF + Cosine shows a wider, yet informative distribution.
  Although TF-IDF presents broader IQRs for both classes (approximately 0.55–0.75 for non-duplicates and 0.78–0.95 for duplicates), the general separation remains effective. However, the mild overlap introduces some ambiguity, explaining its slightly lower Precision observed in Figure 1.
- Jaccard Similarity exhibits poor discriminatory power.
  Jaccard's IQRs for duplicates and non-duplicates largely intersect (roughly 0.25–0.82 for duplicates and 0.10–0.58 for non-duplicates), suggesting limited effectiveness in distinguishing between reworded or paraphrased titles. This overlap aligns with the algorithm's lower Recall and F1-score.

The interquartile overlap in Figure 4 directly correlates with the Recall limitations observed in Figure 2, providing a coherent explanation for algorithmic strengths and weaknesses across different types of title variations. This reinforces the suitability of character-level methods for high-Precision filtering and the potential of TF-IDF as a secondary semantic validator.

4.4 Synthesis of Findings

When contextualized within the literature discussed in Section 2, the findings reveal three key insights for practical implementation:

- Jaro-Winkler is optimal for exact or near-exact duplicate detection.
  Its consistent perfect scores (Precision = Recall = F1-score = 1.0) and tight score distribution (Figure 3) make it suitable as a first-pass filter for high-confidence matches.
- TF-IDF with a stricter threshold (≥ 0.80) provides semantic coverage.
  Although TF-IDF has a wider score range and slight overlap between duplicate and non-duplicate classes, it captures semantic rewordings better than character-level methods. However, it requires calibration to reduce false positives.
- A hybrid, tiered detection strategy is most effective.
  Institutions can leverage score distribution gaps from Figure 3 to design a two-stage system:
  a. Apply Jaro-Winkler to flag titles with scores > 0.90 as confirmed duplicates.
  b. Use TF-IDF for ambiguous pairs scoring between 0.70–0.90.
  c. Refer borderline or conflicting cases to manual review

This layered approach balances semantic sensitivity and computational efficiency, addressing the Precision-Recall trade-offs observed in Figures 2–4. It also aligns with the lightweight, interpretable framework advocated in the literature, offering a scalable solution for academic environments with limited resources. A basic runtime test was conducted on a standard laptop (Intel i5, 8GB RAM). On average, Jaro-Winkler and Edit Distance processed 250 title pairs in under 2 seconds, while TF-IDF took around 5 seconds. Jaccard was the fastest but least accurate. These results support the practical use of character-based methods in low-resource environments.

To strengthen deployment relevance, we envision this framework as a lightweight batch-screening module integrated into a university's academic management system. Given the low computational footprint—under 2 seconds per 250 title comparisons on a standard laptop—this system could be executed periodically during project registration cycles. Future work will explore integrating this into a web-based dashboard or plugin using Flask or FastAPI, enabling real-time screening via RESTful API calls while logging similarity scores and audit trails for review.

5. CONCLUSION

This study demonstrates that lightweight string similarity algorithms—particularly character-based methods like Jaro-Winkler and Edit Distance—achieve exceptional performance in detecting near-duplicate academic project titles, while TF-IDF + Cosine offers valuable semantic coverage despite higher false positive rates, as evidenced by the classification metrics in Figure 1 and score distributions in Figure 2. The clear separation between duplicate and non-duplicate titles in Figure 3's boxplot analysis further validates the superiority of character-based approaches for institutional deployment where computational efficiency and Precision are prioritized. However, limitations persist in handling multilingual titles and deeply paraphrased content, highlighting the need for future work integrating FastText embeddings for cross-lingual generalization and distilled BERT variants for nuanced semantic matching. Practical

implementation should focus on developing hybrid systems that begin with high-confidence filtering using character-based methods, followed by semantic refinement for borderline cases. This tiered approach balances Precision and semantic sensitivity while remaining computationally efficient making it highly suitable for academic institutions with limited resources. By enabling fast, accurate, and low-cost screening of project titles, this work empowers academic institutions to scale supervision efforts, maintain academic standards, and reduce manual oversight burdens—ultimately strengthening institutional integrity and operational efficiency.

ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their valuable comments.

FUNDING STATEMENT

The authors received no funding from any party for the research and publication of this article.

AUTHOR CONTRIBUTIONS

Fahrudin Mukti Wibowo: Supervision, Conceptualization, Methodology, Writing – Original Draft Preparation;
Muhammad Zidny Nafan: Data Collecting, Data visualization, Preprocessing;
Muhamad Azrino Gustalika: Data Collecting, Writing – Review & Editing;
Harinda Fernando: Modelling with Semantic Embedding and Lightweight, Evaluation models;
Muhammad Hussain: Modelling with Semantic Embedding and Lightweight, Evaluation models;
Nur Afiqah Binti Sahadun: Project Administration, Writing – Review & Editing.

CONFLICT OF INTERESTS

The authors declare no conflict of interest.

ETHICS STATEMENTS

This study did not involve human participants, animal experiments, or data from social media platforms, and therefore no ethical approval or informed consent was required. Our publication ethics follow The Committee of Publication Ethics (COPE) guideline. https://publicationethics.org/.

REFERENCES

[1] D. Prakoso, A. Abdi, and C. Amrit, "Short text similarity measurement methods: a review", *Soft Computing*, vol. 25, no. 6, pp. 4699-4723, 2021, doi:10.1007/s00500-020-05479-2.

[2] M. Han, X. Zhang, X. Yuan, J. Jiang, W. Yun, and C. Gao, "A survey on the techniques, applications, and performance of short text semantic similarity", *Concurrency and Computation: Practice and Experience*, vol. 33, no. 5, 2020, doi: 10.1002/cpe.5971.

[3] J. Gatto, O. Sharif, P. Seegmiller, P. Bohlman, and S. M. Preum, "Text encoders lack knowledge: Leveraging generative LLMs for domain-specific semantic textual similarity," *arXiv preprint arXiv:2309.06541*, 2023.

[4] T. Celikten, and A. Onan, "Exploring text similarity in human and AI-generated scientific abstracts: A comprehensive analysis," in *IEEE Access,* vol. 13, pp. 74313-74334, 2025, doi: 10.1109/ACCESS.2025.3564867.

[5] C. Zhou, C. Qiu, L. Liang, and D. Acuna, "Paraphrase identification with deep learning: A review of datasets and methods", *IEEE Access*, vol. 13, pp. 65797-65822, 2025, doi:10.1109/access.2025.3556899.

[6] Z. Amur, Y. Hooi, H. Bhanbhro, K. Dahri, and G. Soomro, "Short-text semantic similarity (STSS): Techniques, challenges and future perspectives", *Applied Sciences*, vol. 13, no. 6, pp. 3911, 2023, doi:10.3390/app13063911.

[7] J. Zhang, L. Qian, S. Wang, Y. Zhu, Z. Gao, H. Yu, and W. Li, "A Levenshtein distance-based method for word segmentation in corpus augmentation of geoscience texts," *Annals of GIS*, vol. 29, no. 2, pp. 293–306, 2023, doi: 10.1080/19475683.2023.2165543.

[8] Y. Chaabi and F. A. Allah, "Amazigh spell checker using Damerau-Levenshtein algorithm and N-gram," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, pp. 6116–6124, 2022, doi:10.1016/j.jksuci.2021.07.015.

[9] O. Rozinek, and J. Mares, "Fast and precise convolutional Jaro and Jaro-Winkler similarity," *2024 35th Conference of Open Innovations Association (FRUCT)*, Tampere, Finland, pp. 604-613, 2024, doi: 10.23919/FRUCT61870.2024.10516360.

[10] N. Ifada, F. Rachman, and S. Wahyuni, "Character-based string matching similarity algorithms for Madurese spelling correction: A preliminary study," in *2023 International Conference on Electrical Engineering and Informatics (ICEEI)*, pp. 1–6, 2023, doi: 10.1109/ICEEI59426.2023.10346716.

[11] L.-C. Chen, "An extended TF-IDF method for improving keyword extraction in traditional corpus-based research: An example of a climate change corpus," *Data & Knowledge Engineering*, vol. 153, pp. 102322, 2024, doi: 10.1016/j.datak.2024.102322.

[12] S. M. M. Hossain, K. M. A. Kamal, A. Sen, and I. H. Sarker, "TF-IDF feature-based spam filtering of mobile SMS using a machine learning approach," in *Applied Intelligence for Industry 4.0*, Boca Raton, FL, USA: Chapman and Hall/CRC, pp. 162–175, 2023, doi: 10.1201/9781003340066-11.

[13] W. Suwarningsih, and N. Nuryani, "Generate fuzzy string-matching to build self attention on Indonesian medical-chatbot", *International Journal of Electrical and Computer Engineering (IJECE),* vol. 14, no. 1, pp. 819, 2024, doi:10.11591/ijece.v14i1.pp819-829.

[14] D. Subramanian, T. Jeyaprakash, M. Preetha, S. Ganga, and S. Sajeev, "Similarities and ranking of documents using TF-IDF, LDA and WAM", *2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS)*, pp. 01-07, 2024, doi:10.1109/adics58448.2024.10533526.

[15] A. Mishra, and V. Panchal, "A novel approach to capture the similarity in summarized text using embedded model", *International Journal on Smart Sensing and Intelligent Systems*, vol. 15, no. 1, 2022, doi:10.2478/ijssis-2022-0002.

[16] H. Arabi, and M. Akbari, "Improving plagiarism detection in text document using hybrid weighted similarity", *Expert Systems With Applications*, vol. 207, pp. 118034, 2022, doi:10.1016/j.eswa.2022.118034.

[17] Z. Balani, and C. Varol, "Combining approximate string matching algorithms and term frequency in the detection of plagiarism," *International Journal of Computer Science and Security (IJCSS),* vol. 15, no. 4, pp. 97–105, 2021.

[18] J. Halim, and D. Lasut, "Document plagiarism detection application using web-based TF-IDF and Cosine similarity methods", *Bit-Tech*, vol. 7, no. 2, pp. 202-213, 2024, doi:10.32877/bt.v7i2.1697.

[19] C. Chang, S. Lee, C. Wu, C. Liu, and C. Liu, "Using word semantic concepts for plagiarism detection in text documents", *Information Retrieval Journal*, vol. 24, no. 4-5, pp. 298-321, 2021, doi:10.1007/s10791-021-09394-4.

[20] S. Torabi, M. Dib, E. Bou-Harb, C. Assi, and M. Debbabi, "A strings-based similarity analysis approach for characterizing IoT malware and inferring their underlying relationships," in *IEEE Networking Letters*, vol. 3, no. 3, pp. 161-165, Sept. 2021, doi: 10.1109/LNET.2021.3076600.

[21] Y. Zhou, C. Li, G. Huang, Q. Guo, H. Li, and X. Wei, "A short-text similarity model combining semantic and syntactic information", *Electronics*, vol. 12, no. 14, pp. 3126, 2023, doi:10.3390/electronics12143126.

BIOGRAPHIES OF AUTHORS

	**Fahrudin Mukti Wibowo** is a lecturer at Department of Informatics Engineering, Faculty of Informatics, Telkom University. He received his master's degree at Universitas Gadjah Mada, Indonesia. He is currently pursuing his Doctoral Degree in Information Technology at Universiti Tun Hussein Onn, Malaysia. His research interests are Internet of Things and deep learning for data sensor. He can be contacted at email: fahrudinw@telkomuniversity.ac.id.
	**Muhammad Zidny Nafan** is a lecturer at Department of Informatics Engineering, Faculty of Informatics, Telkom University. He received his bachelor's degree at Islamic State University of Syarif Hidayatulloh Jakarta and Master's at Universitas Indonesia. He is currently pursuing his Doctoral Degree in Computer Science at Universitas Gadjah Mada, Yogyakarta, Indonesia. His research interests are natural language processing and deep representation learning for closed-domain corpus. He can be contacted at email: muhammadn@telkomuniversity.ac.id.
	**Muhamad Azrino Gustalika** is a lecturer in the Department of Informatics, Faculty of Informatics, Telkom University. He earned his bachelor's degree at Muhammadiyah University of Malang and earned his master's degree at Malang State Polytechnic. His research interest is in the field of image processing and AR/VR. He can be contacted at email: azrino@telkomunivesity.ac.id.
	**Harinda Fernando** is a senior academic with over 18 years of experience in tertiary education across Australia and Sri Lanka. His expertise spans Cyber Security, Networking, and Machine Learning, with a strong background in technical, research, and teaching domains. Over the past 13 years, he has supervised more than 250 undergraduate and 50 postgraduate dissertations, reflecting his deep commitment to academic mentorship and student development. He can be contacted at email: harinda.f@sliit.lk.
	**Muhammad Hussain** is an Assistant Professor in the Department of Information Technology, Faculty of Engineering and Technology, University of Sindh, Jamshoro. He holds a Master's Degree in Computer Information Engineering from Mehran University of Engineering and Technology and is a registered engineer with the Pakistan Engineering Council (PEC). With over five years of experience in teaching and research at the undergraduate level, he has successfully supervised 10 students. His research interests include machine learning, deep learning, and programming. He can be contacted at email: mh.pirzada@usindh.edu.pk.
	**Nur Afiqah Binti Sahadun** received the Diploma in Information Technology from Universiti Tun Hussein Onn Malaysia in 2008, and subsequently earned the Bachelor's, Master's, and Ph.D. degrees in Computer Science from Universiti Teknologi Malaysia in 2011, 2014, and 2023, respectively. She currently serves as a Senior Lecturer (DS13) at Universiti Tun Hussein Onn Malaysia. Her research interests include forensic computing, bioinformatics, feature selection, and artificial intelligence for digital forensics and data-driven analysis. She can be contacted at email: nurafiqah@uthm.edu.my.