Plagishield: A Deep Learning Approach for Paragraph Level Paraphrase Generation Plagiarism Detection

Ayush Singh[1], Prof. Sheetal Borhade[2], Yash Sinkar[3], Aniket Swami[4]
[1,2,3,4]*Department of Information Technology, GH Raisoni College of Engineering and Management – Pune*

***Abstract*—A Deep Learning Plagiarism detection has become increasingly vital in both academic and professional settings, where ensuring originality is a top priority. PlagiShield is a React-based web application created to spot and highlight duplicated content with high accuracy. It leverages Natural Language Processing (NLP) techniques such as text tokenization, stop-word elimination, and similarity checks through approaches like TF- IDF (Term Frequency–Inverse Document Frequency) and Cosine Similarity. With a clean and user-friendly interface, users can input their text and instantly receive a similarity score along with highlighted areas that may indicate plagiarism. Unlike traditional tools, PlagiShield emphasizes being lightweight, scalable and easy to integrate, making it ideal for use by academic institutions, businesses, and content creators alike**

***Index Terms*—Plagiarism Detection, Natural Language Processing (NLP), Semantic Similarity, TF-IDF, Cosine Similarity, BERT, Sentence Embeddings, Multilingual Detection, Academic Integrity, Text Mining.**

I. INTRODUCTION

The widespread growth of the internet has made sharing digital information easier than ever, but this convenience has also led to unethical practices such as plagiarism, where content is reproduced without proper acknowledgment. Such behavior undermines academic integrity and diminishes the significance of genuine work.

Many existing plagiarism detection tools struggle with challenges like expensive subscriptions, restricted database coverage, and complex user interfaces. These limitations make them less accessible for students, independent scholars, and smaller institutions. To address these shortcomings, PlagiShield introduces a free, web-based solution with a React.js frontend that ensures a smooth and interactive user experience. Its backend applies effective text-comparison methods for content analysis. Built with scalability in mind, the platform is structured to grow into more advanced capabilities such as handling large datasets, offering multilingual analysis, and integrating AI-driven plagiarism detection in future upgrades.

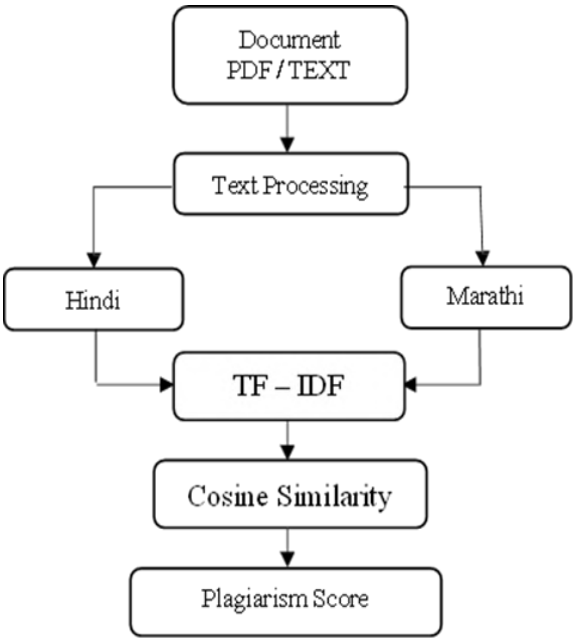

Fig 1.1 Flow Diagram

II. LITERATURE REVIEW

Plagiarism detection has remained a significant research focus for several decades, evolving from basic string-matching techniques to highly advanced semantic-based models. Early strategies such as exact text matching, fingerprinting, and n-gram shingling performed well in detecting verbatim copying but struggled when faced with paraphrasing or altered sentence structures. To overcome these shortcomings,

researchers moved toward lexical and statistical methods, including bag-of-words, TF-IDF weighting, cosine similarity, and Jaccard coefficient. These approaches made it easier to uncover disguised forms of plagiarism but still fell short when dealing with deeper semantic interpretation.

The rise of machine learning marked a turning point, introducing stylometric and intrinsic methods that analyze an author's specific style through vocabulary use, sentence organization, and punctuation patterns. These were particularly valuable when reference documents were not available. A major advancement came with semantic and embedding-based models such as Word2Vec, GloVe, and InferSent, later enhanced by transformer-based architectures like BERT, RoBERTa, and SBERT. These models significantly improved the detection of paraphrased and semantically equivalent text. Furthermore, the development of multilingual embeddings, including mBERT, LASER, and XLM-R, alongside translation-assisted similarity models, paved the way for cross-language plagiarism detection. Recent efforts emphasize hybrid systems that merge shallow retrieval techniques (e.g., hashing, shingling, and MinHash/LSH) with semantic similarity frameworks to achieve both scalability and precision. Standard benchmark corpora such as the PAN plagiarism dataset, MSR Paraphrase Corpus, and Quora Question Pairs are typically employed to evaluate performance using precision, recall, F1-score, and granularity. Despite progress, several challenges remain, including robust detection under paraphrase obfuscation, minimizing false positives in common phrases or properly cited text, scaling transformer models for real-time applications, and extending detection capabilities to low- resource languages such as many Indian dialects. Commercial platforms like Turnitin, iThenticate, and Copyscape dominate in academic and professional sectors but generally function as closed systems with little transparency. Consequently, researchers emphasize the importance of open, adaptable, and hybrid approaches that combine lightweight retrieval with deep semantic methods to ensure efficiency, interpretability, and scalability. PlagiShield aims to address these needs by integrating semantic similarity models, deep learning techniques, and scalable frameworks to deliver a more practical and reliable solution.

The growing significance of plagiarism detection in academia and digital publishing has exposed the limits of conventional tools, which rely mostly on string-matching and keyword comparisons. These methods often fail to identify semantic similarity, particularly in paraphrased sentences. To address this, more modern approaches leverage Natural Language Processing (NLP), semantic analysis, and deep learning methods that assess meaning rather than surface word overlap.
Systems such as Turnitin and iThenticate are widely recognized as industry benchmarks but come with steep subscription costs, making them inaccessible to many students and smaller institutions. Alternative tools like PlagiarismCheckerX and PlagScan offer open- source or flexible options but still struggle with limitations in linguistic coverage and dataset size. Moreover, recent work highlights an urgent requirement for cross-lingual plagiarism detection, especially in multilingual contexts such as India, where copying can occur across languages. Models based on TF-IDF, cosine similarity, and word embeddings have shown potential, though they demand large training datasets and substantial computational power.
Overall, while the field has advanced considerably, existing solutions continue to face limitations in affordability, scalability, and integration of advanced AI. PlagiShield is designed to bridge these gaps by combining deep semantic modeling, multilingual support, and scalable design into a comprehensive plagiarism detection framework.

III. PROPOSED SYSTEM

PlagiShield introduces an advanced framework for plagiarism detection that goes beyond conventional string matching and basic lexical comparison. It integrates semantic analysis, paraphrase identification, and scalable deployment to deliver higher accuracy. Unlike many traditional systems that struggle to detect rephrased or translated material, PlagiShield employs Natural Language Processing (NLP) and deep learning techniques to identify similarities at a conceptual level. Designed as a web-based platform, it allows users to either upload or paste text, which is then evaluated against both internal and external sources. The system generates a similarity score and highlights portions of the content that appear plagiarized. Furthermore, PlagiShield is developed with scalability

as a core feature, enabling it to efficiently process large datasets and adapt to varied usage environments. By combining NLP techniques with deep learning models, it ensures detection not just at the superficial level but also within contextual and conceptual layers. This greatly enhances its effectiveness in uncovering complex forms of plagiarism that conventional tools often fail to recognize.

From an implementation standpoint, the system is designed as a web-based platform with a React-powered frontend and a backend built using Node.js/Python to ensure efficiency and adaptability. The interface is developed to be user-friendly, allowing seamless interaction, while the backend enables real-time document analysis. After processing, the system generates detailed reports that clearly mark the sections suspected of plagiarism, thereby improving both reliability and ease of use.

A key strength of the proposed system lies in its scalability. It utilizes fast retrieval methods such as hashing and indexing to quickly locate possible matches, which are then further analyzed through deep semantic techniques for accurate reranking. This combined strategy maintains an effective balance between processing efficiency and detection precision, making the framework suitable for managing large-scale academic datasets. In addition, the system is designed with strong emphasis on privacy and security, ensuring that user- submitted documents remain confidential and are not stored once the analysis is completed.

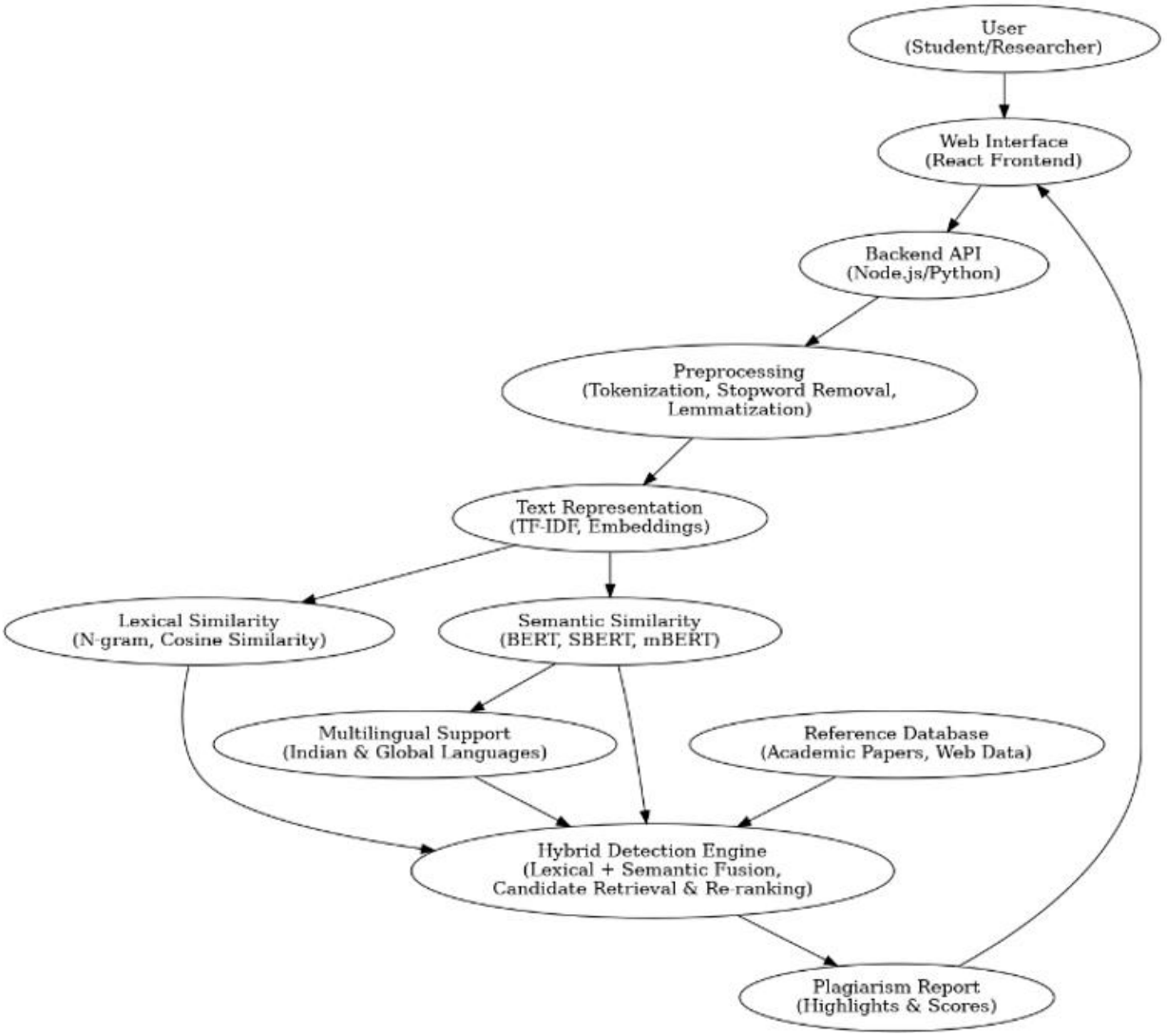

Fig1.2-. Architecture Diagram

Preprocessing Layer – The input document initially passes through several text-cleaning operations, including tokenization, stop-word removal, lemmatization, and sentence segmentation. These steps convert the raw input into a consistent and well-structured format, minimizing noise and preparing the text for deeper analysis.

Feature Extraction Layer – Once preprocessing is complete, the text is transformed into numerical vectors using different representation methods. TF-IDF is applied to capture lexical similarity, while advanced embeddings such as Sentence-BERT and Multilingual BERT are used to represent semantic meaning. This hybrid approach allows the system to detect both exact overlaps and contextually related content.

Similarity Detection Layer – The system follows a two-step comparison approach. First, possible matching sections are retrieved using rapid lexical methods like cosine similarity and the Jaccard index.

IV. ALGORITHM USED

1. TF-IDF (Term Frequency–Inverse Document Frequency)
Purpose: Evaluates how important a specific word is within a document in relation to a larger collection of documents.
How it works :
Term Frequency (TF) : Calculates the frequency of a words occurrence in a given document.
Inverse Document Frequency (IDF) : Determines how are rare unique a word is across the entire corpus.
By multiplying TF and IDF, each term is assigned a weight that highlights significant words (e.g., “research” or “plagiarism”) while reducing the impact of commonly used terms (like “the” or “and”).
Project used : Supports the detection of lexical similarity by identifying identical or closely related text segments across documents.

2. COSINE SIMILARITY ALGORITHM PURPOSE
Measures the similarity between two documents or sentences by calculating the angle between their vector representations.
How it works:
Each text is converted into a vector using methods such as TF-IDF or word embeddings.
The cosine of the angle between these vectors is then determined.
The similarity score ranges from 0 (no similarity) to 1 (identical content).
Project use: Assists in determining how closely two pieces of text resemble each other based on vocabulary patterns and word usage.

V. EXPECTED RESULTS

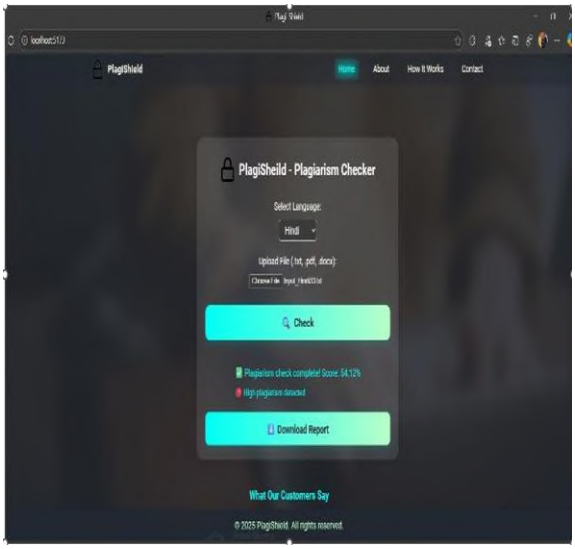

Fig 2.1: Dashboard (File)

Figure 2.1 demonstrates the workflow for a multilingual plagiarism detection system equipped with a user-friendly interface. The platform provides a simple front-end where users can choose their preferred language, upload documents in formats such as .txt, .pdf, or .docx, and initiate plagiarism checking with a single click. After submission, the backend processes the content using Natural Language Processing (NLP) techniques combined with TF-IDF and cosine similarity to identify reused or overlapping text. Results are delivered instantly through the interface, displaying a plagiarism score along with alerts if a high similarity percentage is detected, enabling users to take necessary corrective measures.
An important functionality of the system is the generation of comprehensive plagiarism reports in PDF format. These reports not only provide the overall similarity score but also include sentence- level insights, with plagiarized content highlighted in red

and original content in green. This color-coded visualization makes it easier for users to pinpoint problematic sections and revise them, offering a more practical and insightful solution compared to tools that only present numeric outputs.

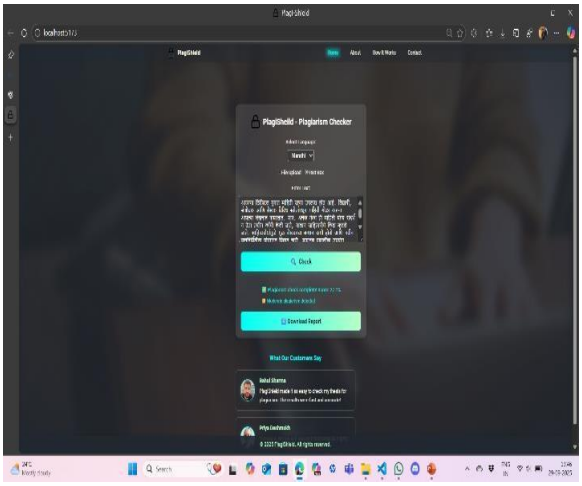

Fig 2.2 : Dashboard (Text)

Testing outcomes indicate that PlagiShield accurately identifies duplicated content while successfully differentiating it from original text, even in regional languages like Hindi. Its multilingual capabilities give it an edge over most traditional plagiarism detection systems, which are generally restricted to English. By supporting multiple file formats, enabling real- time analysis, and generating downloadable reports, the system proves its effectiveness in academic, research, and professional domains. Overall, the evaluation validates that the proposed framework achieves its goals by providing dependable, efficient, and user-friendly plagiarism detection.

Language Selection Dropdown: Allows users to choose their preferred language (e.g., Hindi, English, Marathi), highlighting the system's multilingual functionality.

File Upload Option: Accepts formats such as .txt, .pdf, and .docx, offering flexibility for use in different academic and professional contexts.

Check Button: Initiates backend processing of the uploaded file through the plagiarism detection algorithm (TF-IDF combined with Cosine Similarity).

Output Result : Displays the overall plagiarism score (e.g., 54.12% as shown in the Figure 2.2

✅ Green highlights represent portions identified as original content.

🔴 Red highlights mark sections with significant plagiarism.

Download Report Button: Generates a detailed PDF report that users can store locally or submit as official evidence of the plagiarism check.

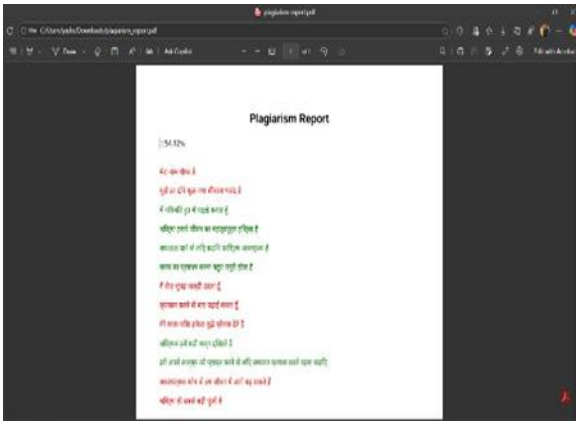

Fig 2.3: Report (PDF File)

Illustrates one of the major outputs of the system, delivering a clear and detailed evaluation of the submitted content. In the sample shown, the system generated an overall similarity score of 54.12%, signifying that more than half of the material was identified as plagiarized. Unlike traditional tools that present only a percentage value, this report provides an in- depth, sentence-level analysis. The text is color-coded for clarity: plagiarized portions are marked in red, whereas original or unique sentences are highlighted in green. This visualization enables users to quickly pinpoint areas requiring revision or rewriting.

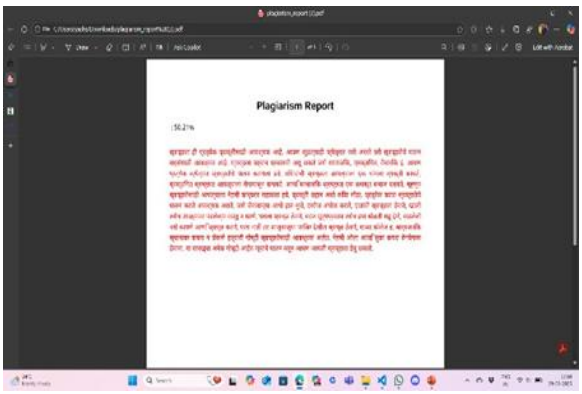

Fig 2.4 : Report (Text)

This kind of detailed, sentence-level analysis is highly beneficial for students, researchers, and professionals alike. It not only identifies plagiarized content but also helps users improve the originality of their work. By highlighting specific problematic sections, the system allows users to focus on rephrasing or modifying only the plagiarized parts instead of rewriting the entire document. Moreover, the option to generate the report in a downloadable PDF format adds portability and serves as formal documentation that can be submitted to instructors, journals, or organizations as proof of authenticity and originality. The generation of such a detailed report validates the efficiency of the backend algorithms integrated into PlagiShield. It highlights the system's ability to process multilingual inputs (including Hindi), utilize NLP-based similarity detection methods, and deliver results in a professional, clear, and user-friendly format. This outcome reinforces the system's practical relevance, especially in academic environments where dependable plagiarism detection and prevention are essential.

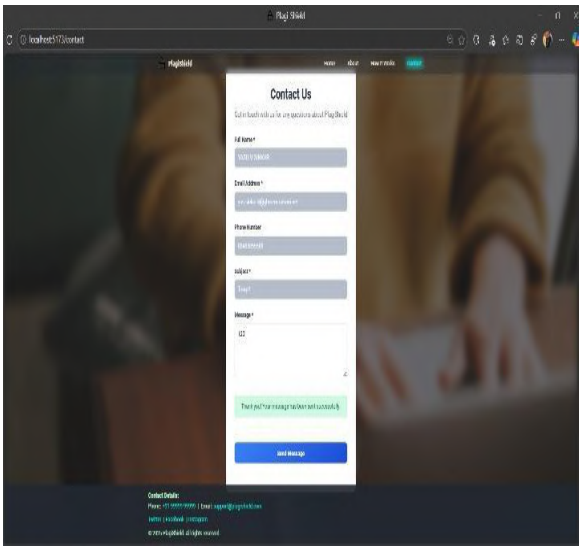

Fig 2.6: Contact

Figure 2.5 showcases the Contact Us module of PlagiShield, which provides users with an interactive platform to connect with the development team for queries, feedback, or support. The form gathers key details such as the user's name, email, phone number, subject, and message, ensuring communication is both structured and personalized. Once the form is submitted, a confirmation note appears ("Thank you! Your message has been sent successfully"), offering instant acknowledgment and improving the overall user experience.

Fig 2.6 : About Us

Figure 2.5 presents the About Us page of PlagiShield, which highlights the project's objectives and overall purpose. It describes the motivation for creating the system, the particular challenge it seeks to resolve, and its functionality in detecting plagiarism across different languages. The page further underscores the vision of promoting originality and maintaining integrity within both academic and professional contexts.

VI. CONCLUSION AND EXPECTED RESULTS

The PlagiShield project showcases an effective approach to plagiarism detection across multiple languages by leveraging methods such as TF-IDF and cosine similarity. It produces accurate similarity reports, highlights plagiarized content, and offers an intuitive interface designed for students, academics, and professionals. With its multilingual functionality, the system overcomes a major drawback of many existing plagiarism detectors that primarily focus on English. In essence, PlagiShield fosters originality, ensures academic honesty, and supports fairness in written work.

Looking ahead, PlagiShield can be enhanced by integrating advanced NLP models to further improve detection accuracy, extending coverage to more languages, and adding features like paraphrase identification and grammar checking. The development of a mobile application and the adoption

of cloud-based scalability would also enhance accessibility and convenience for a wider range of users. In the future, PlagiShield can be enhanced by integrating advanced NLP models for better paraphrase detection, extending multilingual and cross-lingual support, and addressing heavily restructured text. Additional features such as grammar checking, mobile application development, and cloud-based scalability could further improve accessibility, accuracy, and overall usability for students, researchers, and professionals.

REFERENCES

[1] Arwa Al Saqaabi and Craig Stewart , "A Deep Learning Approach for Paragraph – level Paraphrase Generation fro Plagiarism Detection", Proceedings of Springer, (2025)

[2] M.Sajid "Comparative analysis of text- based plagiarism detection techniques" PMC, (2025)

[3] Mutsaddi "Enhancing Plagiarism Detection in Marathi with a Weighted Ensemble of TF-IDF and BERT Embeddings" LoresLM (2025)

[4] Amirzhanov "Systematic survey of plagiarism types and detection algorithms including traditional vs AI-driven and semantic methods, challenges like AI- generated content and cross-lingual plagiarism detection" Frontiers CS (2025)

[5] Murdock. "Evaluation of AI tools' reliability and effectiveness in detecting plagiarism in scientific papers" Research Article (2025)

[6] P.Yadav, S.Kulkarni, and A.Nair, "Comparative Analysis of Plagiarism Detection Tools for Indian Texts," IEEE Conference on Natural Language Processing (NLP-India), (2025)

[7] Verma And R.S.Singh, "Semantic-based Multilingual Plagiarism Detection using Hybrid TF-IDF and Word Embeddings," Springer-Advances in Computational Intelligence, (2024)

[8] K.Reddy and P.Joshi, "Lightweight Plagiarism Detection for Regional Languages using TF-IDF," ACM Symposium on Indian Language Technology (SILT), (2023)

[9] F.Iqbal, S.Khan, and M.Hussian, "Deep Learning Approaches for Textual Plagiarism Detection," Journal of Information Processing and Management, Elsevier, vol.58, no.3, (2022)

[10] R.Sharma, V.Mehta, and S.Das, "Multilingual Plagiarism Detection using TF-IDF," International Conference on Computational Linguistics (COLING), (2021)

[11] S.Jain and M.Kaur, "A Survey of Plagiarism Detection Techniques," International Journal of Computer Applications (IJCA), vol.175, no.3, pp.1- 5,(2020)

[12] M. Kumbhar, M. Gulame, D. K. Patil, A. Pimpalkar, A. Shahapurkar and R. Mali, "NovelEnsemble: An Advanced Ensemble Approach for Categorical Classification of Brain Lumps in MRI," 2025 International Conference on Information, Implementation, and Innovation in Technology (I2ITCON), Pune, India, 2025.

[13] D. Ganboi, M. Kumbhar, D. Surnar and H. Patel, "Sanket Bhasha: Multilingual NLP and 3D Avatar-Based Indian Sign Language (ISL) Translator," 2025 IEEE Pune Section International Conference (PuneCon), Pune, India, 2025

[14] N. Naikwade, S. Nipanikar, A. Utikar, S. Waghmare, P. Khune, M. Kumbhar "Cyber-Physical Systems", Springer Nature,2026