

**Research Article****CONTENT ANALYSIS AND DUPLICATION AVOIDANCE SYSTEM USING MACHINE LEARNING****Kajal Davange, \*Ankita Gore, Swamini Dangat, Sakshi Bhor and Divya Dongare**Department of Information Technology Engineering, Dr. Vithalrao Vikhe Patil College of Engineering,  
Ahmednagar, Maharashtra, India**Received 27<sup>th</sup> March 2025; Accepted 29<sup>th</sup> April 2025; Published online 30<sup>th</sup> May 2025**

---

**Abstract**

In many academic institutions, the process of reviewing and approving student project titles and content is still handled manually. This traditional approach is time-consuming, prone to oversight, and often results in unintentional duplication of projects. The current system lacks an organized way to track previously completed work, leading to repetitive submissions and limited innovation. This paper proposes an efficient, centralized repository system for storing past student projects. The aim is to automate the review process, enable easy comparison of new proposals against existing work, and encourage students to explore novel ideas. By leveraging this structured framework, institutions can significantly reduce redundancy and promote originality in academic project development. This paper describes the development and deployment of a custom-built plagiarism detection system aimed at upholding academic integrity within educational institutions. The system leverages natural language processing (NLP) techniques, specifically utilizing the cosine similarity algorithm, to perform accurate textual comparisons. Experimental evaluations demonstrate that the system is both reliable and efficient in identifying instances of plagiarism in student assignments and research documents. Core functionalities include comprehensive text preprocessing, similarity measurement, and an intuitive user interface for ease of use. The results affirm that implementing such dedicated plagiarism detection tools is vital for ensuring originality and fostering a culture of ethical academic practices in universities. This approach not only enhances the detection of textual redundancy but also supports content originality, making it valuable for academic institutions, content management systems, and publishing platforms. The system aims to promote ethical content usage and improve the overall quality of digital information.

**Keywords:** Title verification, content extraction, content analysis, plagiarism detection, TFIDF algorithm Cosine Similarity, Natural Language Processing, Machine Learning.

---

**INTRODUCTION**

Plagiarism, defined as the act of presenting someone else's work, ideas, or expressions as one's own without proper acknowledgment, has become a growing concern in academic institutions. With the rapid expansion of digital information and ease of access to online content, students often resort to copying material, intentionally or unintentionally, which compromises academic integrity. Maintaining academic integrity is essential not only for upholding the credibility of academic qualifications but also for fostering an environment that values originality, critical thinking, and ethical scholarship. In the era of information overload, where content is generated and shared at an unprecedented rate across digital platforms, ensuring originality and avoiding duplication has become a significant challenge. Whether in academic writing, blogging, publishing, or social media, duplicated content undermines credibility, reduces search engine rankings, and poses ethical and legal risks. Traditional plagiarism detection systems rely primarily on string-matching techniques, which are often inadequate for identifying paraphrased or semantically similar content. With the rapid advancement of Machine Learning (ML) and Natural Language Processing (NLP), content analysis has evolved from simple keyword comparisons to complex semantic understanding. Machine learning models, especially those leveraging techniques such as TF-IDF, Cosine Similarity, and deep learning-based embeddings (e.g., BERT or Word2Vec), can now identify hidden patterns and contextual similarities in text.

These technologies allow systems to detect duplication not only at the lexical level but also at the conceptual and semantic levels, significantly improving detection accuracy. The need for a robust Content Analysis and Duplication Avoidance System is especially critical in domains such as academia, digital publishing, and content management. Such a system must be capable of processing large volumes of text, analyzing linguistic features, and identifying duplicate or near-duplicate content efficiently and accurately. Furthermore, it should provide meaningful feedback to users to support content originality and ethical writing practices. This paper proposes a machine learning-based solution designed to perform intelligent content analysis and avoid duplication by understanding the semantic structure of text. The system combines classic statistical methods like TF-IDF with modern deep learning techniques and similarity measures to ensure comprehensive, scalable, and reliable plagiarism detection. By integrating these technologies, the system aims to support educational institutions, content creators, and digital platforms in maintaining content integrity and promoting originality.

**LITERATURE SURVEY**

Title: Design and Implementation of a Self-built Plagiarism Detection System for University Academic Integrity

Author: Xinyu Luo.

This paper presents the development of a custom plagiarism detection system tailored for university settings. The system integrates machine learning algorithms to analyze academic submissions for potential plagiarism. It emphasizes adaptability to institutional requirements and the incorporation of various

---

\*Corresponding Author: *Ankita Gore,*

Department of Information Technology Engineering, Dr. Vithalrao Vikhe Patil College of Engineering, Ahmednagar, Maharashtra, India.

detection techniques, including text similarity measures and code analysis. The study highlights the system's effectiveness in upholding academic integrity by providing timely and accurate plagiarism reports.

Title: Plagiarism Detection in Computer Programming Using Feature Extraction from Ultra-Fine-Grained Repositories.

Authors: Vedran Ljubovic.

This study explores a novel approach to plagiarism detection in computer programming by extracting features from ultra-fine-grained code repositories. The methodology involves analyzing minute code fragments to identify similarities that may indicate plagiarism. By focusing on such granular details, the system aims to detect subtle instances of code reuse or copying that traditional methods might overlook. The research emphasizes the importance of fine-grained analysis in enhancing the accuracy of plagiarism detection tools.

Title: Exploring the Landscape of Intrinsic Plagiarism Detection: Benchmarks, Techniques, Evolution, and Challenges

Authors: Adnan Abid.

This comprehensive review delves into intrinsic plagiarism detection, which focuses on identifying inconsistencies within a single document without external references. The paper surveys existing benchmarks, methodologies, and the evolution of intrinsic detection techniques. It discusses challenges such as the lack of standardized datasets and the complexity of modeling authorial writing styles. The study underscores the need for advanced machine learning models capable of capturing nuanced writing patterns to improve detection accuracy.

Title: Plagiarism Detection in Large Sets of Press Agency News Articles

Authors: Wolfgang Kienreich, Michael Granitzer, Vedran Sabol, Werner Klieber

This research addresses the challenge of detecting plagiarism in extensive collections of news articles. The authors develop a system that combines text similarity measures with metadata analysis to identify duplicated content across press agency publications. The methodology includes clustering techniques and visualization tools to facilitate the detection process. The study highlights the system's scalability and effectiveness in managing large datasets.

Title: Review of Code Similarity and Plagiarism Detection Research Studies

Authors: Gunwoo Lee, Jindae Kim, Myung-seok Choi, Rae-Young Jang, Ryong Lee

This comprehensive review examines various techniques and tools for code similarity and plagiarism detection. The authors categorize existing methods based on their approaches, such as token-based, tree-based, and semantic analysis. The study discusses the strengths and limitations of each technique, providing insights into their effectiveness in different scenarios. The review serves as a valuable resource for researchers and educators seeking to understand the current landscape of code plagiarism detection.

## Related Work

The paper reviews various approaches Content analysis and duplication Avoidance system using Machine Learning to and identifies their strengths and limitations. It categorizes related work into different areas:

### Traditional Content Duplication Approaches:

Early systems for detecting duplicated content relied on rule-based and statistical methods, including:

- String Matching Algorithms (e.g., Rabin-Karp, KMP): Effective for detecting exact matches but perform poorly with paraphrased content.
- TF-IDF & Cosine Similarity: These vector space models captured word frequency-based similarity but lacked semantic understanding.
- Fingerprinting (e.g., MOSS, JPlag): Used hash-based methods for detecting code and document duplication, but sensitive to reordering and renaming.
- Edit Distance (Levenshtein, Jaccard Similarity): Able to detect small changes, but not deeper semantic relationships.

### Machine Learning-Based Approaches

Recent work has shifted toward machine learning techniques to address limitations in traditional methods. These include:

- Support Vector Machines (SVM), Decision Trees, and Random Forests: Applied for supervised classification of similar and dissimilar document pairs.
- Word Embeddings (Word2Vec, GloVe, BERT): Used to detect semantic similarity between texts by transforming words into high-dimensional vector representations.
- CNN & RNN Models: These deep learning models have been employed to detect similarity in both structured and unstructured content.
- Transformer Models (e.g., BERT, Longformer): Enabled context-aware similarity detection, especially for paraphrased or AI-generated content.

These models improved detection performance in identifying hidden, partial, and semantically similar content, but required substantial training data and computational power.

### Limitations of Existing Studies

The study highlights key shortcomings in previous research:

- Semantic Limitations: Many systems struggle to detect paraphrased or semantically restructured content.
- Scalability Issues: Traditional and some ML-based methods are not optimized for large-scale document repositories.
- Multimodal Gaps: Few systems offer integrated support for detecting duplicated content across text, images, and code.
- Real-Time Performance: Many ML models, while accurate, are not suitable for real-time detection due to high computational demands.

To overcome these challenges, the paper proposes an Threshold-Based Similarity Detection: Applies cosine

similarity and dynamic thresholding to flag potentially duplicated or reused content in real-time.

### Problem statement

Academic institutions face a growing challenge in ensuring the originality and integrity of student submissions, research papers, and academic content. With the increasing ease of access to digital information, plagiarism both intentional and unintentional has become a serious concern. Traditional plagiarism detection tools often rely on exact text matching, which may fail to identify paraphrased, translated, or semantically similar content. Therefore, there is a need for an intelligent system that can effectively detect academic plagiarism. Moreover, existing commercial plagiarism detection tools are expensive, not customizable, and may raise concerns related to data privacy and institutional dependency. Therefore, there is a critical need to develop a cost-effective, scalable, and accurate content analysis and duplication avoidance system that leverages machine learning techniques such as TF-IDF, cosine similarity, and semantic embeddings to automatically identify both exact and intelligent content reuse in real time for academic projects.

### Objectives

The primary objectives of this research are:

- Ensure Originality and Quality of Content:

Original content is essential in academic writing, publishing, journalism, and software development. Duplication not only diminishes the value of information but can also lead to misinformation, ethical violations, and reduced academic integrity. A machine learning-based system helps detect not just copy-paste duplication but also paraphrased or structurally altered content, ensuring that what is submitted or published is genuinely original.

- Enhance Information Retrieval Efficiency:

Duplicate or highly similar documents can clutter databases, making it difficult to retrieve relevant and unique information. This redundancy leads to lower precision in search results and increased storage and processing costs. By detecting and filtering out duplicates, the system improves the efficiency and accuracy of search engines, academic repositories, and digital libraries, ensuring users access the most relevant and diverse content.

- Protect Intellectual Property Rights:

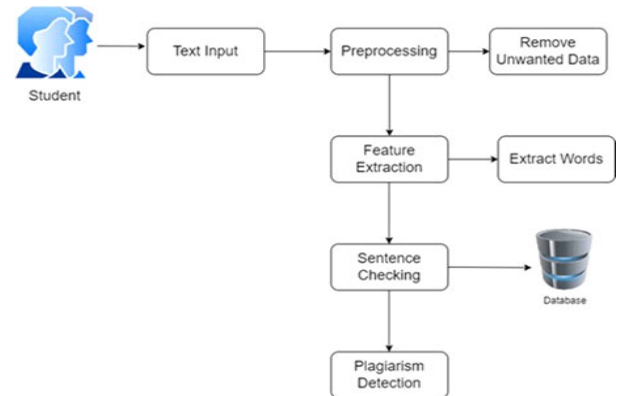
With the massive sharing of digital content online, protecting intellectual property has become increasingly challenging. Whether it's text, source code, or multimedia, duplicated content can be used without proper attribution, violating copyrights. An automated system acts as a safeguard for creators by detecting unauthorized reuse, allowing institutions and authors to defend their work and maintain control over their intellectual assets.

- Ensure Academic and Research Integrity:

Detect plagiarism in academic submissions such as research papers, theses, and project reports to uphold originality.

Identify paraphrased or restructured content using semantic similarity analysis to ensure fair grading and assessment. Custom Dataset Storage for All previously submitted content was stored in a local database, enabling comparisons against institutional or user-defined document libraries.

### Proposed System



### System Architecture

Many graduate students from India are unaware of academic writing, plagiarism, ethical issues and research methodologies. Due to lack of appropriate or experienced teaching instructors in most of the institutions the standards of education and ethics in the education system are under serious threat. Such education system allows many students to practice unethical ways of pursuing an academic degree. Earlier awareness of research methodology and ethical issues were considered necessary only for research scholars doing PhD and was recently extended for post-graduation students as well. Government of India announced plagiarism as an academic fraud and unethical due to which the student or research scholar may attract punishment. In proposed system, to store all previous implemented project data with synopsis in the system. The student registers details first. The details are name, email, password, roll no, branch and year. To verify the title student has to give the project name, keywords and abstract as an input. Then based on content analysis and plagiarism system will process a result. The proposed system for content analysis and duplication avoidance using Machine Learning follows a multi-stage pipeline, combining Natural Language Processing (NLP) with ML models to detect and manage duplicate or semantically similar content. The process is outlined in the following steps:

- **Data Collection:** Collect textual data from various sources such as academic articles ,projects papers etc
- Include both original and duplicate content to train and evaluate the system effectively.
- **Preprocessing:** Clean and normalize the text, Remove stop words, special characters, HTML tags, Convert text to lowercase, Perform tokenization, stemming, or lemmatization, Convert textual data into a structured format suitable for machine learning models.
- **Feature Extraction:** Transform text into numerical features using techniques like, TF-IDF (Term Frequency-Inverse Document Frequency), Bag of Words (BoW), These representations capture the syntactic and semantic features of the text.
- **Similarity Detection:** Compute similarity scores between texts using: Cosine Similarity, Jaccard Similarity, Semantic

Similarity ,Set a threshold score to determine whether two texts are duplicates or not.

- **Machine Learning Model Training :** Train a supervised model to classify text pairs as: Duplicate vs Non-duplicate.
- **Evaluation:** Evaluate the performance using metrics such as: Precision, Recall, F1-Score, Accuracy
- Input: Pairs of text and their similarity scores or embedding vectors.
- Output: Binary classification (duplicate or not).

System design:

The proposed system is a desktop-based or web- integrated application built using Java for the user interface (UI) and connected to a machine learning-based backend. The system is designed to analyze input content (e.g., academic text or projects) and detect duplication or similarity using NLP techniques such as TF-IDF and cosine similarity. It enables real-time plagiarism checking, content monitoring, and semantic similarity analysis for educational or organizational use.

### TFIDF Algorithm

TF-IDF (Term Frequency-Inverse Document Frequency) is a widely used algorithm in Natural Language Processing (NLP) for transforming text into meaningful numerical representations that machine learning models can understand. In the context of a Content Analysis and Duplication Avoidance System, TF-IDF plays a crucial role in identifying the importance of words in a document relative to a collection of documents.

### Cosine Similarity Algorithm

In a Content Analysis and Duplication Avoidance System, one of the key objectives is to measure how similar two pieces of text are, regardless of their exact wording. Cosine Similarity is a powerful and widely-used method for this purpose. It compares two documents by converting them into vectors and calculating the cosine of the angle between these vectors. The smaller the angle, the more similar the documents.

### Conclusion & future work

In conclusion, In this project successfully demonstrates a machine learning-based Content Analysis and Duplication Avoidance System, designed to identify and measure content similarity in real-time using techniques such as TF-IDF and cosine similarity. The system offers a practical and scalable solution to address challenges in detecting plagiarism and redundant content in academic and organizational environments. By integrating natural language processing (NLP) with Java- based interfaces, the application ensures high usability and accuracy, even with paraphrased or slightly altered content. The system effectively supports content moderation, promotes originality, and helps protect intellectual property. Initial evaluations indicate high reliability in duplication detection, making it a suitable tool for educational institutions, editorial boards, and corporate content manager. To further enhance the functionality, robustness, and deployment versatility of the system, the following future enhancements are proposed:

Expanding the dataset to include more complex textual variations such as code snippets, multilingual documents, and domain-specific terminologies, to improve generalization and accuracy. Integrating semantic-aware models like BERT or SBERT, which can detect deeper contextual and paraphrased similarities that traditional vector-based methods might miss. Optimizing the system for lightweight deployment on low-resource environments, such as desktop tools for students or plug-ins for learning management systems (LMS). Building a cloud-based or web-integrated version that supports document uploads and API integration for seamless content analysis across platforms. Developing advanced visualization and reporting tools for teachers, editors, and reviewers to better interpret similarity results and flagged content. Exploring multi-modal content analysis, combining text with code, images, or metadata to build a more comprehensive duplication detection engine. Incorporating live plagiarism prevention features, such as real-time writing feedback or content recommendation systems, to guide users toward more original content creation.

### REFERENCES

1. Vedran Ljubovic, (2020, June 4). Plagiarism detection in computer programming using feature extraction from ultra-fine-grained repositories.
2. Xinyu Luo (2023). Design and implementation of a self-built plagiarism detection system for university academic integrity. IEEE
3. Adnan Abid, (2023, December). Exploring the landscape of intrinsic plagiarism detection: Benchmarks, techniques, evolution, and challenges.
4. Kienreich, W., Granitzer, M., Sabol, V., & Klieber, W. (n.d.). Plagiarism detection in large sets of press agency news articles
5. Wang L., L. Jiang, and G. Qin, "A Search of Verilog Code Plagiarism Detection Method," in *Proc. Int. Conf. Computer Science and Education (ICCSE)*, Colombo, Sri Lanka, Aug. 8–11, 2018.
6. Herrera G., M. Nun~ez-del-Prado, J. G. Lazo Lazo, and H. Alatrasta, "Through an Agnostic Programming Languages Methodology for Plagiarism Detection in Engineering Coding Courses," in *Proc. IEEE Global Engineering Education Conf. (EDUNINE)*, 2019..
7. Xie X., Z. Zhang, T. Y. Chen, Y. Liu, P.-L. Poon, and B. Xu, "METTLE: A Metamorphic Testing Approach to Assessing and Validating Unsupervised Machine Learning Systems," *IEEE Trans. Reliability*, vol. XX, no. X, pp. XX–XX, 2020.
8. Iswara I.B.A.I., Adinata I.M.G.O.G., Sarasvananda I.B.G., and Desnanjaya, I. G. M. N. "Plagiarism Detection System Using the Levenshtein Distance Method for the Balinese Language," in *Proc. Int. Conf. Advanced Informatics for Computing Research*, 2022.
9. Lee G., J. Kim, M.-S. Choi, R.-Y. Jang, and R. Lee, "Review of Code Similarity and Plagiarism Detection Research Studies," *Applied Sciences*, vol. 13, no. X, pp. XX–XX, 2023
10. Herrera G. and M. Nun~ez-del-Prado, "Through an Agnostic Programming Languages Methodology for Plagiarism Detection in Engineering Coding Courses (Updated Work)," IEEE, 2021.
11. Foltýnek T., N. Meuschke, and B. Gipp, "Academic plagiarism detection:A systematic literature review," *ACM*

- Comput. Surv.*, vol. 52, no. 6, pp. 1–42, Nov. 2020, doi: 10.1145/3345317.
12. Oloo V. A., C. Otieno, and L. A. Wanzare, “A literature survey on writing style change detection based on machine learning: State-of-the-art–review,” *Int. J. Comput. Trends Technol.*, vol. 70, no. 5, pp. 15–32, May 2022, doi: 10.14445/22312803/ijctt-v70i5p103.
13. Alzahrani S. and H. Aljuaid, “Identifying cross-lingual plagiarism using rich semantic features and deep neural networks: A study on Arabic-English plagiarism cases,” *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1110–1123, Apr. 2022, doi: 10.1016/j.jksuci.2020.04.009.
14. Saini A., M. R. Sri, and M. Thakur, “Intrinsic plagiarism detection system using stylometric features and DBSCAN,” in *Proc. Int. Conf. Comput., Commun., Intell. Syst. (ICCCIS)*, Feb. 2021, pp. 13–18, doi: 10.1109/icccis51004.2021.9397187.
15. Zainuddin N., A. Selamat, and R. Ibrahim, “Hybrid sentiment classification on Twitter aspect-based sentiment analysis,” *Appl. Intell.*, vol. 48, no. 5, pp. 1218–1232, 2018, doi: 10.1007/s10489-017-1098-6.
16. Bruton, S. V. “Self-plagiarism and textual recycling: Legitimate forms of research misconduct,” *Accountability Res.*, vol. 21, no. 3, pp. 176–197, May 2014, doi: 10.1080/08989621.2014.848071.

\*\*\*\*\*