International Journal of Trend in Scientific Research and Development (IJTSRD)
Special Issue on Advancements and Emerging Trends in Computer Applications - Innovations, Challenges, and Future Prospects
Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Plagiarism Detection: Improving the Peer Review Process in Academic Publishing

Rutika A. Ghotkar

PG Student, Department of Computer Applications, G. H. Raisoni University, Amravati, Maharashtra, India

ABSTRACT

Plagiarism is a infraction for world that impacts the quality, readability, and trustworthiness of scholarly publications. Improving researcher awareness of plagiarism of words, ideas, and graphics is essential for avoiding unacceptable writing practices. A lack of creative thinking and poor academic English skills are believed to compound most instances of redundant and "copy-and-paste" writing. Plagiarism detection software largely relies on reporting text similarities. However, manual checks are required to reveal inappropriate referencing, copyright violations, and substandard English writing. Additionally, the paper discusses the assumptions on technology for plagiarism detection and the importance of focuses on a culture of originality and proper practices among students and researchers. The system utilizes **text-matching algorithms, natural language processing (NLP), and machine learning** techniques to detect various forms of plagiarism, including **direct copy-pasting, paraphrasing, synonym replacement, and code plagiarism**. For making the project we used languages like Html, CSS, Bootstrap and ReactJS for front-end and JAVA for back end and MySql for data base connectivity.

KEYWORDS: *Plagiarism detection, Academic Integrity, Text similarity Analysis, Plagiarism detection, Plagiarism detection techniques, Textual analysis Algorithm, Plagiarism detection Analysis.*

I. INTRODUCTION

This project presents a **Plagiarism Checker**, a system designed to detect similarities between documents by comparing them against a database of existing sources. These tools utilize advanced algorithms and computational techniques to analyse textual content for similarities, matching phrases, and patterns of copied material. While early plagiarism detection systems focused primarily on identifying exact matches, modern tools incorporate a broader range of functionalities, including the detection of paraphrased content, citation errors, and even the analysis of writing style. Digitization made copy-paste Plagiarism and inappropriate reuse of source from the websites, online journals and other electronic media. The global open access movement has made it possible to easily reveal most instances of plagiarism, including copying texts and graphics across digitized old and new sources.

Current anti-plagiarism software may detect unacknowledged recycled (self-plagiarized) texts, the so called salami (data stemming from a single study spread across several papers) and augmented (opposite to salami) texts. Accusations of plagiarism in such cases require thorough manual checks of all similar parts, particularly by experts in the professional field. Plagiarism detection and verification is largely based on text-matching search engines and computer software that report similarity scores. The advanced software is integrated with numerous publishers and online platforms to allow scanning of potential overlaps among countless open-access and subscription literature items. Perhaps the most advanced anti-plagiarism system is iThenticate, which is employed by most established publishers to report the overall similarity score and similarity score from a single source.

II. RELATED WORK

These systems are effective in detecting literal matches and identifying copy-paste plagiarism. They do tend to fail with paraphrased material where sentence structure is changed without altering the meaning. Although useful for broad comparisons, their precision falls short when semantic similarity must be determined.

To overcome this limitation, fingerprinting techniques using methods like n-grams and shingling have been widely adopted in systems. These approaches generate unique fingerprints for documents and perform comparisons to detect partial matches. While fingerprinting methods are computationally efficient and suitable for large databases, they still fail to detect semantic plagiarism or complex sentence rephrasing.

With the help of Natural Language Processing (NLP) advancements, scholars have created more advanced plagiarism detection systems. NLP methods, including Latent Semantic Analysis (LSA) and Word2Vec embedded, compare sentences on the basis of contextual meaning to check for plagiarism. AI-based applications such as Grammarly utilize transformer models such as BERT to identify paraphrased plagiarism with much greater accuracy. Research by Ghosh et al. (2021) proved that an AI plagiarism detection system based on BERT had a 94% accuracy rate, which was higher than conventional systems. Nevertheless, the computational complexity of such models and their dependency on large training datasets are still issues.

III. DATA AND SOURCES OF DATA

A plagiarism detection system may classify text into original and copied content. Describe how proper labelling and data organization are essential for training machine learning models. Likewise, in text data for plagiarism detection, n-grams, sentence structure, and semantic meaning may be features. Describe how feature extraction is significant for both image and text analysis. We can also talk about the need for a clear training and test dataset in both cases to validate the accuracy and reliability of the model. in plagiarism detection, these are also used to measure the efficiency of the plagiarism detection algorithms. Difficulties involve

detecting paraphrasing, contextual understanding, and accommodating varying writing styles. Ethical concerns involve academic integrity and the implications of false positives.

IV. RESEARCH METHODOLOGY

A well-structured investigate strategy for considering workforce engagement and office elements includes a combination of subjective and quantitative approaches to analyse engagement levels, collaboration, and worker well-being. Underneath are the key methodological components.

Figure I:-

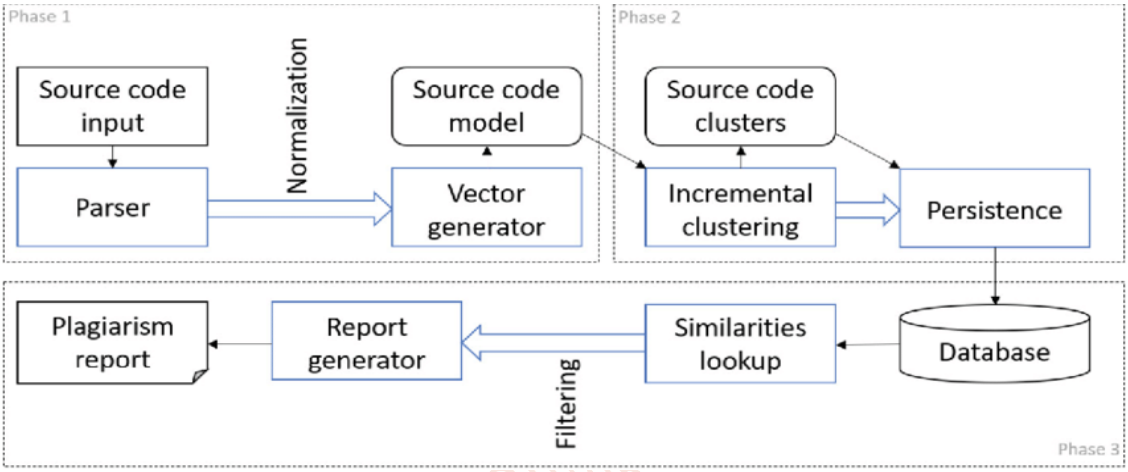

Fig1: Structured image for analyse and detect Plagiarism.

This diagram represents a well-structured approach for a plagiarism checker for source code. It consist of three phases:

1. **Normalization (Phase 1):** Normalization is the procedure of transforming raw source code into a uniform format to enable meaningful and efficient comparison. This assists in detecting plagiarism even when the source code has been slightly altered (e.g., renaming variables, reformatting, adding comments). Normalisation phase sub-classified into two phases:

A. Parsing the Source Code:

- Parsing is the process of breaking down the source code into structured form. • A parser breaks down the code and translates it into tokens (keywords, operators, variables, etc.).
- This process removes superficial variations like additional spaces, indentation, or alternative formatting schemes.

B. Removing Comments & Whitespace:

- Comments and whitespace are not part of what makes the code function, therefore must be stripped. 3. Standardizing Variable & Function Names (Lexical Normalization) • Plagiarists rename functions and variables in an attempt to evade detection.
- The answer is to substitute all function and variable names with placeholders.

C. Converting Code into a Vector Representation

- Instead of storing raw text, convert the normalized code into a numerical vector for easier similarity comparisons.

2. **Clustering & Storage (Phase 2):** Once the source code is normalized, we need to group similar codes together so that new submissions can be efficiently compared. This phase involves incremental clustering and storing the clustered data in a structured database.

A. Generating Feature Vectors for Code Similarity

- Before clustering, we need to convert each normalized source code into a vector representation.
- The idea is to transform code into numerical form so that we can use mathematical distance measures to find similar pieces of code.

B. Clustering Similar Source Codes

- Once we have vector representations of the code, we apply clustering to group similar submissions together.
- These clusters are stored in a database for future comparisons.
- The vectorised representation of source code is grouped into clusters through incremental clustering.

C. Similarity Checking & Report Generation:

Compare newly submitted code against the stored database and generate a plagiarism report.

- Similarities Lookup**:** Searches for matching or highly similar code snippets.
- Report Generator**:** Prepares a human-readable plagiarism report
- When a new source code is submitted, a similarity lookup is performed against the database.
- If similarities are found, a report generator produces a plagiarism report.

V. RESULTS AND DISCUSSION

This bar chart visually represents the **detection accuracy (%)** of a plagiarism checker across different types of plagiarism. Each bar corresponds to a specific plagiarism type and shows how effectively the tool identifies it..

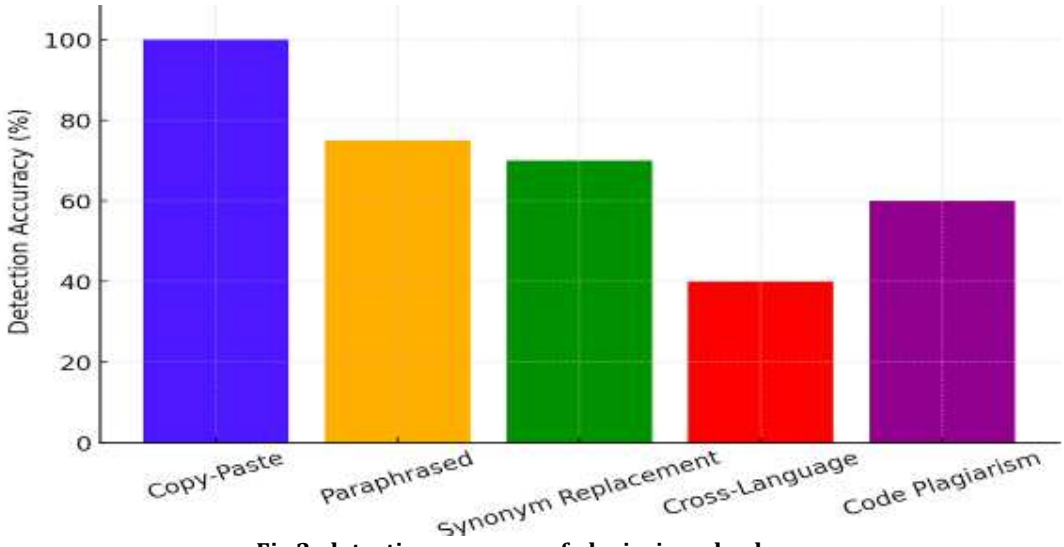

Fig 2: detection accuracy of plagiarism checker.

Figure: Key Observations

- **Copy-Paste Plagiarism:**
  - Has the highest detection accuracy at **100%**.
  - The tool performs best when detecting exact copies of content.
- **Paraphrased Plagiarism:**
  - Accuracy is around **75%**.
  - Some paraphrased content is detected, but some variations may bypass the system.
- **Synonym Replacement:**
  - Detected with **70% accuracy**.
  - The tool catches most synonym changes but may miss advanced rewording.
- **Cross-Language Plagiarism:**
  - The lowest accuracy at **40%**.
  - This suggests that the tool struggles with translated text plagiarism.
- **Code Plagiarism:**
  - Accuracy is **60%**.
  - The tool can detect copied code but may not recognize heavily modified code snippets.

Table1: Strategies For plagiarism Detection.

Text Passages	Matching Identical Passages
Keywords and logical Words	Revealing semantic Overlaps
Writing Styles	Reporting mixed passages with American and British Styles
Methodologies	Distinguishing similarities in the sets and order of tests
References	Comparing similarities in the lists and order of individual references.
Hypothesis	Comparing similarities and reporting episodes of privileged exposure to unpublished intellectual property.
Graphics	Visualization of identical images, tables and figures.

Researchers should be aware of what constitutes plagiarism and how to detect it (Table I). Those authors who master academic English, familiarize themselves with bibliographic searches, and advance their graphics designing skills may avoid most instances of plagiarism, duplication, and copyright infringement. Those who employ anti-plagiarism tools should combine software and human-detection options.

Plagiarism detection and verification is largely based on text-matching search engines and computer software that report similarity scores. The advanced software is integrated with numerous publishers and online platforms to allow scanning of potential overlaps among countless open-access and subscription literature items. Perhaps the most advanced anti-plagiarism system is iThenticate, which is employed by most established publishers to report the overall similarity score and similarity score from a single source. The system offers options to filter direct quotations, bibliographies, and methodologies to minimize chances of erroneous reports. Overall similarity scores (>35%) often point to plagiarism requiring outright rejection. Compared to textual similarity detection, image plagiarism detection is a more challenging task, since it often requires both image processing and semantic mapping techniques [46, 47]. Google Images is a widely available search engine that can be used to reveal identical or manipulated images processed by Google [48]. However, this engine fails to detect copied and modified graphical materials. Semantic analyses are particularly useful in such a scenario of image modification.

```
double sort(int *A[], int n)
{
    int i;int asdf;
    for(i=n-2;i>=0;i--)
    {
        int j;asdf++;
        for(j=0;j<=i;j++)
        {
            asdf++;
            if(asdf==0)
            {
                asdf=0;
            }
            if(A[j][0] > A[j+1][0])
            {
                int *temp;asdf++;
                temp = A[j];asdf++;
                A[j] = A[j+1];asdf++;
                A[j+1] = temp;asdf++;
            }
        }
    }
}
```

```
int main()
{
    int n,m,mod,w,i,j,k;
    scanf("%d %d",&n,&mod);
    int *A[n];
for(i=0;i<n;i++)
{
    scanf("%d",&m);
    int new[m+4];
    A[i]=(int*)malloc((m+4)*
    A[i][0]=0;
    for(j=1;j<=m;j++)
    {scanf("%d",&A[i][j]);
    A[i][0]=(A[i][j]+A[i][0]
    }
    A[i][j]=-1;
}
sort(A,n);
for(i=0;i<n;i++)
{j=0;
while(A[i][j]>=0)
printf("%d\n",A[i][j++]);
}
return 0;
}
```

Fig 3:- Image of Code

Fig 4:- Report Generate

Fig 5:- Result of plagiarised content

VI. CONCLUSION

Plagiarism continues to affect the integrity of scholarly publications worldwide. Digitization and open access pro vide numerous opportunities for accessing and disseminating scientific information. However, some researchers and authors are tempted to intentionally or unintentionally embark on shortcuts and construct their articles with copied and unattributed texts, graphics, and ideas. Arguably, educating authors how to systematically access and process literature and how to master academic English may prevent most instances of modern-day plagiarism.

Systematic searches are necessary for choosing new topics and avoiding redundancies. Processing retrieved articles, appropriately referring to published scientific facts, and writing in one's own words may further improve the ethical standing of new manuscripts.

Researchers and research managers alike need to learn more about globally acceptable writing practices, regularly analyse retractions due to plagiarism, and avoid related errors in their practice. Knowledge of global editorial guidance and plagiarism detection and prevention strategies is essential for successful writing and targeting influential ethical journals. Journal editors should en force a "trust, but verify" policy by performing plagiarism checks, inquiring about authors' writing practices, and asking for disclaimers if suspicion of plagiarism persists.

There are some chances that some users may use copy-paste method from the site. Plagiarism detection give permission to keep your essay and checkout if it is also available somewhere else on the web are not. To check the reality of the content of your work such as an article, poem or essay, use a high quality Plagiarism Checker to find out if your content is Plagiarise or not. This is the citation that is common for library professionals, research scholars andudents also.

VII. REFERENCES

[1] Bouville M. Plagiarism: words and ideas. Science English Ethics 2008; 14: 311–322, DOI:10.1007/s11948-008-9057-6.

[2] de Vasconcelos SM, Roig M. Prior Publication and Redundancy in Contemporary Science: Are Authors and Editors at the Crossroads? Science English Ethics 2015; 21: 1367–1378, DOI: 10.1007/ s11948-014-9599-8.

[3] Roig M. Avoiding unethical writing practices. Food Chemistry Toxi col 2012; 50: 3385–3387, DOI:10.1016/j.fct.2012.06.043.

[4] Wang T, Xing QR, Wang H, Chen W. Retracted Publications in the Biomedical Literature from Open Access Journals. Science English Ethics 2019; 25: 855–868, DOI: 10.1007/s11948-018-0040-6

[5] Heaven D. AI peer reviewers unleashed to ease publishing grind. Nature 2018; 563: 609–610, DOI:10.1038/d41586-018-07245-9.

[6] Mehregan M. Ethical Reviewers are Essential for Scholarly Journals for Timely Processing of Submissions and Avoiding Retractions. J Korean Medical Science 2019; 34: e41, DOI: 10.3346/ jkms.2019.34.e41.

[7] Yi N, Nemery B, Dierickx K. Perceptions of plagiarism by biomedical researchers: an online survey in Europe and China. BMC Med Ethics 2020; 21: 44, DOI:10.1186/s12910-020-00473-7.

[8] Core practices. Available from: https://publicationethics.org/core-practices [Accessed 27.01.2021].

[9] Misra DP, Agarwal V. Integrity of clinical research conduct, re porting, publishing, and post-publication promotion in rheumatology. Clinic Rheumatol 2020; 39: 1049–1060, DOI: 10.1007/ s10067-020-04965-0.