IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2025

DOI: 10.17148/IJIREEICE.2025.13309

A COMPARATIVE ANALYSIS OF JACCARD AND COSINE SIMILARITY FOR PLAGIARISM DETECTION

Kanishkaa. S[1], Santhi. K[2]

Department of Information Technology, Dr N.G.P Arts and Science College, Coimbatore, Tamil Nadu, India[1]

Professor, Department of Information Technology, Dr N.G.P Arts and Science College, Coimbatore, Tamil Nadu, India[2]

Abstract: Plagiarism, the unauthorized use or imitation of another's work without proper acknowledgment, poses a significant challenge in academia, research, and professional content creation, amplified by the widespread sharing of digital information. Reliable plagiarism detection systems are essential to ensure originality and maintain integrity. This paper investigates two widely used algorithms—Jaccard and Cosine similarity—for their effectiveness in detecting textual similarities. Jaccard similarity excels in identifying exact or near-exact overlaps but struggles with rephrased content, whereas Cosine similarity captures deeper semantic similarities, including paraphrasing, but is computationally more demanding. Preprocessing techniques, such as tokenization, stop word removal, and stemming, are employed to optimize the algorithms' performance. The research evaluates their strengths, limitations, and computational efficiency through a detailed comparative analysis, offering insights into their suitability for specific applications. The findings emphasize the importance of balancing detection accuracy with computational demands, guiding the selection of appropriate methods for plagiarism detection in various contexts.

Keywords: Plagiarism Detection, Cosine Similarity, Jaccard Similarity, Text Similarity, Text Preprocessing

I. INTRODUCTION

Plagiarism, or using someone else's work without giving proper credit, is a serious issue in education, research, and professional writing. With the growing amount of digital content and the ease of sharing information, it has become more challenging to ensure originality in written work. This has led to a need for effective plagiarism detection systems.

Plagiarism detection methods usually focus on either syntactic or semantic similarities. Syntactic methods, like the Jaccard similarity algorithm, are good at finding exact or near-exact matches between texts but often miss paraphrased content. Semantic methods, like the cosine similarity algorithm, can detect deeper meanings and paraphrasing but are more complex and require more computing power.

This paper compares the Jaccard and cosine similarity algorithms to understand their strengths and weaknesses in detecting plagiarism. It looks at how well they identify different types of text similarities and how efficient they are in terms of performance. The study also uses preprocessing steps, such as removing common words, breaking text into smaller parts, and simplifying words, to improve the accuracy of both methods.

According to Sastroasmoro in[3], plagiarism based on the percentage of words taken or traced is divided into 3 categories, such as:
a) Light Plagiarism: < 30%.
b) Medium Plagiarism: 30% - 70%
c) Heavy Plagiarism: >70%
This research helps readers understand how these tools work and which one might be best for a particular plagiarism detection task.

II. LITERATURE REVIEW

Plagiarism originated from the Latin "plagiarus" which means kidnapping. The definition according to the Big Indonesian Dictionary is "plagiarism that infringes copyright". Meanwhile, according to [1] plagiarism is the act of copying or stealing other's works such as ideas, writing ideas, then claiming it as a result of his own work without including orientation from the original source. According to Parvati in [2], the type of plagiarism is divided into 4 such as [1]:

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2025

DOI: 10.17148/IJIREEICE.2025.13309

a) Word-for-word Each word is copied exactly without any changes.
b) Plagiarism of authorship the author is changed to his own name and then acknowledges the work to his work.
c) Plagiarism of Ideas. Ideas from others are recognized as his ideas.
d) Plagiarism of Sources The source is not written on the work using the quotation.

More studies have recently begun to focus on the investigation of plagiarism detection techniques and methods. According to the surveys and studies that have been reported (Yerra and Ng, 2005; Ali et al., 2011; Alzahrani et al., 2012; Martins et al., 2014; Osman et al., 2012; Osman et al., 2012; Vani and Gupta, 2016; Alvi et al., 2014), plagiarism detection methods can be broadly classified as character-based, structural-based (Chow & Rahman, 2009), cluster-based, syntax-based, semantic-based (Abdia et al., 2015, Alzahrani et al., 2015, Osman et al., 2012; Vani & Gupta, 2017b), citation-based (Gipp, 2014, Gipp and Beel, 2010), and cross-language based.

Syntactic Similarity Approaches:
Syntactic techniques use word or phrase overlaps to identify commonalities in text structure. Because of its ease of use and effectiveness, the Jaccard similarity algorithm is a well-liked technique in this field. It is useful for finding perfect or almost exact matches between documents since it calculates the intersection over the union of word sets. Because it ignores synonyms and word order variations, the Jaccard similarity algorithm is limited in its ability to detect paraphrased or semantically identical information.

Semantic Similarity Approaches:
Semantic similarity techniques like cosine similarity go beyond mere word matching. The cosine of the angle between text vectors in a multi-dimensional space is computed via cosine similarity, which aids in identifying semantic similarities, including paraphrased content. Its computational complexity may be a disadvantage, particularly when used on lengthy documents or huge datasets, even though it can detect more sophisticated types of plagiarism, such as paraphrase.

III. METHODOLOGY

Text Preprocessing:
Before applying any similarity algorithms, several preprocessing steps are carried out on the raw text data to improve accuracy and efficiency[9]:
1. **Tokenization:** The text is broken down into smaller units called tokens (such as words or phrases) to facilitate the comparison process.
2. **Stop Word Removal:** Commonly used words (such as "the", "and", "is") that do not add significant meaning to the text are removed.
3. **Stemming:** Words are reduced to their root form (e.g., "running" becomes "run") to standardize variations and improve the comparison.
4. **Lowercasing:** All text is converted to lowercase to avoid mismatches due to case differences.

Jaccard Similarity Algorithm:
The **Jaccard similarity** (also known as the Jaccard index or Jaccard coefficient) is a statistic used to measure the similarity and diversity of sample sets. It is particularly useful in comparing the similarity of two sets by calculating the ratio of their intersection to their union. In the context of text analysis and plagiarism detection, Jaccard similarity is often used to compare sets of words (or tokens) between two documents to find overlapping content.

Theory of Jaccard Similarity:
The Jaccard similarity coefficient J(A,B) between two sets A and B is defined as[8]:

$$J(A,B)=|A \cup B| / |A \cap B|$$

Where:
- $|A \cap B|$ is the number of elements common to both sets (the intersection).
- $|A \cup B|$ is the total number of elements in both sets (the union).

A and B represent two documents, the sets are typically composed of individual words or tokens. The Jaccard similarity measures the ratio of the common words in both documents to the total unique words across both documents.

Steps for Jaccard Similarity in Text Analysis:
1. **Tokenization**: Split the text into individual words or tokens.
2. **Stop Word Removal**: Optionally, remove common words (e.g., "the", "and", "is") that do not add much value in detecting meaningful similarities.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2025

DOI: 10.17148/IJIREEICE.2025.13309

3. **Set Creation**: Convert the text into sets, where each word or token in a document is an element of the set.
4. **Jaccard Calculation**: Compute the ratio of the intersection of the two sets to their union.

Jaccard similarity algorithm's limitations in simple terms:

1. **Only Looks for Exact Matches**: It can't detect when text is reworded or paraphrased.
2. **Doesn't Understand Meaning**: It only compares words, not what they mean.
3. **Stops Words Can Mess It Up**: Common words (like "the" or "and") can make the similarity score less accurate if not removed.
4. **Slows Down with Large Datasets**: It can be slow when comparing many documents.
5. **Small Sets Can Give False Results**: If there are only a few words, even one common word can make the similarity look higher than it is.

Cosine Similarity Algorithm:

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space. It is widely used in text analysis to compare the similarity between two documents by calculating the cosine of the angle between them. This measure is particularly useful in high-dimensional spaces like text, where the dimensions correspond to words or tokens.

Theory of Cosine Similarity:

The cosine similarity between two vectors A and B is defined as:

$$\text{cosine\_similarity}(A,B) = A \cdot B / \|A\|\|B\|$$

Where:

- $A \cdot B$ is the dot product of the two vectors A and B.
- $\|A\|$ and $\|B\|$ are the magnitudes (norms) of the vectors A and B, respectively.

Steps for Cosine Similarity in Text Analysis:

1. **Text Vectorization**: Convert the text into vectors using techniques like TF-IDF(Term Frequency-Inverse Document Frequency).
2. **Dot Product Calculation**: Calculate the dot product between the two vectors.
3. **Magnitude Calculation**: Compute the magnitudes of each vector.
4. **Cosine Calculation**: Compute the cosine of the angle between the two vectors using the formula above.

Cosine Similarity Algorithm's Limitations:

1. **Ignores Word Order**: It doesn't consider the order of words, which can affect meaning in some cases.
2. **Doesn't Capture Synonyms**: It treats different words with different meanings (e.g., "car" and "automobile") as unrelated.
3. **Requires Vector Representation**: The text needs to be represented as vectors, which can be computationally expensive and may require advanced methods like TF-IDF.
4. **May Overlook Semantic Meaning**: Cosine similarity focuses on the term frequency and doesn't understand deeper meanings or contexts of words.

IV. SYSTEM ARCHITECTURE

High-Level Workflow:

1. **Input the Document**: The user uploads or enters the document text.
2. **Preprocessing:** The system preprocesses the text (tokenization, stop word removal, stemming).
3. **Text Vectorization:** The system generates vector representations of the text for similarity comparison (using Jaccard or Cosine similarity).
4. **Web Search for Matches:** The system sends the document text or its key terms to search engines or databases to find similar content online.
5. **Similarity Calculation:** Once matches are found on the web, the system calculates the similarity between the input document and the found content using either Jaccard or Cosine similarity algorithms.
6. **Output Similarity Report:** The system generates a report showing the matched percentage, the plagiarized sections, and the sources from the web.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2025

DOI: 10.17148/IJIREEICE.2025.13309

Figure 1: High-level workflow Diagram

V. RESULTS

Table 1: Comparision of Jaccard and Cosine Similarity Algorithms

Category	Jaccard Similarity	Cosine Similarity
Identical Text	High	High
Minor Modifications	Moderate	High
Paraphrased Text	Low	Moderate

VI. APPLICATIONS AND USE CASES

Applications and Use Cases of Plagiarism Detection System:

1. **Academic Institutions:** Used to check student papers, research, and dissertations for plagiarism to ensure academic integrity.
2. **Content Creation Platforms:** Freelance platforms and blogs use it to ensure content is unique.
3. **Media and Journalism:** Check news articles and reports for plagiarism.

Advantages of Web-Based Application:

1. **Easy Access:** Accessible from any device with an internet connection.
2. **Scalable:** Can handle large volumes of documents with flexible pricing.
3. **Instant Results:** Provides quick plagiarism checks and reports.
4. **Cross-Platform:** Works on various devices without compatibility issues.
5. **Cost-Effective:** No infrastructure or maintenance costs for users.
6. **Collaboration:** Multiple users can collaborate and integrate with other platforms.

These tools are essential for maintaining originality and preventing plagiarism across different sectors.

Analysis:

From the conclusion of the Table 1:

- Jaccard Similarity is effective for struggles with paraphrasing.
- Cosine Similarity performs better for text with minor modifications, capturing frequency-based variations.

VII. RECOMMENDATIONS FOR FUTURE WORK

1. **Detecting Image Plagiarism**: Extend the system to identify plagiarism in images, which is increasingly important in fields like visual arts and digital media.

2. **Detecting Code Plagiarism**: Adapt the approach for code plagiarism detection, especially in academic and professional programming contexts.

3. **Multilingual Support**: Develop capabilities to handle plagiarism detection in multiple languages, expanding the tool's applicability globally.

4. **Real-Time Detection**: Integrate real-time plagiarism detection in content creation platforms, such as social media or blogsites, to prevent unauthorized copying instantly.

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.021 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2025

DOI: 10.17148/IJIREEICE.2025.13309

VIII. CONCLUSION

This paper examined the effectiveness of Jaccard and Cosine similarity algorithms in plagiarism detection, focusing on their strengths and limitations. Jaccard similarity proved efficient in identifying exact or near-exact text overlaps but struggled with detecting paraphrased or contextually rephrased content. In contrast, Cosine similarity demonstrated its strength in capturing semantic similarities, including paraphrasing, making it more suitable for nuanced text analysis. However, this depth comes with higher computational complexity. The findings emphasize the importance of choosing the right algorithm based on specific detection requirements, balancing accuracy and efficiency for optimal plagiarism detection.

REFERENCES

[1]. S. Dewanto, Indriati and I. Cholissodin, "Deteksi Plagiarisme Dokumen Teks menggunakan Algoritma Rabin-Karp dengan Synonym Recognition".

[2]. I. W. S. Priantara, D. Purwitasari and U. L. Yuhana, "Implementasi Deteksi Penjiplakan dengan Algoritma Winnowing pada Dokumen Terkelompok," in Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh November Surabaya, pp. 1-9, 2011.

[3]. Herqutanto, "Plagiarisme, Runtuhnya Tembok Kejujuran Akademik," eJurnal Kedokteran Indonesia , vol. I, no. 1, pp. 1-3, 2013

[4]. Zhang, X., & Zhang, L. (2010). A Comparative Study of Similarity Measures for Plagiarism Detection in Texts. Proceedings of the 2nd International Conference on Computer Engineering and Technology, 5, 321-325.

[5]. Sundararajan, V., & Keerthi, S. S. (2002). Cosine Similarity and Its Applications in Plagiarism Detection. International Journal of Computer Applications, 6(3), 22-28.

[6]. Yin, H., & Wang, J. (2010). A Cosine Similarity-based Method for Detecting Plagiarism. International Journal of Computer Science and Information Security (IJCSIS), 7(1), 58-63.

[7]. Haque, M. M., & Choi, M. (2014). A Study on Text Similarity Using Cosine Measure and TF-IDF for Plagiarism Detection. Journal of Applied Mathematics and Computation, 25(2), 45-58.

[8]. Sam Fletcher, Md Zahidul Islam. Comparing sets of patterns with the Jaccard index. Australasian Journal of Information Systems,2018, Vol 22, Research Article.

[9]. Dr. S. Vijayarani, Ms. J. Ilamathi, Ms. Nithya, Preprocessing Techniques for Text Mining - An Overview, Dr.S.Vijayarani et al, International Journal of Computer Science & Communication Networks, Vol 5(1),7-16.