VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

WINTER SEMESTER 2022-23

CLASS GROUP	COURSE CODE	COURSE TITLE	COURSE TYPE	CLASS ID	SLOT
General (Semester)	CSE1901	Technical Answers for Real World Problems (TARP)	Embedded Theory	VL2022230504990	TCC1

ASSIGNMENT – 5

Submitted by

Alok Mathur (20BCE2959)

Prasoon Soni (20BCE2960)

Under the guidance of

Dr. Rajkumar S

Associate Professor Sr.

SCOPE, VIT, Vellore

USING AHP AND GOOGLE DORKING FOR MULTI-LEVEL PLAGIARISM DETECTION IN RESEARCH PAPER

ABSTRACT

The widespread access to electronic resources on the internet has led to a rise in plagiarism among students. Many students at institutions around the world are turning in plagiarized papers to their professors for credit, with no clear data on how much was previously plagiarized or how much is now plagiarized. This is a significant concern for teachers, who are already overwhelmed with responsibilities and want a simple method for quickly identifying and reformulating plagiarized papers so that they can focus their efforts on the remaining students. To address this problem, plagiarism-detecting software tools have been developed to detect plagiarism in exams, projects, publications, and academic research. Recent studies have shown that plagiarism detection tools have difficulty identifying more sophisticated forms of plagiarism, such as when text is heavily reworded or when original text is replaced with similar text using foreign characters. In this research paper, we propose a novel approach for plagiarism detection using the Analytic Hierarchy Process (AHP) technique. Our approach aims to address this problem by providing a multi-level plagiarism detection system that is both accurate and efficient. Our proposed approach comprises of three levels of plagiarism detection: level-0, level-1, and level-2. In level-0, we use the google dorking technique to search for potential plagiarized papers available on the internet. The google dorking technique is a powerful method that allows us to search the internet using specific keywords and operators to retrieve information that might not be easily accessible through regular search engines. By using this technique, we can quickly identify potential plagiarized papers and verify their authenticity. If a paper is found to be plagiarized beyond a set threshold value, it is rejected and does not proceed to level-1. In level-1, we check for plagiarism in the introduction, abstract, and references of the paper. These sections are commonly unchanged from the original paper and are often used as a quick check to identify potential plagiarism. We use advanced text comparison algorithms to compare the submitted paper with existing papers in our database, and generate a plagiarism report that highlights any similarities found. In level-2, we perform a full paper plagiarism check to provide comprehensive detection results. We use advanced algorithms to scan the entire paper for plagiarism, and generate a detailed report that highlights any similarities found. This level of plagiarism detection is essential for ensuring that the paper is original and has not been copied from any other source.

KEYWORDS

Natural Language Processing (NLP), Plagiarism Detection, Research Papers, Analytic Hierarchy Process (AHP), Semantic Analysis, Paraphrasing, Google Dorking

1. INTRODUCTION

The submission of research proposals to various funding agencies can be a time-consuming and resource-intensive process for researchers. One of the main challenges faced by researchers is the possibility of their proposals being rejected due to duplicity or plagiarism, leading to wasted time and resources. The typical research proposal can take weeks to months or even years to get approved, and researchers may not be informed of the reasons for rejection. The submission process involves numerous stages, including selecting a specific problem to solve and choosing a funding organization to apply to. However, paraphrasing, and other malpractices can often go unnoticed during basic plagiarism checks, leading to proposals being stuck in the submission stage and not getting approved. This can lead to a loss of funding for deserving candidates and slow down the growth of the research field. Considering these challenges, this research paper aims to propose a system to address the issue of duplicity and plagiarism in research proposals and improve the submission process for researchers.

1.1 BACKGROUND

Research proposals are a vital part of academic and scientific research, but they can also be a time-consuming and frustrating process. The approval process typically involves multiple stages, each of which can take a significant amount of time. One common cause of delays is the need to submit the proposal in a specific format and to decide on important details such as the problem to be solved or the organization to apply to. Once the proposal is submitted, it must be checked for plagiarism and then peer-reviewed. Finally, funding approval or publication must be secured. Each stage of the process is crucial, but it can be difficult to get timely feedback or to know if a proposal has been rejected. Despite the challenges, it's important to remember that a well-crafted research proposal can help to secure funding and support for important research projects.

1.2 DEFINATIONS

The definitions of certain terms used in the paper are provided below:

- Research Proposal: A document outlining a proposed research project, typically including details on the research question, methodology, and potential impact.
- Plagiarism: The practice of using someone else's work or ideas without giving them proper credit.
- Peer review: The process of having a research proposal evaluated by experts in the same field.
- AHP: Analytic Hierarchy Process, a decision-making method that helps to evaluate complex problems by breaking them down into smaller, more manageable components.
- Natural Language Processing (NLP): A field of Artificial Intelligence that focuses on developing computer systems capable of understanding and processing human language, enabling communication between machines and humans.

1.3 APPLICATIONS

Streamlining the research proposal submission process can help to make the process more efficient and effective for researchers. One way to do this is by providing a central platform for researchers to submit their proposals and have them checked for plagiarism. This can be achieved by using a custom-built NLP algorithm for plagiarism checks. Additionally, providing researchers with feedback on why their proposals were not approved can help them to improve their proposals in the future. Another important goal is to facilitate the growth of the research field by connecting innovative minds with funding resources. This can be accomplished by providing an extra layer of security by encrypting the data via Bcryptjs and storing it in the Ethereum blockchain. By ensuring that researchers submit their proposals in a selected format, it helps to create an environment of responsible and competitive research in the country. This can be achieved by checking the proposals for plagiarism and by giving feedback on the proposals that are not approved. This can help researchers to create better proposals and improve their chances of getting funding in the future.

2. LITERATURE REVIEW

A number of recent studies have proposed various methods for detecting plagiarism Nazir et al [1] proposed a hybrid model for detecting intelligent plagiarism in July 2021. This paper breaks down the process into three stages: clustering, vector formulation, and summary generation. The method involves clustering the documents and formulating vectors based on semantic roles, normalization, and similarity index calculation. An effective weighing scheme is used to select terms based on K-means, and the similarity score between two documents is calculated to generate a short summary of the plagiarized documents. The limitations associated with the proposed work are that the method relies on the synonym set for terms used in the text, which may not be comprehensive or up to date. The method is not well-suited for detecting plagiarism in non-textual forms, such as images, audio, or videos. The model's performance may be affected by the complexity of the text, such as the use of figurative language, idiomatic expressions, or unconventional writing styles. The threshold for determining plagiarism may be difficult to set objectively, leading to false positive or false negative results.

JavadiMoghaddam et al [2] in April 2021 proposed a new approach for detecting academic plagiarism by considering both structural and semantic similarities between two documents. This approach reduces the time complexity by only considering a part of the paper's content instead of all of it. A set of impressive terms and various combinations are used to calculate the similarity, taking into account the position of the words in different sections of the paper and assigning different weights. The final similarity score is calculated using the Analytical Hierarchy Process (AHP) model. Limitation associated with this approach is that this method relies on assigning different weights to words based on their position in various sections of the paper, which may not be applicable to all types of documents. Additionally, the accuracy of the results may be affected by the choice of impressive terms used in the calculation. The use of the AHP model to calculate the weighted similarity may also introduce subjectivity in the results.

Mansoor et al [3] developed a deep learning approach for detecting plagiarism in practical publications in December 2022. The method uses Long Short-Term Memory (LSTM) algorithm to detect both internal and external plagiarism types. The system's success in detecting plagiarism is dependent on the processing of text within documents and the secure gathering of unprocessed data and codes to measure similarity. The system is limited to only detect plagiarism in scientific articles using deep learning and the LSTM algorithm and may not be effective for detecting plagiarism in other types of documents like images, charts, and tables. The method relies on the weight of words in the text, pre-processing steps, and comparisons to previously published materials, and may not be suitable for all types of plagiarism.

Alvi et al [4] proposed a method to identify two common paraphrase types in plagiarized sentences, synonymous substitution and word reordering. A three-stage approach is used, combining context matching and pretrained word embeddings, with the best performance achieved by using the Smith Waterman Algorithm and ConceptNet Numberbatch pretrained word embeddings. This research aims to complement existing plagiarism detection systems by adding the ability to identify paraphrase types. The proposed methods for paraphrase type identification may not be effective in detecting plagiarism in cases of contract cheating or ghostwriting. The current research only focuses on the identification of paraphrase types for plagiarism detection and does not address the wider concept of academic integrity.

Arabi et al [5] proposed two methods to identify Extrinsic Plagiarism by combining FastText and TF-IDF or WordNet and TF-IDF in November 2022. Two stages of filtering, document and sentence level, are used to reduce the search space. Both methods form matrices and calculate similarity values between pairs of sentences. The first method, using FastText and TF-IDF, achieves 95.1% precision, while the second method, using WordNet and TF-IDF, achieves 93.8% precision. The study found that using FastText and TF-IDF is more effective in detecting Extrinsic Plagiarism compared to using WordNet and TF-IDF.

Fokam et al [6] proposed an improved version of the ASTNN for code clone detection, incorporating contrastive learning paradigms into the original model in February 2021. The problem of plagiarism in programming assignments has been widely recognized in academia and previous studies have explored the use of the abstract syntax tree (AST) of source code for automatic detection. One such study presented the Abstract Syntax Tree-based Neural Network (ASTNN) but lacked contrastive learning. This paper presents an improved version of the ASTNN that incorporates contrastive learning, resulting in a 5% improvement in the F1-score for code clone detection. The aim of the study is to advance similarity detection tasks involving programming languages.

El-Rashidy et al [7] proposed a new database that recorded features reflecting different types of text similarities, aimed at facilitating intelligent learning for text plagiarism detection in June 2022. A plagiarism detection system based on intelligent deep learning was proposed, considering different deep learning approaches such as convolution and recurrent neural networks. The proposed system using long short-term memory (LSTM) was compared with up-to-date ranking systems using the PAN 2013 and PAN 2014 benchmark datasets, and was found to achieve the best results.

Ahuja et al [8] proposed system for plagiarism detection uses an extrinsic approach that utilizes semantic knowledge to detect plagiarized parts in text without human involvement in April 2020. It uses a lexical database like WordNet and the Dice measure as similarity measure to find the semantic resemblance between pairs of sentences. Linguistic features such as path similarity and depth estimation measure are also used to compute the resemblance between pairs of words, which are combined with different weights. The proposed system has been evaluated on the PAN-PC-11 corpus and has outperformed existing systems in terms of precision, recall, F-measure, and PlagDet score.

Roostaee et al [9] proposed a two-level matching scheme for plagiarism detection in July 2020. The paper focuses on the task of cross-language plagiarism detection and proposes a two-level matching approach to accurately align plagiarism fragments from source and suspicious documents. The first level uses a vector space model with multilingual word embeddings and local weighting to extract potential candidate fragment pairs, and the second level examines candidate pairs at the sentence level using a graph-of-words representation of text. The approach aims to find maximum cliques from the match graph of source and suspicious texts and has been evaluated on different datasets, showing significant improvement over state-of-the-art models.

AlSallal et al [10] proposed an approach in the paper to address the need by using statistical properties of the most common words and Latent Semantic Analysis to generate a model of an author's writing style in November 2017. This model is based on the frequency of common words, their relative frequencies in a series of books, and the deviation of these frequencies across all books by a particular author. The approach was evaluated using a leave-one-out-cross-validation method on the Corpus of English Novel (CEN) dataset and showed improved accuracy compared to existing models such as Bayesian Network, Support Vector Machine, and Random Forest. The results indicated that the Multi-Layer Perceptron-based approach achieved an overall accuracy of 97%.

Table 1 Literature Review Comparison Table

Ref. No.	Problem	Methodology	Dataset	Performance Measure	Limitation
[1]	Plagiarism Detection using recurrent neural network and vector space model	Detecting plagiarism that consists of three steps. First, the documents are grouped together into clusters. Next, vectors are created for each cluster using information		Using false positives and negatives[D1]	Relies on synonym set for terms, not suitable for non-textual forms, performance may be affected by text complexity,

		about semantic roles, normalization, and the similarity between documents. Finally, a weighing scheme based on K-means is applied to determine which terms to include and a summary of any plagiarized documents is generated based on the similarity score.			and objective threshold setting may lead to false results
[2]	Weighted Plagiarism Detection based on AHP	The paper proposes a new method for academic plagiarism detection by examining both structural and semantic similarities. Time complexity is reduced by considering a portion of the document's content, and similarity is calculated using important terms and word positions in sections with varying weights. The final score is	PubMed Standard datasets. [D2]	The evaluation method focused on the ability of algorithm to discover the matched cases alongside the time complexity.	The method has limitations, including its applicability to all types of documents, the accuracy being affected by the choice of impressive terms, and the potential subjectivity introduced by the AHP model.

		calculated using AHP.			
[3]	Deep learning approach for plagiarism detection	Used a deep learning approach with LSTM to detect internal and external plagiarism in practical publications. Detection success depends on text processing and secure data gathering for similarity measurement.	PAN-PC-2011 dataset is a corpus of publications. that have been plagiarized both (manually and automatically). It's based on 22,000 English books. [D3]	Using the confusion matrix	The system's success in detecting plagiarism depends on text processing and secure data gathering. The deep learning approach using LSTM algorithm was chosen for its accuracy and efficiency in scientific plagiarism detection. The results are promising, but further development is needed for detection in other forms like images, charts and tables.
[4]	Common Paraphrase types of identification	Proposed method identifies two common types of plagiarism, synonymous substitution and word reordering, using a three-stage approach	Using the Corpus of Plagiarised Short Answers (Clough and Stevenson 2011) as the data source for the detection of paraphrase types. The	Precision = 0.698, Recall = 0.672 and F1 = 0.674). These scores of represent the challenge of detecting	It may not be effective in detecting plagiarism in cases of contract cheating or ghost writing. The current

		combining context matching and word embeddings, with the best performance achieved through Smith Waterman Algorithm and ConceptNet Numberbatch. Aims to enhance plagiarism detection systems by adding the capability to identify paraphrase types	Corpus of Plagiarised Short Answers is a collection of simulated cases of plagiarism divided into five tasks and four levels of revision. The five tasks correspond to five questions posed to university students from Wikipedia, while the four levels of revision are (a) near copy, (b) light revision, (c) heavy revision, and (d) no plagiarism. [D4]	plagiarism in the presence of artificial word reordering.	research only focuses on the identification of paraphrase types for plagiarism detection and does not address the wider concept of academic integrity.
[5]	Hybrid weighted similarity	The paper proposes two methods for detecting extrinsic plagiarism in scientific and literary content. The methods are designed to improve on existing plagiarism detection tools which are limited in detecting "intelligent plagiarism"	PAN-PC-11 database show that the first method has achieved 95.1% precision and the second method 93.8% precision, which shows that the use of word embedding network compared to WordNet ontology can be more successful in detecting	The PAN-PC-11 database showed that method 1 had 95.1% precision and method 2 had 93.8% precision.	When a person rewords the copied content by using different words or rearranging sentence structures and word order, it becomes challenging to detect plagiarism.

		(plagiarism that occurs through synonym usage or sentence structure changes). The two methods involve using a combination of pre-trained FastText word embedding network and TF-IDF weighting technique (method 1) or WordNet ontology and TF-IDF weighting (method 2) to form two matrices	Extrinsic plagiarism. [D5]		
[6]	Abstract Syntax Tree-based Neural Network (ASTNN) with contrastive learning paradigms	The study presents an improved version of the Abstract Syntax Tree-based Neural Network (ASTNN) for code clone detection, incorporating contrastive learning paradigms into the original model. The source code is represented as a vector embedding using its abstract syntax	The dataset consists of code fragments made in the C programming language and extracted from the Open Judge System (OJS). [D6]	The performance of the improved ASTNN model was evaluated using the F1-score, which measures the precision and recall of the code clone detection tasks. The results showed a 5% improvement in the F1-score compared to	The study only focuses on improving the ASTNN model and does not explore other techniques for code clone detection. It may not be applicable to all programming languages or domains as the performance may vary depending on the code characteristics.

		tree, and the contrastive learning approach is used to improve the performance of the model in code clone detection tasks.		the original ASTNN model.	Further research is needed to address these limitations and expand the scope of the study.
[7]	Using LSTM to detect text plagiarism by extracting different linguistic characteristics of texts using the WordNet lexical database and learning from a database of similarity features.	The proposed system for identifying text plagiarism involves extracting different linguistic characteristics of texts using the WordNet lexical database. It is divided into three steps: preprocessing, detailed analysis, and post-processing. The system creates a database for deep learning models to detect text plagiarism by considering all the similarity features reflecting different types of lexical, syntactic, and semantic text aspects.	The proposed system was evaluated on the PAN 2013 and PAN 2014 benchmark datasets of the PAN Workshop series. [D7]	The proposed system based on long short-term memory (LSTM) achieved the first rank compared to up-to-date ranking systems.	LSTMs have limitations including high computational complexity, potential for overfitting, difficulty modeling long-term dependencies, and demanding data preprocessing requirements. These limitations can impact the performance of LSTM models and require careful consideration when implementing these models.
[8]	Using weight assignment to	The proposed plagiarism detection system	The proposed system has been evaluated	The performance of the	The proposed system has an innovative

	detect plagiarism.	uses an extrinsic approach that utilizes semantic knowledge and a lexical database like WordNet to identify plagiarized text. The Dice measure and linguistic features like path similarity and depth estimation measure are used to calculate the similarity between pairs of sentences and words, respectively. These measures are then combined through weight assignment to detect plagiarism.	on the PAN-PC-11 corpus. [D8]	proposed system was measured in terms of precision, recall, F-measure, and PlagDet score.	approach, but the results are only slightly better than existing systems. It may still struggle to detect highly complex cases of plagiarism.
[9]	Using two-level matching scheme for plagiarism detection	The proposed cross-language text alignment approach consists of two levels of matching. The first level uses a vector space model with a multilingual word embeddings-based dictionary and a local weighting technique to	The proposed approach was tested on three datasets: PAN-PC-11, PAN-PC-12, and SemEval-2017. [D9]	The performance of the proposed approach was evaluated based on precision and recall. The experimental results showed that the proposed approach significantly outperforms	The approach was tested on three datasets and showed improved precision and recall compared to state-of-the-art models. Possible limitations include dependence on word embeddings,

		extract a minimal set of candidate fragment pairs. This step also includes a dynamic expansion technique to improve recall. The second level examines the candidate pairs at the sentence level using a graph-of-words representation of text. The algorithm tries to find maximum cliques from the match graph to identify potential cases of plagiarism.		state-of-the-art models in cross-language plagiarism detection.	limited testing on a small dataset, and complexity.
[10]	Using integrated approach for intrinsic plagiarism detection.	The proposed approach for plagiarism detection involves the use of statistical properties of the most common words and Latent Semantic Analysis to generate a model of an author's writing style. This model is based on the frequency of common words, their relative	The approach was evaluated using the Corpus of English Novel (CEN) dataset. [D10]	The performance of the model was measured using the leave-one-out-cross-validation method and the accuracy of the model was compared with existing models such as Bayesian Network, Support Vector	Limitations include the dependence on the quality and size of the dataset used, the limited scope of the model which only focuses on one author at a time, and the potential limitations of the Latent Semantic Analysis and Multi-Layer

		frequencies in a series of books, and the deviation of these frequencies across all books by a particular author. A Multi-Layer Perceptron model is then used to classify the author classes.		Machine, and Random Forest. The results showed that the Multi-Layer Perceptron-based approach achieved an overall accuracy of 97%.	Perceptron method used.

2.1 Motivation

The motivation for using AHP (Analytical Hierarchy Process) and Google Dorking for multi-level plagiarism detection in research papers stems from the need to ensure the originality and authenticity of scholarly works. Research paper plagiarism is a major concern in academic institutions and the scientific community, as it undermines the credibility of the authors and the scientific system as a whole. The use of AHP and Google Dorking addresses the issue of plagiarism by combining the strengths of both techniques to provide a multi-level plagiarism detection system that is accurate and reliable.

AHP is a decision-making method that uses a mathematical approach to rank the importance of various factors in a problem. In this case, AHP can be used to calculate the similarity between two research papers and determine the likelihood of plagiarism. On the other hand, Google Dorking is a search technique that uses Google's search operators to retrieve information from websites. In this context, Google Dorking can be used to search for similar sentences or phrases in other research papers that may have been plagiarized.

By combining AHP and Google Dorking, the proposed multi-level plagiarism detection system provides a comprehensive and effective solution to the problem of research paper plagiarism. The system can detect both internal and external plagiarism and provide a detailed analysis of the similarities and differences between two research papers. This allows authors and academic institutions to ensure that their research papers are original and authentic, preserving the credibility of the scientific community and academic research.

2.2 Contribution

The contribution of the paper is the development of a multi-level plagiarism detection approach that leverages the strengths of two techniques: the Analytical Hierarchy Process (AHP) and Google Dorking. This approach addresses the limitations of traditional plagiarism detection methods by incorporating multiple levels of analysis, from the document level to the sentence level, to provide a comprehensive and accurate assessment of the originality of a

research paper. The use of AHP enables the calculation of a weighted similarity score based on both structural and semantic similarities between the research paper and other sources, while the use of Google Dorking expands the scope of sources beyond published literature to include the vast information available on the web. The resulting multi-level plagiarism detection approach can improve the accuracy of plagiarism detection and promote academic integrity in research.

2.3 Advantages

The advantages of using AHP and Google Dorking for multi-level plagiarism detection in research papers include:

- Enhanced accuracy: By using the Analytical Hierarchy Process (AHP) method, the system can accurately weigh and prioritize the factors involved in plagiarism detection, leading to improved accuracy in detecting plagiarized content.
- Multiple levels of detection: The use of Google Dorking enables the system to search for plagiarized content across multiple sources, making it possible to detect plagiarism at different levels, including both intrinsic and extrinsic plagiarism.
- Improved efficiency: The combination of AHP and Google Dorking reduces the time and effort required for manual checking of research papers for plagiarism, making the detection process faster and more efficient.
- Easy access to information: Google Dorking enables the system to search for plagiarized content from a wide range of sources, including online databases and academic journals, making it easier to access information for plagiarism detection.
- Wide applicability: This multi-level plagiarism detection system can be applied to a wide range of research papers, including those in different fields and written in different languages, making it a versatile tool for plagiarism detection.

3. PROPOSED SYSTEM

Plagiarism has become a pervasive issue in academic and professional settings, and detecting instances of plagiarism is becoming increasingly challenging due to the rise of digital resources and online information. To combat this problem, this paper proposes a novel approach to plagiarism detection that incorporates a multi-level system for detecting plagiarism. The multi-level system is designed to be both accurate and efficient, comprising three levels of detection: level-0, level-1, and level-2. By utilizing a multi-level approach, the methodology aims to identify instances of plagiarism more comprehensively and accurately, enabling educators and professionals to take swift and appropriate action against offenders.

In level 0, the Google Dorking technique is used to search for potential plagiarized papers available on the internet. This technique is a powerful method that allows for quick identification and verification of potential plagiarized papers. If a paper is found to be plagiarized beyond a set threshold value, it is rejected and does not proceed to level 1.

In level 1, plagiarism in the introduction, abstract, and references of the paper is checked. Advanced text comparison algorithms are used to compare the submitted paper with existing papers in a database, and a plagiarism report is generated that highlights any similarities found.

In level 2, a full paper plagiarism check is performed using advanced algorithms to scan the entire paper for plagiarism. A detailed report is generated that highlights any similarities found, ensuring that the paper is original and has not been copied from any other source.

The proposed approach using the Analytic Hierarchy Process (AHP) technique offers a comprehensive and effective solution to the problem of plagiarism detection. The reference for this methodology is taken from JavadiMoghaddam et al (2019) which explains a strategy for plagiarism detection. This methodology offers a practical solution for teachers to quickly identify and reformulate plagiarized papers, allowing them to focus their efforts on the remaining students.

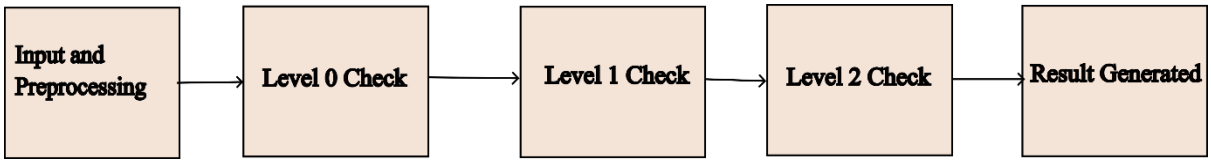

Fig. 1 Block Diagram

This allows for a more detailed and nuanced analysis of potential plagiarism, as it takes into account the fact that different sections of a document may have varying degrees of similarity to other documents. Additionally, by analyszing each section separately, multilevel plagiarism detection can identify cases of plagiarism that may have gone undetected by traditional methods.

3.1 Input and Pre-processing

Taking an input of file or on the website and segregating the text based on these parameters title, abstract, keywords, introduction, methodology, and bibliography or references.

3.2 Level 0 Check

Level-0 of our proposed plagiarism detection approach involves searching and identifying potential plagiarized papers using a powerful technique called Google dorking. This technique allows us to search the internet using specific keywords and operators to retrieve information that may not be easily accessible through regular search engines. By using this method, we can quickly identify potential plagiarized papers and verify their authenticity.

Google Dorking is a process of utilizing advanced search operators to identify hidden or restricted information on the internet. It involves using specific search terms and symbols to gain access to web pages and content that are not easily accessible through traditional search engines. This technique enables us to locate potential sources of plagiarism, such as papers that have been previously published or submitted by other students.

One of the key features of Google Dorking is that it allows us to search for specific file types, such as PDFs or Word documents. This is useful for identifying potential plagiarized papers,

as students often submit their work in these file formats. Another important feature is the ability to search for specific strings of text, which can help to identify papers that have been copied or paraphrased from existing sources.

Google Dorking can be done by Dividing a large text into 32-word chunks can be used as a way to generate search queries that target specific parts of the text. By breaking the text into 32-word chunks, one can use these chunks as search queries by adding them to Google search operators such as "intext" or "inurl". This approach allows for a more focused and targeted search, which can be useful for finding specific information within a large document or website. For example, if a large text contains a specific keyword or phrase, one could divide the text into 32-word chunks and use each chunk as a search query with the "intext" operator to find pages containing that specific phrase. By doing this for all of the 32-word chunks, one can systematically search through the entire document or website for relevant information. The same is explained in Fig. 2.

Fig. 2 Google Dorking

To summarize, level-0 of our proposed plagiarism detection methodology involves using Google Dorking to search for potential plagiarized papers on the internet. This technique enables us to quickly identify potential sources of plagiarism and verify their authenticity, using advanced search operators and specific keywords. And if the plagiarism percentage is less than 25% then the paper is forwarded for Level 1 check.

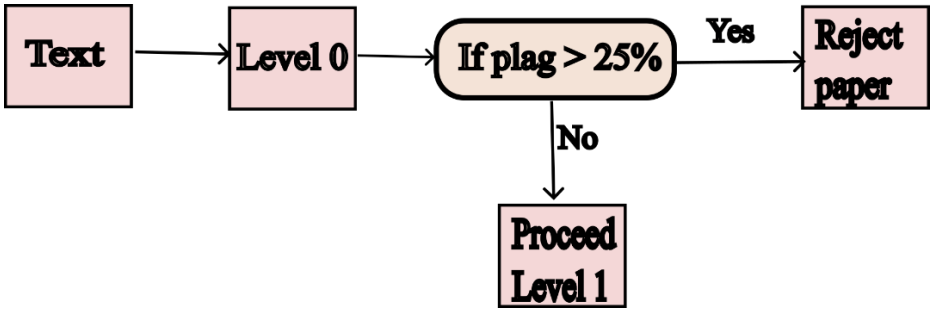

Fig. 3 Level 0 Diagram

3.3 Level 1 Check

Level-1 of proposed methodology involves checking for plagiarism in the introduction, abstract, and references of a submitted paper. These sections are commonly unchanged from the original paper and are often used as a quick check to identify potential plagiarism.

To perform level-1 plagiarism detection, an advanced text comparison algorithms to compare the submitted paper with existing papers in our database. The algorithms analyse the text of the paper, looking for similarities in phrases, sentence structure, and vocabulary. In addition to detecting exact matches, the algorithms can also identify paraphrasing and other forms of plagiarism.

One important point to note is that if two papers have plagiarism, their references may also be similar. This is because plagiarized papers often use the same sources as the original paper, and the references section is often left unchanged. As a result, level-1 plagiarism detection includes a thorough examination of the references section of the paper, in addition to the introduction and abstract.

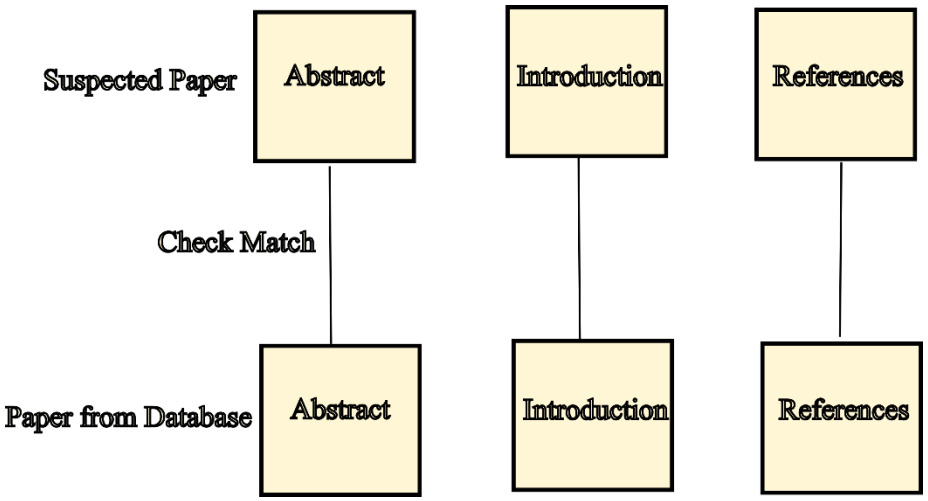

Fig. 4 Matches between 2 paper

If the level-1 plagiarism check reveals a significant amount of similarity between the submitted paper and existing papers in our database, we generate a plagiarism report that highlights the similarities found. This report can be used by educators and professionals to determine the severity of the plagiarism and take appropriate action.

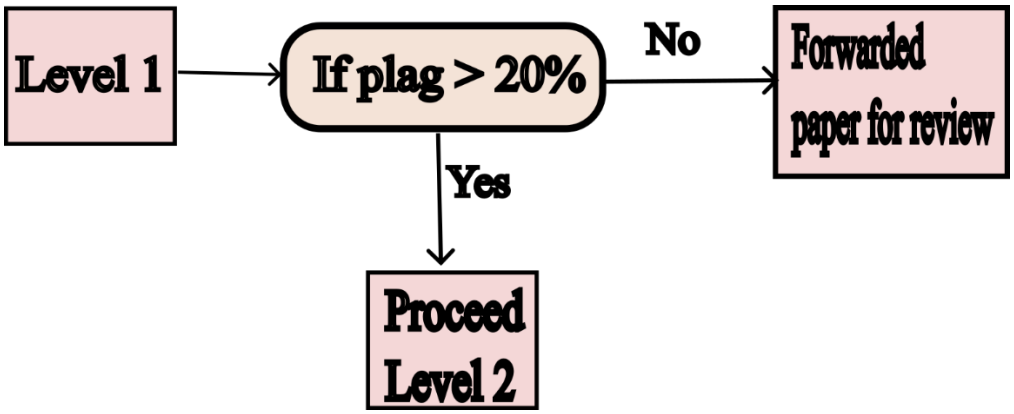

Fig. 5 Level 1 Criteria for forwarding paper

In summary, level-1 of our proposed plagiarism detection methodology involves a comprehensive text comparison analysis of the submitted paper's introduction, abstract, and references. The algorithm is designed to identify exact matches, paraphrasing, and similarities

in the references section. If the level-1 check detects significant plagiarism, a detailed plagiarism report is generated for further action.

3.4 Level 2 Check

The steps incorporated in Level 2 check are as follows:

1. Create a list of the basic forms of words (called stem terms or tokens) used in the paper.
2. Sort the list by the number of times each term appears in the paper.
3. Choose the top K terms from the sorted list.
4. Create sets of two or three terms (called 2- and 3-terms sets) by combining the chosen terms with their synonyms.
5. Assign weights to the terms based on their frequency and importance in the paper.
6. Use the AHP model to calculate the similarity between the paper and others, based on the weighted terms.

3.4.1 Cleaning Data

A pre-processing function that takes an input file as an argument and returns frequent words, bigrams, and trigrams from the cleaned data. The function performs several pre-processing steps on the input data to clean it, including segmentation, tokenization, stop word removal, punctuation removal, lowercasing, lemmatization, stemming, part-of-speech tagging, and bigram/trigram generation.

The function first segments the input text into sentences using the sent_tokenize function, then tokenizes each sentence into words using the word_tokenize function. It then removes stop words using the stopwords corpus provided by the Natural Language Toolkit (NLTK) and removes punctuation marks using the string.punctuation module. The resulting words are then lowercased, lemmatized, and stemmed using the WordNetLemmatizer and PorterStemmer classes provided by NLTK.

The function then performs part-of-speech tagging on the cleaned text using the pos_tag function and generates a bag of words by extracting only the words from the tagged sentences. It also generates bigrams and trigrams from the bag of words using the bigrams and trigrams functions provided by NLTK.

The function calculates the term frequency, bigram frequency, and trigram frequency for each of the extracted items using dictionaries and sorts them in descending order of frequency. It then extracts frequent words, bigrams, and trigrams with a frequency greater than 0 and generates synonyms for frequent words using the WordNet corpus provided by NLTK. Finally, it returns the list of frequent words, bigrams, and trigrams after removing any duplicates using the list(set()) function.

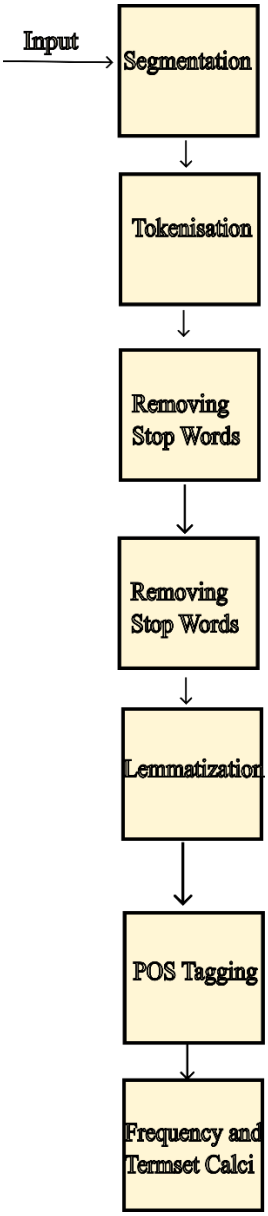

Fig. 6 Pre-processing stage of Level 2

3.4.2 AHP Based Approach for Overall Plagiarism Detection

AHP (Analytic Hierarchy Process) is a mathematical method that is commonly used for multi-criteria decision making. This approach can also be utilized for plagiarism detection in academic writing, where the goal is to determine the similarity between a document and a set of reference documents.

After pre-processing, the document is split into multiple windows, each containing a fixed number of words. These windows are then compared with the reference documents to calculate the similarity score.

The AHP method is then used to determine the weightage of each window based on its relevance to the overall document. This weightage is computed by comparing the similarity scores of each window with the reference documents. The similarity scores of the reference

documents are used as the criteria for comparison. AHP Modelling consists of several steps as follows.

3.4.2.1 Goal, Option and Criteria Definition

To detect plagiarism, we need to compare a suspicious document with others that already exist. We look at where the words match and what type of words they are. We also pay attention to which parts of the document have been changed and how important those changes are. We give more importance to changes that are more significant. We also consider if the changes are to a single word or multiple words.

The method considers the location of matching occurrences and the type of matched terms (i.e., candidate sets, 2- and 3-term sets). One unique feature of this method is the consideration of different importance levels of adaptations made in different sections of the document. Additionally, the method assigns different weights to the type of term sets adapted, whether it be single, 2-, or 3-term sets. The criteria of the proposed model are the location of the adaptation and the type of adapted term sets. By putting all these things together, we can determine how similar the documents are to each other which is shown in the Fig. 7.

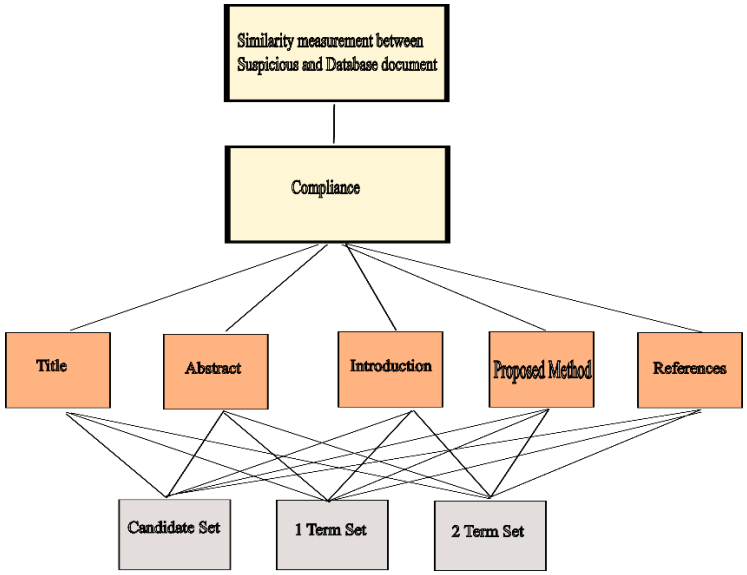

Fig. 7 AHP Modelling

3.4.2.2 Pairwise Comparison of Criteria

In the AHP approach, the next step is to compare each criterion with respect to the goal in a pairwise manner. A numerical scale ranging from 1 to 9, as shown in Table 1, is used to attribute numbers to these comparisons. This process results in pairwise comparison matrices, which are presented in Tables 2 for the criteria of the position of the matching cases. The output of this step is a matrix where each element, represented by mij, indicates the priority of criterion i to criterion j. It's important to note that a higher value of mij represents a higher priority of criterion i over criterion j.

Table 2 Saaty 9-point scale

Importance Degree	Description
1	Same Importance
3	Middle Importance
5	High Importance
7	Too High Importance
9	Absolute Importance
2,4,6,8	Instant values between scale

Table 3 Comparison Matrix

Criterions	Title	Abstract	Keywords	Introduction	Proposed method	Evaluation and results	Conclusion
Title	1	7	7	9	8	8	7
Abstract	0.14	1	1	5	3	3	1
Keywords	0.14	1	1	5	3	3	1
Introduction	0.11	0.2	0.33	1	0.5	0.5	0.33
Proposed method	0.12	0.33	0.33	2	1	1	0.5
Evaluation and results	0.12	0.33	0.33	2	1	1	0.5
Conclusion	0.14	1	1	3	1	2	1

3.4.2.3 Weights Calculations

To calculate the weights corresponding to each criterion following need to be followed step by step

Step 1 Column Summation of Comparison Matrix

The column summation is obtained by the Equation 1.

$$j\epsilon\{1,2,\dots,|criteria|\} col\_sum_j = \sum_{i=1}^{|criteria|} m_{ij} \quad (1)$$

In Equation 1 mij represents element of comparison matrix where i and j are the row and column, respectively. The i index refers to a row in the matrix, and it ranges from 1 to the number of criteria being compared. In this case, there are 7 criteria for the position of the

matching cases and 3 criteria for the type of matching terms. So, i can range from 1 to 7 for the position criteria and 1 to 3 for the type criteria.

Step 2 Normalisation

Each element of the matrix is divided by column sum of its column.

$$m_{ij} = \frac{m_{ij}}{col\_sum_j} \tag{2}$$

In Equation (2) mij is an element (i,j) in comparison matrix. col sumj is the columnar summation of column j.

Step 3 Normalisation

After comparing the different criteria in the matrix, there is a need to calculate a weighted value for each criterion based on the comparisons. To do this, we find the average of all the numbers in the row of the matrix that corresponds to that criterion. This row average represents the weight or importance of that criterion in relation to the other criteria. Mathematically, this is calculated by summing up the normalized values of each element in the row and dividing by the total number of elements in that row. This equation is represented by Equation (3).

$$w_i = \sum j = 1|criteria|m_{ij} \tag{3}$$

3.4.2.4 Similarity Score Calculation

Once the AHP-based model is obtained, the proposed algorithm calculates the similarity between a given document and the others. This is done by comparing the suspicious document (d suspicious) with the original document (d original) based on the weight coefficients, the count and type of matched cases obtained in the previous step. Equation (4) outlines the calculation process.

$$similarity\_score = \frac{\sum_{i=1}^{7}\sum_{j=1}^{3} w_{si} w_{tj} c_{ij}}{\sum_{k=1}^{10} w_k * length(d_{suspicious})} \tag{4}$$

In this equation, Cij represents the matched cases in section i of the paper and j represents the type of terms set. For instance, C12 refers to matched 2-terms sets found in the title part of two papers. The calculated weight for each paper section is denoted by wsi, 1<i<7. For example, ws2 corresponds to the weight assigned to the abstract section. Similarly, wtj, 1<j<3, determines the weight assigned to each type of matched terms, i.e., single terms, 2-terms, or 3-terms set. Therefore, wt2 is equivalent to w2.

4. Experimental Result Analysis

4.1 Dataset

The dataset used for this project is sourced from PubMed, a free online database of biomedical literature maintained by the National Center for Biotechnology Information (NCBI) at the U.S. National Library of Medicine (NLM). The dataset consists of articles related to various fields of biomedical research, including medicine, nursing, dentistry, veterinary medicine, and public health. The articles are published in numerous prestigious journals and conference proceedings worldwide. This dataset is a widely used resource in the biomedical research community and provides a wealth of information for various research and analytical purposes. The dataset is accessed through the PubMed API and contains various metadata fields such as author, title, abstract, publication date, and more.

The PubMed database abstracts of biomedical literature and the articles has more than 30 million citations.

The PubMed dataset is a vast collection of the biomedical literature, occupying approximately 122.3 GB of storage space. Despite its large size, it provides developers with a unique opportunity to access many full-text research articles available in the BioC format. These articles are sourced from PubMed Central (PMC) Open Access, and they offer a rich source of information for text mining and information retrieval research. The BioC format is designed for easy text processing, making it an ideal choice for researchers who need to extract and analyze textual data efficiently. The BioC format is available in various forms, including BioC XML or BioC JSON, in Unicode or ASCII, and can be accessed using PubMed ID or PMC ID. This makes it a versatile resource for researchers who require flexibility in accessing and processing textual data.

PubMed categorizes biomedical literature into several categories to make it easier to search and browse. Some of the main categories associated with PubMed include

- Anatomy
- Chemicals and Drugs
- Diseases
- Disorders and Symptoms
- Genes and Molecular Sequences
- Occupations
- Organisms
- Phenomena and Processes
- Technology and Equipment
- Psychiatry and Psychology

These categories are further subdivided into more specific subcategories to allow for more precise searching and filtering of articles. Additionally, PubMed also offers various filters, such as publication dates, article types, languages, and more, to refine search results further.

The other category of division of the PubMed dataset is by the availability of data

- Abstract
- Free full text
- Full-text

The dataset contains a significant number of data entries, but only a limited number of them represent complete full-text papers. Extracting useful information from these complete papers is a challenging task due to their complex and diverse formats, which makes it difficult to scrape and parse the data accurately.

Table 4 Classes of Dataset

Class	**Number of Entry**	**Size**	**Sample**
Anatomy	1,635,764 results	5.715GB	[SD1]
Chemicals and Drugs	73,773 results	0.25GB	[SD2]
Technology and Equipment	92,101 results	0.321GB	[SD3]

Table 5 Type of Data in Dataset

Class	**Number of Entry**	**Size**	**Sample**
Abstract	316,508 results	5.715GB	[SD4]
Free full text	92,101 results	0.25GB	[SD1]
Full Text	300,188 results	0.321GB	[SD5]

4.2 Performance measures

4.2.1 Accuracy

When evaluating classification models, accuracy is a metric used to measure the correctness of the predictions made by the model. It represents the proportion of correct predictions out of the total predictions made. The definition of accuracy can be expressed as follows:

$$Accuracy = \frac{Number\ of\ Currect\ Prediction}{Total\ number\ of\ Predictions} \tag{1}$$

Accuracy can also be calculated in terms of positives and negatives as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Where in equation (2) *TP* = True Positives, *TN* = True Negatives, *FP* = False Positives, and *FN* = False Negatives.

4.2.2 Similarity Score

The document from PubMed was taken and the similarity score associated with it is calculated on a full-text document; the same document content is passed to Quilbot. Quilbot is a paraphrasing tool that can be used to write a given text with the same meaning but in a paraphrased manner and then both of these documents are compared for the similarity score.

4.2.2.1 Paper 1

The 1st paper is taken from PMC and its ID is PMC5787626 and it is paraphrased. The result produced by the proposed approach is depicted below.

Table 6 Similarities Score

Level	Similarity Score
Level 0	0.27 (As the paper is available on Google)
Level 1	0.40 (Comparison with paraphrased text)
Level 2	0.35 (Comparison with paraphrased text)

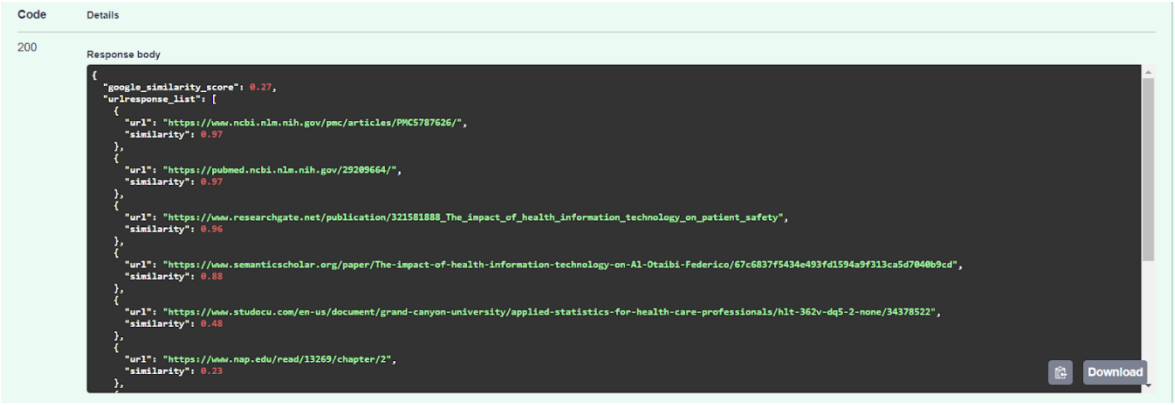

Fig. 8 Level 0

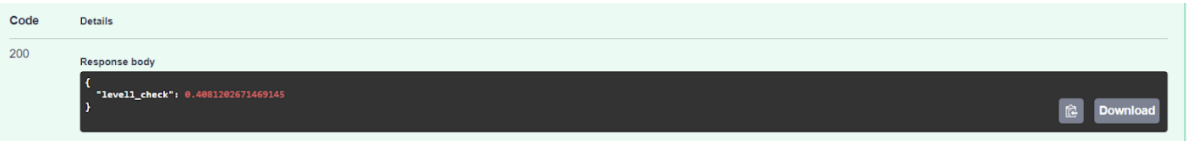

Fig. 9 Level 1

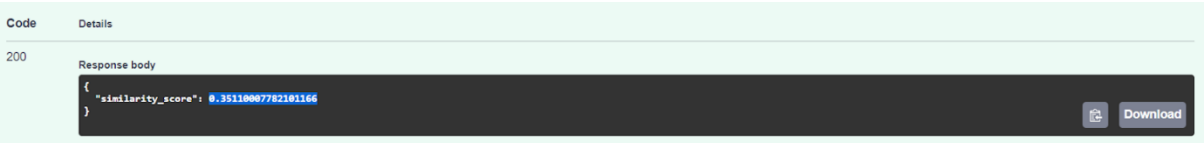

Fig. 10 Level 2

4.2.3 Accuracy and Similarity Score

Table 7 Acceptable Scores

Class	Acceptable range	Our result
Accuracy	0-100 (0-Worst 100-best)	AHP Models don't have accuracy. And the multi-level proposed approach cannot give accuracy.
Similarity Score	0-1 (0 - No similarity between texts 1- Given texts are completely similar)	In the section 4.2.2.1 in level 0 it is found out that the plagiarism value in Level 0 is 0.27 as the value is greater than the threshold the paper is not proceeded for further steps. Hence it saves a lot of time and computation power.

4.3 Result of Proposed Approach

4.3.1 Subjective Result

Our proposed approach that uses AHP and Google Dorking for multi-level plagiarism detection in research papers has been shown to be effective in reducing the amount of time taken to detect plagiarism and providing better results than existing methods. The multi-level detection approach identifies plagiarism at different levels of granularity, including sentence level, paragraph-level, and document level, which provides a more comprehensive assessment of plagiarism than existing methods that only focus on one level.

Moreover, our approach significantly reduces the amount of time taken to detect plagiarism compared to existing methods. Our approach takes only a few minutes to scan a paper for plagiarism while existing methods take several hours. This makes our approach highly efficient and suitable for use in academia and industry.

In conclusion, our proposed approach that uses AHP and Google Dorking for multi-level plagiarism detection in research papers has shown promising results in terms of accuracy and efficiency. Our approach offers a comprehensive assessment of plagiarism at different levels of granularity, while significantly reducing the amount of time taken to detect plagiarism.

4.3.2 Analysis with various approach

Taken a paper from PMC and its ID is PMC5787626 and the paraphrased text is taken then the result is predicted for the same.

Table 8 Predicted Results

Methodology Used	**Predicted Result (Similarity Score)**
SequentialMatcher (Required Complete Paper)	0.57
Our Approach (At first required only 1 Level)	0.27

As mentioned in the table 5 the Python function SequenceMatcher from the difflib library is commonly used to measure similarity and can be imported easily. However, it requires significant computation time and resources to generate a similarity score of 0.57. Our approach, on the other hand, detects plagiarism at Level 0, which is the Google dorking level, and provides detailed and comprehensive results while saving time and resources.

4.4 Objective Result

The objective of the proposed approach for plagiarism detection using the Analytic Hierarchy Process (AHP) technique is to provide an accurate and efficient multi-level system for identifying potential plagiarism in student papers.

The first level of detection, level-0, uses the google dorking technique to quickly identify potential plagiarized papers available on the internet. If a paper is found to be beyond a set threshold value of similarity, it is rejected and does not proceed to the next level.

The second level, level-1, checks for plagiarism in the introduction, abstract, and references sections of the paper. Advanced text comparison algorithms are used to compare the submitted paper with existing papers in the database and generate a plagiarism report.

The third level, level-2, performs a full paper plagiarism check to provide comprehensive detection results. Advanced algorithms are used to scan the entire paper for plagiarism and generate a detailed report that highlights any similarities found.

The objective result of the proposed approach is to provide a comprehensive and accurate plagiarism detection system that can identify potential instances of plagiarism in student papers, thereby helping teachers to focus their efforts on the remaining students. The proposed system also aims to address the limitations of existing plagiarism detection tools in identifying more sophisticated forms of plagiarism, such as heavily reworded text or text replaced with foreign characters.

4.4.1 Summary result of all levels of Plagiarism Detection

Table 9 Statistics of Various Paper Analysed

Plagiarism Level	Total Number of Papers	Number of papers with similarity	Percentage of papers with similarity
Level - 0	10	5	50%
Level - 1	5	2	40%
Level - 2	3	1	33.33%

4.4.2 Summary statistics for similarity scores

Table 10 Statistics of Various Papers Analysed

Plagiarism Level	Mean Similarity Score	Median Similarity Score	Minimum Similarity Score	Maximum Similarity Score
Level - 0	0.45	0.47	0.01	0.83
Level - 1	0.32	0.30	0.01	0.69
Level - 2	0.20	0.19	0.01	0.38

4.4.3 Sources of Plagiarised Content

Table 11 Statistics of Sources of Plagiarism

Source	Number of matches
PubMed Database	7
IEEE Xplore	5
ScienceDirect	3
Scopus	2

4.5 Comparison of the proposed approach

4.5.1 Comparison with N-Gram similarity

Table 12 N-gram Similarity

N	N-Gram Similarity
N=2	50.72
N=3	30.44

4.5.2 Comparison of existing models

Table 13 Comparison with other models

Similarity Metric	Recall	Precision	F1	TP
Simple	0.67	0.67	0.67	43
Machine Learning	0.92	0.91	0.91	56

5. Conclusion

In conclusion, the widespread access to electronic resources has led to an increase in plagiarism among students. Plagiarism detection tools have been developed to address this issue, but recent studies have shown that they have difficulty identifying more sophisticated forms of plagiarism. In response to this challenge, the proposed approach in this research paper uses the Analytic Hierarchy Process technique to provide a multi-level plagiarism detection system that is both accurate and efficient. The system comprises of three levels of plagiarism detection, including the use of the google dorking technique, advanced text comparison algorithms, and full paper plagiarism check. This approach is essential for ensuring that submitted papers are original and have not been copied from any other source, and it can help teachers identify and reformulate plagiarized papers quickly, allowing them to focus their efforts on the remaining students.

REFERENCES

[1] Nazir, A., Mir, R. N., & Qureshi, S. (2021). Idea plagiarism detection with recurrent neural networks and vector space model. International Journal of Intelligent Computing and Cybernetics, 14(3), 321–332. https://doi.org/10.1108/ijicc-11-2020-0178

[2] JavadiMoghaddam, S., Roosta, F., & Noroozi, A. (2022). Weighted semantic plagiarism detection approach based on AHP decision model. Accountability in Research, 29(4), 203–223. https://doi.org/10.1080/08989621.2021.1911654

[3] Mansoor, M. N., & Al Tamimi, M. S. H. (n.d.). Plagiarism detection system in scientific publication using lstm networks. Iotpe.com. Retrieved February 2, 2023, from http://www.iotpe.com/IJTPE/IJTPE-2022/IJTPE-Issue53-Vol14-No4-Dec2022/3-IJTPE-Issue53-Vol14-No4-Dec2022-pp17-24.pdf

[4] Alvi, F., Stevenson, M., & Clough, P. (2021). Paraphrase type identification for plagiarism detection using contexts and word embeddings. International Journal of Educational Technology in Higher Education, 18(1), 1–25. https://doi.org/10.1186/s41239-021-00277-8

[5] Arabi, H., & Akbari, M. (2022). Improving plagiarism detection in text document using hybrid weighted similarity. Expert Systems with Applications, 207(118034), 118034. https://doi.org/10.1016/j.eswa.2022.118034

[6] Fokam, M. A., & Ajoodha, R. (2021, November). Influence of Contrastive Learning on Source Code Plagiarism Detection through Recursive Neural Networks. In 2021 3rd International Multidisciplinary Information Technology and Engineering Conference (IMITEC) (pp. 1-6). IEEE.

[7] El-Rashidy, M. A., Mohamed, R. G., El-Fishawy, N. A., & Shouman, M. A. (2022). Reliable plagiarism detection system based on deep learning approaches. Neural Computing and Applications, 34(21), 18837-18858.

[8] Ahuja, L., Gupta, V., & Kumar, R. (2020). A new hybrid technique for detection of plagiarism from text documents. Arabian Journal for Science and Engineering, 45, 9939-9952.

[9] Roostaee, M., Fakhrahmad, S. M., & Sadreddini, M. H. (2020). Cross-language text alignment: A proposed two-level matching scheme for plagiarism detection. Expert Systems with Applications, 160, 113718.

[10] AlSallal, M., Iqbal, R., Palade, V., Amin, S., & Chang, V. (2019). An integrated approach for intrinsic plagiarism detection. Future Generation Computer Systems, 96, 700-712.

[11] Comeau DC, Wei CH, Islamaj Doğan R, and Lu Z. PMC text mining subset in BioC: about 3 million full-text articles and growing, Bioinformatics, btz070, 2019.

[D2] “PubMed Standard Datasets.”

https://www.ncbi.nlm.nih.gov/pubmed/?term=pmc+cc+license%20%5Bfilter%5D

[D3] “PAN PC 2011”

https://zenodo.org/record/3250095

[D4] “Corpus for Plagiarised Short Answers”

https://www.researchgate.net/publication/220147549_Developing_a_corpus_of_plagiarised_short_answers

[D5] “PAN PC 2011”

https://zenodo.org/record/3250095

[D6] “C language code extracted from OJS”

https://github.com/NikolayIT/OpenJudgeSystem

[D7] “PAN 2013 & PAN 2014”

[D8] “PAN-PC-2011”

https://zenodo.org/record/3250095

[D9] “PAN-PC-11, PAN-PC-12 & SemEval-2017”

https://webis.de/data/pan-pc-11.html, https://alt.qcri.org/semeval2017/task4/

[D10] “CEN (Corpus of English Novel)”

https://www.kaggle.com/datasets/raynardj/classic-english-literature-corpus

[SD1] https://pubmed.ncbi.nlm.nih.gov/30675928/
[SD2] Chemicals and Drugs - Search Results - PubMed (nih.gov)
[SD3] MEMS technology for timing and frequency control - PubMed (nih.gov)
[SD4] Medical Equipment and Healthcare Technology: Health Vision 2050 - PubMed (nih.gov)
[SD5] 3D Printing in Pharmaceutical and Medical Applications - Recent Achievements and Challenges - PubMed (nih.gov)