

Development of a Scientific Journal Aggregator System Using Multi-API with a Text Mining TF-IDF and Cosine Similarity

Fakhrul Razi ^{1*}, Muhammad Fikry ^{2*}, Yesy Afrillia ^{3*}

* Informatics Engineering, Malikussaleh University

fakhrul.210170029@mhs.unimal.ac.id ¹, muh.fikry@unimal.ac.id ², yesy.afrillia@unimal.ac.id ³

Article Info

Article history:

Received 2026-06-29

Revised 2026-08-30

Accepted 2026-10-17

Keyword:

*Scientific Journal Aggregator,
Multi-API Integration,
Article Recommendation,
Text Mining.*

ABSTRACT

The rapid growth of scientific publications has made it increasingly difficult for researchers to efficiently identify relevant literature from multiple scholarly sources. Integrating heterogeneous scientific databases with an effective recommendation mechanism can improve both the relevance and efficiency of literature retrieval. This study aims to design and implement a scientific journal aggregator system that integrates multiple scholarly databases through Application Programming Interface (API) technology and incorporates a text mining-based recommendation mechanism to improve the relevance of search results. The proposed system integrates article metadata from OpenAlex, Semantic Scholar, and the Directory of Open Access Journals (DOAJ). Text preprocessing, including case folding, tokenization, and stopword removal, is applied before calculating Term Frequency–Inverse Document Frequency (TF-IDF) weights and Cosine Similarity to measure the relevance between user queries and scientific articles. The proposed system was evaluated using multiple search keywords, including *Artificial Intelligence*, *Machine Learning*, and *Technology*, to assess its recommendation performance across different research topics. Experimental results demonstrate that the system consistently achieved high recommendation performance, with accuracy ranging from 95.32% to 97.95%, precision ranging from 95.24% to 97.53%, recall consistently reaching 100.00%, and F1-score ranging from 97.56% to 98.75%. These findings indicate that integrating multiple APIs with TF-IDF and Cosine Similarity provides an effective approach for improving the relevance, coverage, and efficiency of scientific literature retrieval across different search topics.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. INTRODUCTION

The development of digital technology has driven significant changes in various fields, including the academic world. The use of web-based information systems has increased the ease of accessing, managing, and disseminating information quickly so as to support various academic and research activities [1][2]. On the other hand, the increasing number of scientific publications every year makes it easier for researchers to obtain references, but it also poses challenges in finding really relevant articles. Along with the increasing number of scientific publications, the article recommendation system is one of the most developed approaches to help researchers find suitable references faster

and more efficiently[3]. Based on research [4], the number of open access journals continues to increase in various scientific indexes so that the literature search process becomes increasingly complex. Currently, the article search process is still mostly carried out through various platforms such as Semantic Scholar, DOAJ, OpenAlex, and Scopus separately so that it takes longer and the search results obtained have not been integrated [5][6]. This condition causes the literature search process to be less efficient, especially in research activities such as literature studies and Systematic Literature Review (SLR). The recommendation system has been widely used to overcome information overload by filtering information and displaying content that best suits user needs based on data characteristics and user

preferences[7]. This research offers a solution in the form of developing a multi-API-based scientific journal aggregator website that applies text mining techniques using the Term Frequency–Inverse Document Frequency (TF-IDF) and Cosine Similarity methods to produce article recommendations based on the level of similarity of content. Several previous studies have developed a search system and recommendations for scientific articles with different approaches. Research by [2] shows that web-based information systems are able to increase the effectiveness of information management and delivery. Research [8] Explaining that the text mining technique can be used to extract information from unstructured text data to support the information search process more effectively. Furthermore, [9] successfully applied the TF-IDF and Cosine Similarity methods to the film-based recommendation system *content-based filtering* and generate relevant recommendations based on synopsis similarities. Research [10] also proves that the combination of the TF-IDF and Cosine Similarity methods is able to increase the relevance of search results in the tourist destination dataset. In addition, [11] develop a TF-IDF-based recommendation algorithm combined with user characteristics so that it can improve the quality of recommendations. In the field of scientific metadata management, [12] introducing OpenAlex as an open scientific index that provides publication metadata from various sources so that it can be used as the basis for the development of a search system and aggregation of scientific articles. [13]also developed a journal recommendation system *open access* based on article metadata. Nonetheless, the study focused on the recommendations of publication objectives (*journal recommendation*), rather than on a system of scientific article aggregators that integrate multiple metadata sources through an approach *multi-API*, so there is still an opportunity to develop a more comprehensive search and article recommendation system.

Based on these studies, there are still several limitations that have become *research gap*. Most studies have focused only on the development of recommendation or search systems by utilizing a single data source (*single API*), so that the scope of the articles obtained is still limited. Other studies have applied the TF-IDF and Cosine Similarity methods, but their implementation is more widely used in film recommendation systems, tourist destinations, and general documents, and has not been widely applied to scientific journal-based aggregator systems *multi-API* [7][8]. In addition, some journal provider platforms already have an artificial intelligence-based article recommendation feature, but the algorithm mechanism used is closed so it is difficult to evaluate or be further developed [4]. [3][14] explained that the scientific literature recommendation system has an important role in helping researchers find relevant publications amid the increasing number of scientific articles. However, most of the systems developed still utilize specific approaches and have not comprehensively integrated various scientific data sources. This condition shows that there is still

an opportunity to develop a scientific journal aggregator system that is able to integrate various data sources through an approach *multi-API* and implement an open and evaluable recommendation algorithm. Therefore, this study presents a novelty in the form of the development of a scientific journal aggregator website that integrates several data sources, namely Semantic Scholar, DOAJ, and OpenAlex, as well as applying the TF-IDF and Cosine Similarity methods as an article recommendation algorithm so that it is able to produce a wider, relevant, and measurable search.

The purpose of this research is to design and implement a multi-API-based scientific journal aggregator website that is able to integrate metadata from various scientific journal sources and provide article recommendations using the TF-IDF and Cosine Similarity methods. This approach is expected to be able to improve the quality of article recommendations through the content-based recommendation mechanism, which is one of the approaches that is widely applied to modern recommendation systems [7]. The developed system is then tested using the Black Box Testing method to ensure that all functions run according to the user's needs. This research is expected to improve the efficiency of the scientific literature search process, help researchers obtain more relevant references, and support research activities such as Systematic Literature Review (SLR) and research mapping more effectively. In addition, the results of this research are expected to be an alternative to the development of academic information systems that utilize multi-API integration and text mining techniques in improving the quality of scientific article search services.

II. METHOD

A. Research Stages

This study aims to design and implement a multi-API-based scientific journal aggregator website equipped with article recommendation features using the Term Frequency–Inverse Document Frequency (TF-IDF) and Cosine Similarity methods. The system was developed to integrate the metadata of scientific articles from several open sources, namely Semantic Scholar, OpenAlex, and the Directory of Open Access Journals (DOAJ). The development of a website-based system was chosen because it is able to provide information search services that are easily accessible, interactive, and accessible through various devices [1][15]. The following is a picture of the research work scheme:

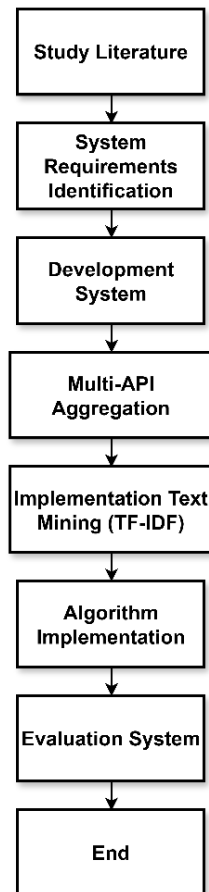


Figure 1. Research System

Based on Figure 1, the research began with a literature study to collect references regarding the journal aggregator system, Application Programming Interface (API) integration, text mining, as well as the TF-IDF and Cosine Similarity methods. The next stage is to identify system needs and design the system as the basis for application development. Next, multi-API aggregation was carried out to obtain scientific article metadata from Semantic Scholar, OpenAlex, and DOAJ. The integration of multiple APIs allows the system to obtain a wider data coverage than using a single data source and makes it easier to exchange information between systems through the REST API service [16]. The data that was successfully obtained was then processed using the text mining technique with TF-IDF weighting, followed by the calculation of the similarity level of documents using Cosine Similarity. After the system is completed, the system is tested using the Black Box Testing method to ensure that all functions run according to the needs of the user. The final stage is the evaluation of the system to assess the quality of implementation as well as the relevance of the resulting article recommendations.

B. System Implementation

The implementation of the system is carried out by building a website-based application using the Flask

framework as the backend and SQLite as the database. The use of SQLite was chosen because it is lightweight in size, easy to integrate with Python-based applications, and is able to store data efficiently without the need for a separate database server [17]. The system also utilizes the REST API to connect the application with multiple journal metadata sources so that the data retrieval process can be done automatically. Article metadata is obtained through the integration of three API services, namely Semantic Scholar, OpenAlex, and DOAJ. The information taken includes the title of the article, abstract, name of the author, year of publication, DOI, and source of the article. All metadata is stored in the SQLite database so that the search and recommendation process can be done more efficiently.

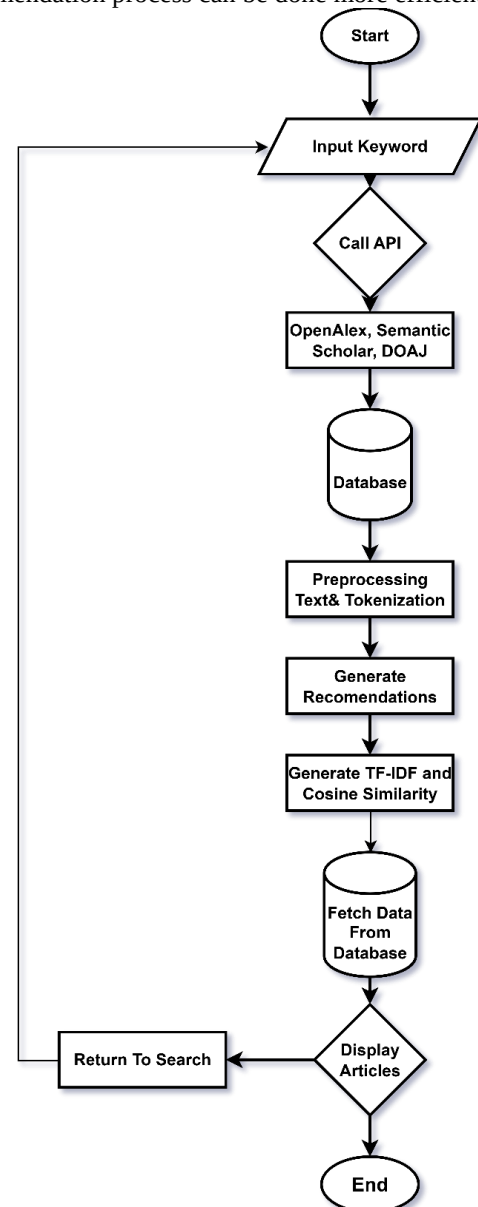


Figure 2. System Framework

Based on Figure 2, the user first enters the search keyword on the website page. The system then sends a request to multiple API services to obtain the article's metadata that matches that keyword. The metadata that is successfully obtained is then stored in the database before the text preprocessing process is carried out. After the data is processed, the system calculates the weight of each word using the TF-IDF method and measures the degree of similarity between articles using the Cosine Similarity method. Articles that have the highest similarity value will be displayed as recommendations to users so that the literature search process becomes more effective.

C. Implementation of Methods

Text mining is a systematic process to extract information, patterns, and knowledge from unstructured text data so that it can be converted into more meaningful information. This technique combines statistical methods, Natural Language Processing (NLP), and machine learning to process text into a representation that can be analyzed in various applications, such as document classification, information retrieval systems, and recommendation systems [8]. The application of text mining has also been widely used to improve the quality of information retrieval and text-based decision-making [18]. In this study, the Term Frequency-Inverse Document Frequency (TF-IDF) method was used to determine the level of importance of each word (*term*) in a document based on the frequency of its occurrence in the document and the rate of its occurrence in the entire document collection. The more often a word appears in one document but the less rarely it is found in other documents, the higher the TF-IDF weight will be. On the other hand, words that appear on almost all documents will have a lower weight because they are considered less able to distinguish the content of a document [10][19]. The results of the TF-IDF weighting are then used as a numerical representation of the document before the similarity measurement process is carried out.

D. Data Collection (Multi-API Integration)

The system integrates several external scholarly APIs including Semantic Scholar, OpenAlex, and DOAJ. Data retrieved from each source includes article title, abstract, author information, publication year, DOI, and source URL. The use of multiple APIs enables the system to gather scientific articles from heterogeneous sources, increasing data diversity compared to single-source retrieval systems. The broad scholarly coverage provided by OpenAlex and other academic databases, while [12].

E. Text Preprocessing

The collected textual data undergoes preprocessing to improve quality and consistency. This stage includes tokenization, case folding, stopword removal, and stemming. These steps aim to reduce noise and standardize the data for further analysis. The importance of preprocessing in text mining systems has been explained by [8].

The preprocessing stage plays an important role in transforming raw textual data obtained from journal metadata into a structured format suitable for text mining analysis. Case folding is used to convert all text into lowercase letters to maintain consistency, while tokenization separates sentences into individual terms that can be processed independently. Stopword removal eliminates common words that do not contribute significantly to semantic meaning, thereby reducing irrelevant information. Through these processes, the textual data becomes more optimized for TF-IDF weighting and similarity calculation, enabling the recommendation system to generate more accurate and relevant article recommendations, consistent with findings from [8][20].

F. Feature Extraction (TF-IDF)

After preprocessing, textual data is transformed into numerical representation using TF-IDF weighting. TF-IDF calculates the importance of a term within a document relative to the entire document collection.

The formula used is:

$$TF-IDF(t, d) = TF(t, d) \times IDF(t)$$

TF-IDF allows the system to identify significant terms that represent article content more accurately.

After obtaining the TF-IDF weights on each document, the next stage is to calculate the degree of similarity between the documents using the Cosine Similarity method. This method measures the similarity of two documents based on the angle formed from the TF-IDF vector representation. The greater the Cosine Similarity value or the closer it is to 1, the higher the level of similarity between the document and the search keyword. On the other hand, a value close to 0 indicates that the two documents have a low degree of similarity. This method is widely used in content-based recommendation systems because it is able to provide relevant recommendation results based on the similarity of the content of the document [9][21].

G. Similarity Measurement (Cosine Similarity)

Cosine Similarity is applied to measure similarity between user queries and article documents represented as TF-IDF vectors.

The formula used is:

$$\text{Similarity} = (A, B) = (A \cdot B) / (\|A\| \times \|B\|)$$

Similarity values range between 0 and 1, where values closer to 1 indicate higher relevance.

H. Evaluation Method

System evaluation is conducted using the Confusion Matrix to assess the performance of the proposed recommendation mechanism. The evaluation compares the recommendation results generated by the system with manually assigned ground truth labels to determine the correctness of article classification. Based on the confusion matrix, four evaluation metrics are calculated, namely Accuracy, Precision, Recall, and F1-Score, which provide a

comprehensive assessment of the recommendation performance.

The Confusion Matrix was selected because it is a widely adopted evaluation method for information retrieval and recommendation systems, enabling the identification of correctly and incorrectly classified articles through True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. In this study, manually assigned ground truth labels were used as the reference to determine whether each recommended article was relevant to the search keyword. The evaluation was performed using multiple search keywords, including *Artificial Intelligence*, *Machine Learning*, and *Technology*, to examine the consistency of the proposed recommendation mechanism across different research topics. Through this evaluation, the effectiveness of the TF-IDF and Cosine Similarity methods in retrieving relevant scientific literature can be quantitatively measured using Accuracy, Precision, Recall, and F1-Score.

III. RESULTS AND DISCUSSION

A. System Implementation Results

The scientific journal aggregator system was successfully implemented as a web-based application capable of integrating multiple academic APIs simultaneously. The system successfully retrieves scientific article metadata from Semantic Scholar, OpenAlex, and DOAJ based on user search queries. All API integration modules functioned correctly and returned article metadata including titles, abstracts, publication year, author information, and external links. Furthermore, the integration mechanism automatically consolidates data obtained from different sources into a unified data structure, allowing users to access search results through a single interface without manually searching each platform individually. This integration improves the efficiency of scientific literature retrieval while reducing the

time and effort required to collect relevant publications from multiple repositories.

In addition to data aggregation, the system incorporates a text mining-based recommendation module that processes article abstracts using preprocessing techniques, including case folding, punctuation removal, tokenization, stopword removal, and stemming. The processed text is then transformed into TF-IDF vectors, and Cosine Similarity is applied to measure the semantic relevance between the user's search query and the retrieved articles. Based on these similarity scores, the system ranks the articles in descending order of relevance, enabling users to identify the most appropriate references more efficiently. The implementation demonstrates that integrating multiple APIs with a text mining approach provides not only broader coverage of scientific publications but also more relevant and organized search results, thereby improving the overall effectiveness of the journal discovery process.

B. Text Processing Results

The text preprocessing module successfully standardized textual data collected from multiple academic API sources before further analysis. The preprocessing stage was applied to article metadata, particularly titles and abstracts retrieved based on the search keyword "*Artificial Intelligence*."

The preprocessing pipeline consisted of four main stages: case folding, tokenization, stopword removal, and stemming. For example, an article title such as: "*Artificial Intelligence Applications in Medical Diagnosis Systems*" After preprocessing becomes: "*artificial intelligence application medical diagnosis system*".

Case folding converted all text into lowercase letters to eliminate inconsistencies caused by letter capitalization. Tokenization separated the text into individual terms for easier processing. Stopword removal eliminated words with low semantic importance, while stemming reduced words into their root forms to minimize term variation.

TABLE I
RETRIEVED ARTICLE DATASET

No	Code	Document Contents
1	D1	Conception of Legal Subjects and Criminal Responsibility Artificial Intelligence
2	D2	Challenges of Artificial Intelligence Technology in Student Learning Activities
3	D3	The role of Artificial Intelligence in achieving the Sustainable Development Goals
...	...	...
376	D376	Artificial Intelligence-enhanced electrocardiography in cardiovascular disease management
377	D377	Rethinking evidence-informed policies and practices in the era of generative Artificial Intelligence
378	D378	Design of an Educational Space-Based Photography Laboratory at Al Bunyan School Bogor

C. Case Folding Results

The case folding stage is a preprocessing step used to standardize the textual data by converting all characters in the document into lowercase letters. This process aims to eliminate inconsistencies caused by differences in letter capitalization, thereby reducing data redundancy and

improving the effectiveness of subsequent text processing stages.

TABEL II CASE FOLDING RESULTS

No	Code	Document Contents
1	D1	conception of legal subjects and criminal responsibility artificial intelligence
2	D2	challenges of artificial intelligence technology in student learning activities
3	D3	the role of artificial intelligence in achieving the sustainable development goals
...	...	...
376	D376	artificial intelligence-enhanced electrocardiography in cardiovascular disease management
377	D377	rethinking evidence-informed policies and practices in the era of generative artificial intelligence
378	D378	design of an educational space-based photography laboratory at ai bunyan school bogor

The dataset presented in Table II consists of scientific article titles that have undergone the *case folding* process, where all characters are converted into lowercase format to ensure text consistency during preprocessing. The collected documents were retrieved from multiple API sources and contain various research topics, with the majority focusing on *Artificial Intelligence*. Each document is represented by a unique code (D1–D378) to simplify the identification and analysis process.

From the dataset, it can be observed that several documents contain frequently occurring terms such as *machine* and *learning*, indicating strong thematic relevance to the search keyword used in this study. In contrast, documents such as D378 contain terms unrelated to the primary topic, suggesting not relevant levels during the recommendation process. These documents are subsequently processed through tokenization, stopword removal, stemming, and TF-IDF weighting to determine the relevance score and generate article recommendations based on text similarity.

D. Tokenization Results

The tokenization stage is performed to separate each document into individual words or terms that can be processed by the system. This process aims to break down textual data into smaller units, allowing the system to identify and analyze the occurrence of specific terms related to the search keywords. Tokenization plays an important role in text preprocessing because unstructured textual information cannot be directly processed for analysis and similarity measurement. Therefore, converting sentences into individual terms simplifies the representation of textual data and prepares it for subsequent processing stages.

In this study, the tokenization process is applied to scientific article titles that have previously undergone the case folding stage. During this process, each sentence is divided into separate terms based on spaces and delimiters, resulting in a collection of individual words that represent the content of each document.

TABEL III TOKENIZATION RESULTS

No	Code	Document Contents
1	D1	conception, subject, law, liability, criminal, artificial, intelligence
2	D2	challenges, technology, artificial, intelligence, activities, learning, student
3	D3	role, artificial, intelligence, development
...	...	...
376	D376	electrocardiography, artificial, intelligence, management, disease, cardiovascular
377	D377	policy, practice, artificial, intelligence, generative
378	D378	design, laboratory, photography, educational, space, school

Based on Table III, each document has been successfully separated into individual tokens. The resulting terms represent the main keywords contained in each document and serve as the initial input for subsequent preprocessing stages.

E. Stopword Removing Results

The *stopword removal* stage is used to eliminate common words that do not contribute significantly to the meaning of the document. Removing these terms reduces irrelevant information and improves the efficiency of the text mining process.

TABEL IV STOPWORD REMOVING

No	Code	Document Contents
1	D1	conception, subject, law, liability, artificial, intelligence
2	D2	technology, artificial, intelligence, learning, student
3	D3	artificial, intelligence, development
...	...	...
376	D376	electrocardiography, artificial, intelligence, management, cardiovascular
377	D377	artificial, intelligence, generative
378	D378	laboratory, photography, educational, school

Based on Table IV, words with low semantic importance have been removed from the documents. As a result, only meaningful terms remain, allowing the system to focus on more relevant information during similarity calculations.

Terms frequently appearing within a document but rarely appearing across other documents receive higher TF-IDF weights, indicating stronger importance in representing document content. TF-IDF successfully transformed unstructured textual metadata into measurable numerical vectors suitable for similarity analysis.

F. TF-IDF Calculation Results

TABLE V
TF-IDF CALCULATION RESULTS

No	Term	IDF	D1	D2	D3	D376	D377	D378
1	artificial	0.079	0.079	0.079	0.079	0.079	0.079	0
2	intelligence	0.079	0.079	0.079	0.079	0.079	0.079	0
3	conception	0.778	0.778	0	0	0	0	0
4	law	0.778	0.778	0	0	0	0	0
5	liability	0.778	0.778	0	0	0	0	0
6	technology	0.778	0	0.778	0	0	0	0
7	learning	0.778	0	0.778	0	0	0	0
8	student	0.778	0	0.778	0	0	0	0
9	development	0.778	0	0	0.778	0	0	0
10	electrocardiography	0.778	0	0	0	0.778	0	0
11	management	0.778	0	0	0	0.778	0	0
12	cardiovascular	0.778	0	0	0	0.778	0	0
13	generative	0.778	0	0	0	0	0.778	0
14	laboratory	0.778	0	0	0	0	0	0.778
15	photography	0.778	0	0	0	0	0	0.778
16	educational	0.778	0	0	0	0	0	0.778
17	school	0.778	0	0	0	0	0	0.778

G. Cosine Similarity Results

Cosine Similarity is used to measure the degree of similarity between the user query and document vectors generated from TF-IDF weighting. This method calculates the cosine angle between two vectors, namely the query vector and document vectors. Similarity values range from 0 to 1, where values closer to 1 indicate a stronger relationship between the query and the document, while values closer to 0 indicate lower relevance.

H. Multi-API Integration Analysis

The proposed journal aggregator integrates metadata from OpenAlex, Semantic Scholar, and the Directory of Open Access Journals (DOAJ) through Application Programming Interface (API) technology. During the integration process, several challenges were encountered due to differences in metadata structures provided by each API. For example, author names were represented using different formats, publication years were stored under different field names, and some APIs did not consistently provide abstracts, keywords, or Digital Object Identifiers (DOIs). In addition, differences in response formats and the availability of metadata required additional processing before the retrieved information could be utilized by the recommendation system.

To address these inconsistencies, the retrieved metadata were normalized into a unified data structure consisting of article title, abstract, author(s), publication year, DOI, source, and article URL. Each API response was processed independently and then merged into a single dataset to ensure that all retrieved articles shared the same metadata format regardless of their original source. This normalization process enabled subsequent text preprocessing and similarity analysis to be performed consistently across all integrated repositories.

Another challenge involved duplicate articles retrieved from multiple APIs because the same publication may be indexed by more than one scholarly database. Duplicate detection was performed by comparing the Digital Object Identifier (DOI) whenever available. If a DOI was unavailable, duplicate records were identified using title similarity. After duplicate removal and metadata normalization, the processed dataset was used for text preprocessing and recommendation generation. This harmonization process ensured that articles obtained from different scholarly repositories could be analyzed consistently using the proposed TF-IDF and Cosine

Similarity approach, thereby improving the reliability and consistency of the recommendation results.

Each integrated API also has several limitations that affect the data retrieval process. For instance, the availability and completeness of metadata vary across OpenAlex, Semantic Scholar, and DOAJ, with some articles lacking abstracts, keywords, or DOI information. In addition, differences in response time and API request limitations may influence the number of retrieved articles during a search session.

To ensure consistent data retrieval, the proposed system sends API requests sequentially to each scholarly repository based on the user query. The retrieved metadata from each API are then normalized and merged into a unified dataset before duplicate detection, text preprocessing, and similarity analysis are performed. This synchronization mechanism enables the system to process heterogeneous scholarly metadata consistently while providing integrated search results from multiple sources.

I. Analysis of Recommendation Results

The evaluation results presented in Table VI demonstrate that the proposed recommendation mechanism consistently achieved high performance across different search keywords. Nevertheless, a number of retrieved articles were still classified as non-relevant during the manual assessment process. This section discusses the factors contributing to these results and the influence of metadata quality on the recommendation performance.

The occurrence of false positive (FP) recommendations indicates that several retrieved articles were predicted as relevant by the system but were manually labeled as non-relevant in the ground truth. This situation mainly occurred because the TF-IDF and Cosine Similarity methods measure textual similarity based on the occurrence and distribution of terms rather than the semantic meaning of the documents. Consequently, articles containing similar keywords in their titles, abstracts, or keywords may obtain relatively high similarity scores even though their primary research focus differs from the user's intended query. For example, articles containing the keyword *Artificial Intelligence* may discuss its application in unrelated domains, causing them to be recommended despite being considered less relevant during manual evaluation.

The number of non-relevant articles also reflects the limitations of keyword-based text representation. Although TF-IDF effectively assigns higher weights to discriminative terms, it cannot fully capture contextual relationships between words. As a result, documents discussing different research topics but sharing similar terminology may still be identified as relevant. This phenomenon explains why a small number of articles were categorized as false positives, even though the overall evaluation achieved high Accuracy, Precision, Recall, and F1-Score values.

Another factor influencing the recommendation results is the quality and completeness of metadata obtained from the integrated APIs. OpenAlex, Semantic Scholar, and DOAJ

provide different metadata structures and levels of completeness. Some articles contain complete titles, abstracts, keywords, and Digital Object Identifiers (DOIs), while others provide only limited metadata. Since the recommendation mechanism relies on textual information extracted from article metadata, incomplete abstracts or missing keywords may reduce the effectiveness of TF-IDF weighting and similarity calculation. To minimize this issue, the proposed system performs metadata normalization before text preprocessing and similarity analysis, allowing documents retrieved from different repositories to be processed using a consistent representation.

Despite these limitations, the experimental results indicate that the proposed recommendation mechanism performs consistently across different search keywords. The evaluation achieved high Accuracy, Precision, Recall, and F1-Score values, demonstrating that the integration of multiple scholarly databases with TF-IDF and Cosine Similarity can effectively retrieve and rank relevant scientific literature. The relatively small number of false-positive recommendations compared with the total number of retrieved articles also indicates that the proposed system provides reliable recommendations while maintaining broad literature coverage from multiple scholarly repositories.

J. Evaluation System

The performance of the proposed scientific journal aggregator system was evaluated using four widely adopted classification metrics, namely Accuracy, Precision, Recall, and F1-Score. These metrics were calculated based on the confusion matrix obtained by comparing the system's recommendation results with manually assigned ground truth labels. Manual labeling was performed by assessing whether each recommended article was relevant to the search keyword Artificial Intelligence. The evaluation produced 360 True Positives (TP), 18 False Positives (FP), 7 True Negatives (TN), and no False Negatives (FN). Accuracy measures the proportion of correctly classified articles among all evaluated

articles. The proposed system achieved an accuracy of 95.32%, indicating that most scientific articles were classified correctly according to their relevance. This result demonstrates that the combination of TF-IDF and Cosine Similarity provides reliable performance for identifying relevant scientific publications retrieved from multiple scholarly repositories.

Precision evaluates the proportion of recommended articles that are actually relevant. The obtained precision value of 95.24% indicates that the majority of articles recommended by the system were relevant to the user's search query. Nevertheless, the presence of 18 false-positive articles shows that several publications with relatively low contextual relevance were still recommended. This condition is expected because the similarity threshold of 0.01 was intentionally selected to maximize the retrieval of potentially relevant literature.

Recall measures the capability of the system to retrieve all relevant articles available in the evaluation dataset. The proposed system achieved a recall of 100%, indicating that every article manually labeled as relevant was successfully retrieved by the recommendation mechanism. Since no false-negative cases were observed during evaluation, the recommendation system did not fail to identify any relevant scientific publication. This result demonstrates that the selected recommendation strategy effectively maximizes literature coverage, which is particularly important in scientific literature retrieval where missing relevant references should be minimized.

The F1-Score represents the harmonic mean of Precision and Recall, providing a balanced evaluation of recommendation performance. The obtained F1-Score of **97.56%** indicates that the proposed recommendation mechanism successfully maintains a balance between retrieving relevant articles and minimizing irrelevant recommendations. The high F1-Score confirms that the integration of TF-IDF weighting and Cosine Similarity effectively ranks scientific articles according to their textual relevance while maintaining consistent recommendation performance across the evaluation dataset.

Overall, the evaluation results demonstrate that the proposed scientific journal aggregator provides highly effective recommendation performance. Although several irrelevant articles were still retrieved due to the relatively low similarity threshold, the system successfully identified all relevant scientific publications while maintaining high classification accuracy. These findings indicate that integrating multiple scholarly repositories with a text mining-based recommendation mechanism can significantly improve the effectiveness and efficiency of scientific literature retrieval.

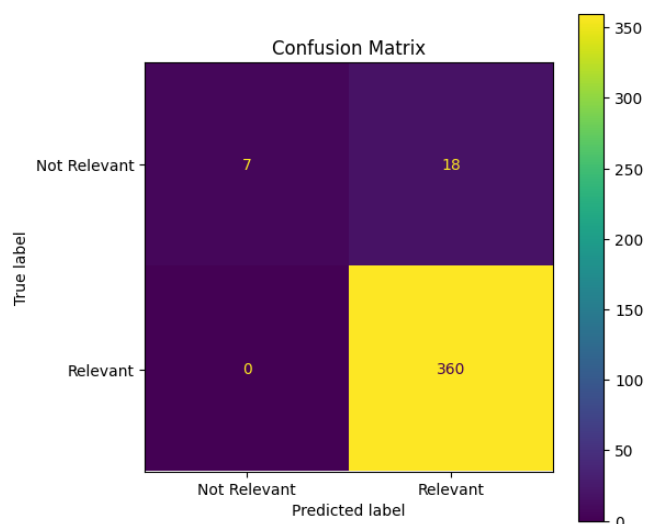


Figure 3. Confusion Matrix

Figure 3 presents the confusion matrix obtained during system evaluation. The proposed recommendation system correctly identified 360 relevant articles as relevant (True

Positive) and correctly classified 7 non-relevant articles as non-relevant (True Negative). In addition, the system produced 18 false-positive recommendations, indicating that several non-relevant articles were incorrectly recommended. However, no false-negative cases were observed, meaning that every relevant article in the evaluation dataset was successfully retrieved. This outcome explains why the system achieved a recall of 100%, while maintaining high accuracy, precision, and F1-Score. The confusion matrix demonstrates that the proposed recommendation mechanism prioritizes retrieving all relevant scientific literature, even though a small number of additional irrelevant articles may also be recommended.

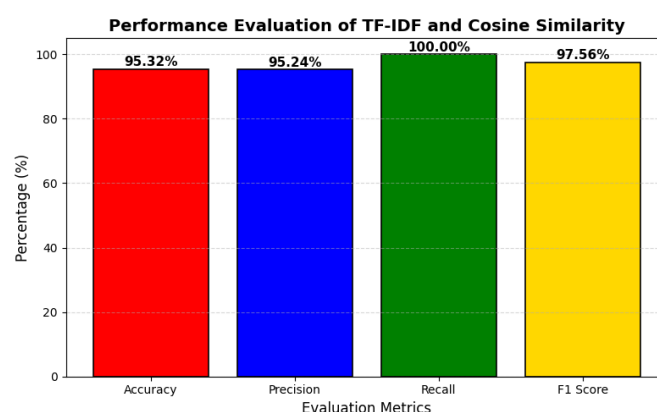


Figure 4. Evaluation Metrics

Figure 4 illustrates the evaluation results of the proposed recommendation system using Accuracy, Precision, Recall, and F1-Score. The graph shows that the system achieved an accuracy of 95.32%, indicating that most recommended articles were classified correctly according to their relevance. The precision value of 95.24% demonstrates that the majority of recommended articles were relevant, although a small number of irrelevant articles were still retrieved. Furthermore, the system achieved a recall of 100%, indicating that all manually identified relevant articles were successfully retrieved without omission. Consequently, the F1-Score reached 97.56%, reflecting a well-balanced performance between recommendation precision and completeness. These results confirm that the proposed TF-IDF and Cosine Similarity-based recommendation mechanism is capable of providing highly relevant scientific article recommendations while maintaining comprehensive literature retrieval.

K. System Design

The system design illustrates the overall architecture and workflow of the proposed scientific journal aggregator system. The system integrates multiple API sources to collect journal data automatically. Retrieved data undergo text preprocessing stages including case folding, tokenization, stopword removal, and stemming to improve text quality and consistency. The processed data are then weighted using the

TF-IDF method and analyzed using Cosine Similarity to determine the relevance between user search queries and available articles. Finally, the system generates ranked recommendations and displays the most relevant scientific articles to users.

1) *Use Case Diagram*: The Use Case Diagram illustrates the interaction between users and the proposed scientific journal aggregator system. The user acts as the main actor who interacts with the system by entering search keywords, accessing journal information, viewing article details, and obtaining article recommendations. The system processes user requests by retrieving data from multiple API sources and generating relevant recommendation results based on text mining techniques.

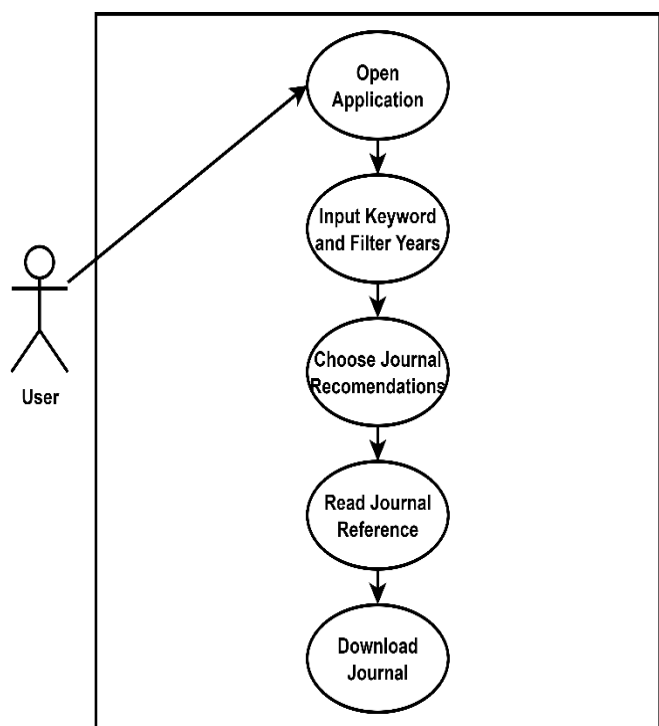


Figure 5. Use Case Diagram

2) *Activity Diagram*: The Activity Diagram describes the workflow of the scientific journal aggregation process. The process begins when the user enters a search keyword into the system. The system then retrieves article data from multiple API sources and performs text preprocessing stages including *case folding*, *tokenization*, *stopword removal*, and *stemming*. After preprocessing, TF-IDF weighting and Cosine Similarity calculations are applied to determine document relevance. Finally, the system displays ranked scientific article recommendations to the user.

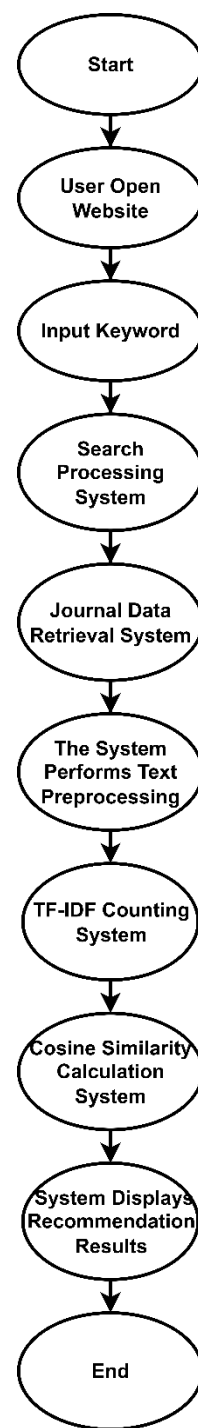


Figure 6. Activity Diagram

3) *Sequence Diagram*: The Sequence Diagram illustrates the communication flow and interaction among system components during the article retrieval and recommendation process. The process begins when the user enters a search query through the system interface. The system then sends requests to multiple API sources, such as OpenAlex, Semantic Scholar, and DOAJ, to retrieve relevant journal data

including titles, authors, publication years, abstracts, and other metadata information.

After the data retrieval process, the system performs text preprocessing stages, including *case folding*, *tokenization*, *stopword removal*, and *stemming*, to improve data quality and prepare the textual information for analysis. The processed data are then weighted using TF-IDF and analyzed using Cosine Similarity to determine the relevance between user queries and article documents. Finally, the system ranks the articles based on similarity scores and displays the most relevant recommendation results to the user.

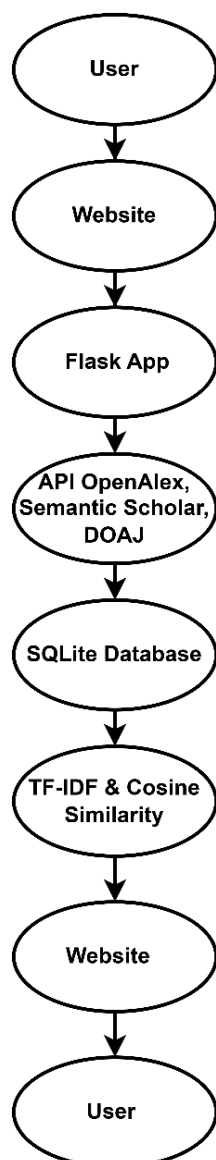


Figure 7. Sequence Diagram

4) **Component Diagram:** The Component Diagram describes the structural architecture of the proposed system and the relationship among its components. The system consists of several modules including the user interface, API

integration module, preprocessing module, TF-IDF calculation module, Cosine Similarity module, and database or storage component. Each component performs specific tasks and interacts with other components to support the overall functionality of the scientific journal aggregation system.

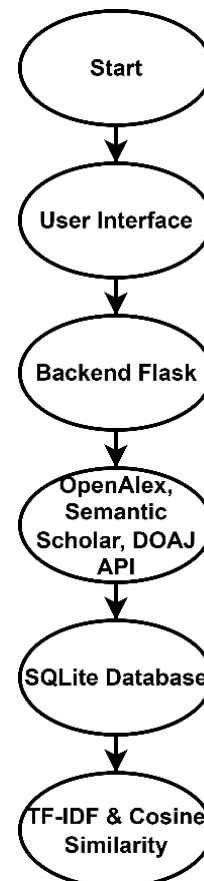


Figure 8. Component Diagram

L. System Performance

The performance of the proposed scientific journal aggregator system was evaluated based on data retrieval capability, preprocessing effectiveness, recommendation relevance, and system response. The developed system successfully integrated multiple API sources and retrieved scientific article metadata from OpenAlex, Semantic Scholar, and DOAJ. The retrieved data were processed through text preprocessing stages, including case folding, tokenization, stopwords removal, and stemming.

The implementation of TF-IDF and Cosine Similarity methods enabled the system to generate relevant article recommendations according to user search queries. Experimental results showed that documents containing terms similar to the input query produced higher similarity scores and were ranked at the top recommendation positions. The ranking mechanism effectively filtered irrelevant documents and improved the quality of search results.

In addition, system functionality testing indicated that all major features operated successfully, including journal searching, API data retrieval, preprocessing, similarity calculation, and recommendation generation. The system was able to provide structured and relevant scientific articles while improving the efficiency of literature discovery.

TABLE VI KEYWORD TESTING

No	Search Keyword	Retrieved Articles	Relevant	Not Relevant	Accuracy(%)	Precision(%)	Recall(%)	F1 Score(%)
1	Artificial Intelligence	385	360	25	95.32	95.24	100	97.56
2	Machine Learning	416	400	16	96.88	96.85	100	98.40
3	Technology	390	316	74	97.95	97.53	100	98.75

Table VI presents the performance evaluation of the proposed scientific journal aggregator using three different search keywords, namely *Artificial Intelligence*, *Machine Learning*, and *Technology*. The evaluation was conducted using manually assigned ground truth labels and measured using Accuracy, Precision, Recall, and F1-Score.

For the keyword *Artificial Intelligence*, the system retrieved 385 scientific articles, consisting of 360 relevant and 25 non-relevant articles. The proposed recommendation mechanism achieved an Accuracy of 95.32%, Precision of 95.24%, Recall of 100.00%, and an F1-Score of 97.56%. The recall value of 100% indicates that all manually identified relevant articles were successfully retrieved, while the slightly lower precision shows that a small number of non-relevant articles were also recommended.

When using the keyword *Machine Learning*, the system retrieved 416 articles, of which 400 were manually identified as relevant and 16 as non-relevant. The evaluation results showed an Accuracy of 96.88%, Precision of 96.85%, Recall of 100.00%, and an F1-Score of 98.40%. Compared with the previous keyword, the system demonstrated higher classification performance, indicating that the proposed recommendation mechanism effectively handled a larger collection of scientific publications while maintaining high recommendation quality.

For the keyword *Technology*, the system retrieved 390 scientific articles, including 316 relevant and 74 non-relevant articles. Despite the broader scope and more diverse nature of this keyword, the proposed system achieved an Accuracy of 97.95%, Precision of 97.53%, Recall of 100.00%, and an F1-Score of 98.75%. These results indicate that the recommendation mechanism remained highly effective even when processing a more general search topic that covers multiple research domains.

Overall, the evaluation results demonstrate that the proposed journal aggregator consistently achieved high performance across different search keywords. The recall value remained 100% for all evaluation scenarios, indicating that the recommendation mechanism successfully retrieved all articles labeled as relevant during manual assessment.

Furthermore, the consistently high Accuracy, Precision, and F1-Score values confirm that the integration of multiple scholarly databases with TF-IDF and Cosine Similarity provides reliable and accurate scientific article recommendations across different research topics.

IV. CONCLUSION

This study successfully designed and implemented a web-based scientific journal aggregator that integrates multiple scholarly databases through the Application Programming Interface (API), including OpenAlex, Semantic Scholar, and the Directory of Open Access Journals (DOAJ). The proposed system enables users to retrieve scientific articles from multiple repositories through a single search interface, thereby improving the efficiency and comprehensiveness of scientific literature retrieval. To enhance recommendation relevance, the system applies text preprocessing techniques consisting of case folding, tokenization, and stopword removal, followed by TF-IDF weighting and Cosine Similarity to measure the similarity between user queries and scientific articles.

The experimental results demonstrate that the proposed recommendation mechanism effectively identifies and ranks relevant scientific literature. Based on the evaluation using manually assigned ground truth labels, the system achieved an accuracy of 95.32%, a precision of 95.24%, a recall of 100.00%, and an F1-score of 97.56%. These results indicate that the recommendation system successfully retrieved all relevant scientific articles while maintaining a high level of recommendation precision. Although several non-relevant articles were also recommended, the overall evaluation confirms that the proposed approach provides reliable and effective recommendation performance for scientific literature retrieval.

Furthermore, integrating multiple scholarly repositories significantly expands the coverage of retrieved publications compared with conventional single-source search systems. The combination of multi-API integration with TF-IDF and Cosine Similarity provides an effective solution for improving both the relevance and efficiency of scientific

article retrieval. Therefore, the developed system can serve as a valuable tool for researchers, students, and academics in obtaining relevant scientific references more quickly and comprehensively.

Although the proposed recommendation mechanism demonstrated promising performance, this study evaluated the recommendation model independently without comparing it with other retrieval approaches, such as BM25 or embedding-based methods. Therefore, the comparative effectiveness of the proposed approach against more advanced retrieval models has not yet been investigated and constitutes a limitation of the present study.

Future work may focus on comparing the proposed recommendation mechanism with alternative retrieval approaches, such as BM25, Sentence-BERT, MPNet, or other transformer-based embedding models, while integrating additional scholarly databases to further increase literature coverage and recommendation accuracy. Future studies may also investigate adaptive similarity thresholds and larger evaluation datasets to further improve the effectiveness and generalization capability of the proposed scientific journal recommendation system.

REFERENCES

- [1] J. Teknologi, F. Aryani, and A. Voutama, "Design of a Motorcycle Rental Management Information System in the Bekasi Area Using the SDLC Method," vol. 15, pp. 15–28, 2025, doi: 10.34010/jati.v15i1.12636.
- [2] Y. Afrillia and S. Ramadani, "Book Layout Search Information System in Libraries Using Binary Search Method," *J. Inform. Kaputama*, vol. 6, no. 1, pp. 31–35, 2022, doi: 10.59697/jik.v6i1.132.
- [3] C. K. Kreutz and R. Schenkel, *Scientific paper recommendation systems : a literature review of recent publications*, vol. 23, no. 4. Springer Berlin Heidelberg, 2022. doi: 10.1007/s00799-022-00339-w.
- [4] A. Maddi, M. Maisonobe, and C. Boukacem-Zeghmouri, "Geographical and disciplinary coverage of open access journals: OpenAlex, Scopus, and WoS," *PLoS One*, vol. 20, no. 4 April, pp. 1–21, 2025, doi: 10.1371/journal.pone.0320347.
- [5] J. Lin, J. Song, Z. Zhou, Y. Chen, and X. Shi, "Automated scholarly paper review: Concepts, technologies, and challenges," *Inf. Fusion*, vol. 98, p. 101830, 2023, doi: <https://doi.org/10.1016/j.inffus.2023.101830>.
- [6] R. Ofori, B. Magaly, A. Martins, and N. Wiratunga, *language processing , machine learning , and deep learning .*, vol. 57, no. 8. Springer Netherlands, 2024. doi: 10.1007/s10462-024-10844-w.
- [7] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *J. Big Data*, 2022, doi: 10.1186/s40537-022-00592-5.
- [8] S. Kumar, A. K. Kar, and P. V. Ilavarasan, "Applications of text mining in services management: A systematic literature review," *Int. J. Inf. Manag. Data Insights*, vol. 1, no. 1, p. 100008, 2021, doi: 10.1016/j.jjime.2021.100008.
- [9] A. H. J. P. Juni Permana and Agung Toto Wibowo, "Movie Recommendation System Based on Synopsis Using Content-Based Filtering with TF-IDF and Cosine Similarity," *Int. J. Inf. Commun. Technol.*, vol. 9, no. 2, pp. 1–14, 2023, doi: 10.21108/ijoict.v9i2.747.
- [10] R. Al Rasyid and D. H. U. Ningsih, "Application of TF-IDF and Cosine Similarity Algorithms for Search Queries in Tourist Destination Datasets," *J. JTik (Journal of Technology, Inf. and Communication)*, vol. 8, no. 1, pp. 170–178, 2024, doi: 10.35870/jtik.v8i1.1416.
- [11] J. Ni, Y. Cai, G. Tang, and Y. Xie, "Collaborative filtering recommendation algorithm based on TF-IDF and user characteristics," *Appl. Sci.*, vol. 11, no. 20, 2021, doi: 10.3390/app11209554.
- [12] J. Priem, H. Piwowar, and R. Orr, "OpenAlex: A fully-open index of scholarly works, authors, venues, institutions, and concepts," vol. 27330, pp. 2018–2022, 2022, [Online]. Available: <http://arxiv.org/abs/2205.01833>
- [13] E. Entrup *et al.*, "Comparing different search methods for the open access journal recommendation tool B ! SON," *Int. J. Digit. Libr.*, vol. 25, no. 3, pp. 505–516, 2024, doi: 10.1007/s00799-023-00372-3.
- [14] Z. Zhang *et al.*, "Scholarly recommendation systems: a literature survey," *Knowl. Inf. Syst.*, vol. 65, no. 11, pp. 4433–4478, 2023, doi: 10.1007/s10115-023-01901-x.
- [15] S. A. P. Agustin Shafa; Budi Santoso, Gatot, "Web-Based File Upload And Download Application Using Python," *Metr. Ser. Hum. and Science*, no. Vol. 5 No. 2 (2024): Oktober 2024, pp. 90–95, 2024, [Online]. Available: <https://publikasi.kocenin.com/index.php/huma/article/view/594/487>
- [16] N. A. Prasetyo and Y. Saintika, "Integration between Moodle and Academic Information System using Restful API for Online Learning," *J. Ilm. Tek. Electro Compute. and Inform.*, vol. 7, no. 2, p. 358, 2021, doi: 10.26555/jiteki.v7i2.21816.
- [17] E. Daraghmi, Z. Qaroush, and M. Hamdi, "applied sciences Forensic Operations for Recognizing SQLite Content (FORC): An Automated Forensic Tool for Efficient SQLite Evidence Extraction on Android Devices," 2023.
- [18] Y. A. P. G. S, G. L. Ginting, and M. Panjaitan, "Optimization of New Student Admissions with the Application of Text Mining and TF-IDF Algoritma," vol. 2, no. 3, pp. 109–115, 2023, doi: 10.47065/comforch.v2i3.941.
- [19] G. Yunanda, D. Nurjanah, and S. Meliana, "Recommendation System from Microsoft News Data using TF-IDF and Cosine Similarity Methods," *Build. Informatics, Technol. Sci.*, vol. 4, no. 1, pp. 277–284, 2022, doi: 10.47065/bits.v4i1.1670.
- [20] G. H. Setiawan and I. M. B. Adnyana, "Improving Helpdesk Chatbot Performance with Term Frequency-Inverse Document Frequency (TF-IDF) and Cosine Similarity Models," vol. 7, no. 2, pp. 252–257, 2023.
- [21] R. Huang, "Improved content recommendation algorithm integrating semantic information," *J. Big Data*, vol. 10, no. 1, 2023, doi: 10.1186/s40537-023-00776-7.