IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 27, Issue 1, Ser. 4 (Jan. – Feb. 2025), PP 04-10
www.iosrjournals.org

Research Abstracts Similarity Implementation By Using TF-IDF Algorithm

Yusra Nuri[1], Edip Senyurek[2]
[1,2](Department Of Computer Engineering, Vistula University, Poland)

Abstract:
Background: *It can be time-consuming to go through publications to extract information that is specific to the research topic. While obtaining results based on keywords or titles is simple, this study seeks to give more relevant results by comparing the author's summary of a specific topic with a collection of existing research papers. This study describes the effects of the application for a TF-IDF (Term Frequency – Inverse Document Frequency) algorithm using C# and MS-SQL. The focus is on how this algorithm can provide more specific and relevant research results.*
Methodology: *In this study, 49 randomly selected research papers were preprocessed, which involved cleaning the abstract texts by removing punctuation and common words. Once the preprocessing stage was completed, the TF-IDF algorithm and other statistical measures were calculated to rank the 49 articles based on their relevance to the target abstract.*
Results: *Since this article includes 29 references, we specifically examined the top 29 ranked articles to see how many of our references were included in the list. Out of these 29 references, 26 were ranked within the top 29, while three did not appear in this list. This means the algorithm accurately identified 26 references as highly relevant, resulting in a similarity rate of 89.6%.*
Conclusion*: The results demonstrated a high similarity rate, which means 89.6% of the referenced articles were correctly identified as the most relevant publications, highlighting the effectiveness of this approach.*
Key Word: *TF-IDF, Filtering, Information Retrieval, Z-score, Similarity, Normalization.*

Date of Submission: 08-02-2025 Date of Acceptance: 18-02-2025

I. Introduction

The abundance of information on the internet has made it challenging to identify relevant works effectively, thereby making text-mining techniques more essential than ever. To effectively search and categorize from a vast corpus of data, information retrieval systems are necessary. Amongst this abundant content, effective text-mining methods are critical for deriving meaningful insights from large datasets.

TF-IDF addresses this by quantifying the importance of terms within a document and across a group of documents, enabling precise retrieval of relevant abstracts. TF-IDF calculates how relevant a word is by multiplying two metrics: Term Frequency (*TF*) and Inverse Document Frequency (*IDF*). Term frequency computes how common a term is in a particular document, while inverse document frequency computes how common a word is across the entire document.

TF-IDF is chosen because of its wide variety of applications and easy-to-implement nature. It also gives better results for content-based filtering.

This study focuses on exploring the practical applications of TF-IDF in information retrieval, specifically in listing the most similar publications according to their abstracts. By focusing on abstracts, which are short summaries of research papers, we aim to simplify research by helping researchers find more relevant and specific content.

II. Literature Review

As technology advances rapidly, it brings new challenges that demand advanced solutions. Researchers use algorithms like TF-IDF to address issues in fields such as recommendation systems, text analysis, and detecting harmful activities. This section has two parts: the first part looks at how TF-IDF is used in text analysis and recommendations, while the second one focuses on how it helps detect spam, fake users, and other security issues.

Recommendation Systems

TF-IDF is a widely used technique for processing text data and is often used with other tools to improve various systems. This section explores research on the application of TF-IDF for recommending products, analyzing sentiments in reviews, and organizing topics within large datasets. It also highlights how this technique improves experiences in areas such as online shopping and education.

Computational techniques were used to tackle the big challenge of hate and offensive content directed towards people or groups on social media [1]. Similarly, Reddy et al. [2] analyzed ways to find fake users and spammers on Twitter. Furthermore, a technique that groups different techniques to detect spam on Twitter is presented, based on how well they can identify fake content, URL spam, trending content spam, and fake users.

Content-based recommendation systems that use TF-IDF and cosine similarity significantly improved personalized recommendations in online purchasing [3]. This was achieved by providing high precision and recall by calculating the cosine similarity after the conversion of the pre-processed data into a TF-IDF grid. A customer-tailored recommendation system for network teaching that uses a combination of filtering methods and deep learning in Natural Language Processing (*NLP*) [4]. This system is more accurate and reliable than others of a similar kind because it helps create better curriculum suggestions based on the data.

A study by Zhou et al [5] introduced an optimized method to group news topics on Spark improving how large datasets are handled using TF-IDF. By using TF-IDF with a count vectorizer in the LDA topic model, the system makes sure texts can be restored and processed efficiently. For spam filtering using various feature extraction methods, including TF-IDF, a model that offers high accuracy and quicker training times was constructed by Prosun et al [6].

For text sentiment analysis, Liu et al [7] designed a weight-distributing technique that integrated TF-IDF with a sentiment dictionary. As an alternative, Dessi et al. [8] applied TF-IDF and word embeddings to analyze clinical notes and determine patient health status. Their findings reveal that traditional TF-IDF methods performed better than deep learning techniques, highlighting their ongoing relevance in text analysis.

To improve how professional knowledge is structured and expressed, the research by Zhang [9] integrated TF-IDF with Case-Based Reasoning. This approach improves the efficiency of case reasoning models by incorporating keyword weighting. Meanwhile, Soufyane et al [10] developed an intelligent medical chatbot using TF-IDF and Natural Language Processing (*NLP*) to diagnose illnesses and give personalized consultations. This chatbot demonstrates the beneficial impacts of TF-IDF across fields, reducing healthcare costs and improving access to medical information.

In the field of educational technology, Mohammed & Nazila [11] developed a model based on Bloom's taxonomy, leveraging Term Frequency-Part of Speech Inverse Document Frequency (*TFPOS-IDF*) and Word to Vector (word2vec) for classifying exam questions. By combining these techniques with multiple classifiers, their method achieved impressive accuracy in sorting exam questions. Sundaram et al [12] took a different approach applying TF-IDF to perform emotional analysis in text, organizing feelings into six categories: sadness, happiness, fear, disgust, anger, and surprise. This method allowed them to achieve notable results in emotion classification through their method.

The study by Wang [13] introduced an automatic English question-answering model that uses TF-IDF and unsupervised word-splitting algorithms to make online network education faster and more efficient. This model did better than other traditional techniques in terms of accuracy, performance, and F1 values. Similarly, Danyal et al [14] compared TF-IDF and Count Vectorizer on two movie review datasets to see which one was more precise and turned out that TF-IDF was slightly better.

Chen & Lin [15] introduced Sem-TF-IDF, an advanced version of TF-IDF, which is a simple unsupervised learning technique. This method uses Instruction-tuned large models (*IT-LLMs*) to identify relevant information in documents. This paper generalizes the classic TF-IDF as suitable for information retrieval applications. The study by Sul & Cho [16] combined quantitative and qualitative research methods, including TF-IDF, in their evidence-based ethnographic approach to inform the design and development of Internet of Things (*IoT*) products and improve user experience.

In their study, Kim & Kim [17] developed a dynamic data preprocessing technique that uses TF-IDF and sliding windows to recognize harmful activities efficiently. Their findings have shown a significant improvement in malware detection accuracy. On the other hand, Zhang [18] examined the use of TF-IDF vectorization, collaborative filtering, and deep learning recommendation techniques, demonstrating how merging these methods with deep learning can enhance the precision and relevance of recommendations.

To address challenges in rough classification, text summarization, and response analysis, Iwendi et al [19] introduced a temporal Louvain algorithm combined with the TF-IDF algorithm. This combination led to significant improvement in execution time across various datasets and accuracy. Similarly, Mishra et al [20] applied TF-IDF and cosine similarity to analyze user reviews and provide more accurate hotel recommendations.

In their study, Bounabi et al [21] evaluated different machine learning algorithms to enhance classification performance. By applying Fuzzy TF-IDF and various machine learning techniques, they discovered

that combining the Naïve Bayes classifier with TF-IDF resulted in higher accuracy when classifying exam questions according to the level of Cognitive Domain. This approach allowed them to achieve higher precision in the classification process.

The study by Yuan et al [22] aimed to improve topic classification accuracy by comparing different methods with Bidirectional Encoder Representation from Transformers (*BERT*), and as a result, their study found which methods were suitable in different areas. For instance, the BERT model works best for categorizing government microblogs, while the TF-IDF-BERT model performs better with texts that are either longer than 140 words or shorter than 70 words. For texts between 70 and 140 words, LDA-BERT was the most effective and it works well on We Media Microblogs. Similarly, Du et al [23] in their study used various machine learning techniques based on TF-IDF to improve the detection of harmful apps, making mobile environments safer.

It has become challenging to choose the right beauty product these days, especially for facial products, with so many options and online reviews. To address this issue, Kirana & Al Faraby [24] make use of K-Nearest Neighbor (*KNN*) and TF-IDF algorithms to analyze customer reviews. These techniques helped them figure out which reviews were positive or negative and which words were important in deciding the review's sentiment. This information helped beauty companies better understand their customers. Similarly, Poulami et al [25] demonstrated how machine learning, along with TF-IDF and the Natural Language Toolkit (*NLTK*), can improve the classification of legal documents, making legal processes more efficient.

Application of TF-IDF in Security

Online platforms face issues like spam, fake users, and harmful content. To address these issues, researchers are utilizing machine learning and TF-IDF. This section examines studies that aim to improve security and prevent harmful activities, contributing to a safer online environment for all users.

To provide personalized recommendations for specialty sightseeing items, He, C., & Hua, C. [26] presented the use of collaborative filtering algorithms and user profiling aiming to improve the accuracy of services for tourists by addressing the large and scattered online travel industry.

In another text analysis application, Popoola et al [27] conducted sentiment analysis on tweets related to financial news using Naïve Bayes (*NB*), a Random Forest classifier (*RF*), and K-Nearest Neighbor (*KNN*) to analyze sentiment and capture public perceptions of financial news.

Time-Aware Term Frequency-Inverse Document Frequency (*TA TF-IDF*), a refined TF-IDF algorithm, was introduced by Zhu et al [28] to identify trending topics by incorporating time distribution information with user attention. In another study, Alammary [29] developed multiple datasets of Arabic assessment questions as per Bloom's taxonomy. This study concluded that this approach significantly improved the classification accuracy compared to traditional methods.

Table 1 shows the summary of algorithms and methodologies that are implemented in selected datasets on articles.

Table 1: Algorithm and Dataset used by different Researchers

Authors	Dataset	Algorithm and Methodology
(Raut et al., 2023)	Social media	TF-IDF, Naïve Bayes and SVM
(Lumintu, 2023)	E-commerce	TF-IDF and Cosine Similarity
(Xia, 2024)	Network teaching	Deep learning and Filtering
(Zhou et al., 2020)	News text	TF-IDF and Count Vectorizer
(Prosun et al., 2021)	Spam	TF-IDF & Retrieval techniques
(Popoola et al., 2021)	Tweets on financial news	TF-IDF and Naïve Bayes
(Liu et al., 2022)	Text	TF-IDF and Sentiment dictionary
(Dessi et al., 2021)	Clinical notes	TF-IDF and Word embeddings
(Zhang, 2021)	Professional knowledge	TF-IDF and Case-Based Reasoning
(Soufyane et al., 2021)	Medical	TF-IDF and NLP
(Mohammed & Nazlia, 2021)	Exam questions	TF-IDF and word2vec
(Sundaram et al., 2021)	Text	TF-IDF
(Wang, 2024)	English questions answers	TF-IDF
(Danyal, 2024)	Movie reviews	TF-IDF and Count vectorizer
(Chen & Lin, 2024)	Document	Sem-TF-IDF and IT-LLMs
(Sul & Cho, 2024)	IoT product design	TF-IDF
(Kim & Kim, 2024)	Dynamic analysis	TF-IDF and sliding windows
(Zhang, 2023)	Recommendation	TF-IDF and Collaborative Filtering
(Iwendi et al., 2019)	Various	TF-IDF and Temporal Louvain
(Mishra at al., 2019)	Hotel reviews	TF-IDF and Cosine similarity
(Zhu et al., 2019)	News	TA-TF-IDF and K-means
(Alammary, 2021)	Arabic assessment questions	TF-IDF
(Bounabi et al., 2020)	Exam questions	TF-IDF and Naïve Bayes
Our research	Abstracts	TF-IDF and Z-score

III. Methodology

This section outlines the theoretical foundations and mathematical calculations for TF-IDF, focusing on its application to the unique words in document abstracts. It covers calculations such as mean, standard deviation (STD), normalization, Z-score, and weighted balance. It then describes the practical applications of the methods used and the results achieved.

This study has employed the Term Frequency – Inverse Document Frequency (TF-IDF) algorithm to list the most similar articles by comparing their abstracts. The methodology involved analyzing 49 randomly chosen articles, published between 2021 and 2024, that were relevant to the topic of this study and based on the keyword "TF-IDF". A single target abstract was used for comparison.

Term Frequency (TF) measures the frequency of a term in a document. It is calculated as shown in **Equation 1**:

$$TF = \frac{\#\ of\ term\ frequency}{total\ \#\ of\ terms\ in\ a\ document} \qquad (1)$$

Inverse Document Frequency (*IDF*) measures how relevant a term is in the collection of documents. It is calculated as shown in **Equation 2**:

$$IDF = log\frac{\#\ of\ documents}{\#\ of\ documents\ containing\ the\ term} \qquad (2)$$

TF-IDF is the multiplication of TF and IDF which are shown above. The $Mean$ is one of the measures of central tendency that is calculated by adding the total TF-IDF values in each document and dividing it by the total number of unique words. It is calculated as shown in **Equation 3**:

$$Mean = \frac{sum\ of\ TF{-}IDF}{total\ \#\ of\ unique\ words} \qquad (3)$$

Standard deviation (*STD*) is the measure of variation in a dataset. It tells how far the values deviate from the mean and it is calculated as illustrated in **Equation 4** where x represents each value, and $\bar{x}$ represents the mean value of all values while n denotes the number of values.

$$STD = \sqrt{\frac{\sum(x-\bar{x})}{n-1}} \qquad (4)$$

Normalization means scaling data in numeric variables within the range of 0 and 1. **Equation 5** shows how to calculate the mean:

$$Normalization = X_i - \bar{X} \qquad (5)$$

The Z-score is a measure of how many STDs are above or below the mean data point. It is calculated as shown in **Equation 6**:

$$Z - score = \frac{N_i}{STD_i} \qquad (6)$$

The weighted value is the summation of the product of Z-score$_i$ and Z-score50 divided by the product of STD$_i$ and STD$_{50}$. It is calculated as shown below in **Equation 7**:

$$W_{i,50} = \frac{\sum Z{-}score_i * Z{-}score_{50}}{STD_i * STD_{50}} \qquad (7)$$

Where Z-score$_i$ is the Z-score of *i*th documents from the relevant 49 articles, and Z-score$_{50}$ is the Z-score of the 50th document whereas STD$_i$ is the STD of *i*th document from the 49 articles and STD$_{50}$ is the STD of the *50*th document. The 50th document is the target document that is the abstract of this paper to find the most relevant articles.

The initial dataset consisted of a table with columns for Document_Id, Document, link, title, authors, and year of publication. To prepare the data, we performed data cleaning through common words and punctuation removal on the "Document" column. We then stored the unique words and created a new table called Processed Document Abstracts, which listed the Unique words for each of the documents. Once we had this table ready, we calculated TF-IDF, Mean, Standard Deviation, Normalization, Z-Score and, Similarity Estimation. Using the similarity estimation results, we ranked the abstracts from highest to lowest based on their similarity scores.

The process flow, from preprocessing data to ranking publications based on their similarity estimation, is shown in **Figure 1**.

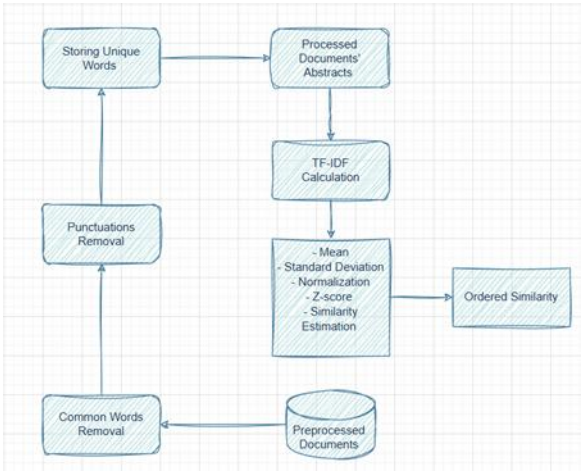

Figure 1: Process flow diagram

The preprocessing stage which involves the removal of common words and punctuations, plays a crucial role in improving the accuracy and relevance of the final rankings. Common words such as "the", "and", and "for" do not carry significant meaning, so their removal allows the algorithm to focus on the content of the documents, which leads to more accurate TF-IDF calculations. Additionally, preprocessing makes the algorithm more efficient by reducing the number of terms to process, which speeds up calculations and improves performance. Overall, the preprocessing stage creates a more focused and relevant dataset, which improves the accuracy and reliability of the final ranking.

IV. Results And Discussion

The first 29 results are listed by document IDs based on weight calculations as shown below in **Table 2**. In the list, the most relevant articles appear at the top based on the target abstract. The second column in the table represents the references mentioned on the reference lists.

Table 2: Experiment Result

Similarity Order	Reference from list
1	(Zhang, 2021)
2	(Yuan et al, 2020)
3	(Lumintu, I. 2023)
4	(Soufyane, et al., 2021)
5	(Kim & Kim, 2024)
6	---
7	(Danyal et al., 2024)
8	(Mohammed & Omar, 2020)
9	(Zhu et al, 2019)
10	(Mishra & Urolagin, 2019)
11	(Zhang, 2024)
12	(Poopola, 2024)
13	(Sul & Cho, 2024)
14	(Alammary, 2021)
15	(Iwendi et al, 2019)
16	(Liu et al, 2022)
17	(Zhou et al, 2021)
18	---
19	(Wang, 2024)
20	(Dessi et al, 2021)
21	(Sundaram et al, 2021)
22	(Chen & Lin, 2024)
23	(Prosun et al., 2021)
24	(Reddy et al., 2023)
25	(Raut et al, 2023)
26	(Xia, 2024)
27	(Wang et al., 2024)
28	---
29	(He & Hua, 2023)

Some abstracts score higher in similarity due to their closer alignment with the target abstract in terms of topic and key terms. The preprocessing step, where common words, numbers, and punctuation were removed, focused on unique and meaningful terms to make it easier. These unique terms played a big role in the weight calculations, so abstracts with more shared important terms ranked higher in similarity.

All the references from this study and some extra ones were examined. Then, after the abstracts were preprocessed, like removing common words, digits, and punctuations, the dataset was examined in the application. The application was developed using C# and implemented on MS SQL.

In total 49 research abstracts were used to compare similarity with the target abstract that is the abstract of this study. While this paper was written 29 articles were used as a reference already. As the aim of the study was to find out the success of the project after we examined the 49 abstracts, we took the top 29 abstracts as seen in **Table 2**. In the table, we put --- for the ones in 49 abstracts that we never used in our article. We calculated our success by the rate of how many abstracts of the references we used in this article.

V. Conclusion

The primary purpose of this paper was to enhance the accuracy of information retrieval by determining the relevance of terms within documents and across the collection of documents. The application was developed by integrating MS-SQL and C# programming, which processes and stores article abstracts, extracts unique terms, and calculates the relevance of these terms in documents.

A key challenge addressed in this study was the lack of a predefined database containing abstracts of articles on the specific topic. To overcome this limitation, we created a dataset by selecting 49 articles randomly related to the topic of interest. While this dataset was relatively small, it provided a valuable basis for applying and validating the methodology.

The implementation showed the practical applications of the TF-IDF algorithm, enhanced with additional statistical techniques namely mean, standard deviation, normalization, Z-score, and weight balance calculations. The results showed that approximately 89.66 % of the referenced articles were correctly identified as the most relevant publications, highlighting the effectiveness of this approach.

In our future work, we plan to expand the dataset significantly and we will compare the efficiency of alternative similarity measures, such as cosine similarity to assess their impact on relevance evaluation.

The research contributes to advancing current knowledge in information retrieval by addressing the limitations of dataset availability and demonstrating how normalization techniques, such as Z-score and weighted balance, improve the accuracy of document ranking. By bridging these gaps, the study provides valuable insights for improving information retrieval systems and their practical applications.

References

[1] Raut, R. M., Et Al. (2023) "Social Media Content Filtering Using Machine Learning Techniques." International Journal Of Advances In Engineering And Management (IJAEM). 794-798.

[2] Reddy, M. P., Et Al. (2023)"Identifying And Filtering Out Inauthentic Users On Social Media." International Journal Of Advanced Research In Science And Technology: 299-306.

[3] Lumintu, I. (2023). Content-Based Recommendation Engine Using Term Frequency-Inverse Document Frequency Vectorization And Cosine Similarity: A Case Study. In 2023 IEEE 9th Information Technology International Seminar (ITIS) (Pp. 1-6). IEEE.

[4] Xia, K. (2024). Personalized Recommendation For Network Teaching Courses Based On Combined Filtering Of Deep Learning And K-Means. In Intelligent Computing Technology And Automation (Pp. 1159-1165). IOS Press.

[5] Zhou, Z., Qin, J., Xiang, X., Tan, Y., Liu, Q., & Xiong, N. N. (2020). News Text Topic Clustering Optimized Method Based On TF-IDF Algorithm On Spark. Computers, Materials & Continua, 62(1).

[6] Prosun, P. R. K., Alam, K. S., & Bhowmik, S. (2022). Improved Spam Email Filtering Architecture Using Several Feature Extraction Techniques. In Proceedings Of The International Conference On Big Data, Iot, And Machine Learning: BIM 2021 (Pp. 665-675). Springer Singapore.

[7] Liu, H., Chen, X., & Liu, X. (2022). A Study Of The Application Of Weight Distributing Method Combining Sentiment Dictionary And TF-IDF For Text Sentiment Analysis. IEEE Access, 10, 32280-32289.

[8] Dessi, D., Helaoui, R., Kumar, V., Recupero, D. R., & Riboni, D. (2021). TF-IDF Vs Word Embeddings For Morbidity Identification In Clinical Notes: An Initial Study. Arxiv Preprint Arxiv:2105.09632.

[9] Zhang, L. (2021). Research On Case Reasoning Method Based On TF-IDF. International Journal Of System Assurance Engineering And Management, 12(3), 608-615.

[10] Soufyane, A., Abdelhakim, B. A., & Ahmed, M. B. (2021). An Intelligent Chatbot Using NLP And TF-IDF Algorithm For Text Understanding Applied To The Medical Field. In Emerging Trends In ICT For Sustainable Development: The Proceedings Of NICE2020 International Conference (Pp. 3-10). Cham: Springer International Publishing.

[11] Mohammed, M., & Omar, N. (2020). Question Classification Based On Bloom's Taxonomy Cognitive Domain Using Modified TF-IDF And Word2vec. Plos One, 15(3), E0230442.

[12] Sundaram, V., Ahmed, S., Muqtadeer, S. A., & Reddy, R. R. (2021). Emotion Analysis In Text Using TF-IDF. In 2021 11th International Conference On Cloud Computing, Data Science & Engineering (Confluence) (Pp. 292-297). IEEE.

[13] Wang, H. (2024). Automatic Question-Answering Modeling In English By Integrating TF-IDF And Segmentation Algorithms. Systems And Soft Computing, 6, 200087.

[14] Danyal, M. M., Khan, S. S., Khan, M., Ullah, S., Ghaffar, M. B., & Khan, W. (2024). Sentiment Analysis Of Movie Reviews Based On NB Approaches Using TF–IDF And Count Vectorizer. Social Network Analysis And Mining, 14(1), 1-15.

[15] Chen, W., & Lin, B. (2024). Sem-TF-IDF: A Simple Semantic Approach To Generalize TF-IDF By Employing Instruction Tuned Large Language Models.

[16] Sul, S., & Cho, S. B. (2024). Understanding People's Attitudes In Iot Systems Using Wellness Probes And TF-IDF Data Analysis. Multimedia Tools And Applications, 1-20.
[17] Kim, M., & Kim, H. (2024). A Dynamic Analysis Data Preprocessing Technique For Malicious Code Detection With TF-IDF And Sliding Windows. Electronics, 13(5), 963.
[18] Zhang, S. (2024). Restaurant Recommendation System Based On TF-IDF Vectorization: Integrating Content-Based And Collaborative Filtering Approaches. In 2023 International Conference On Data Science, Advanced Algorithm And Intelligent Computing (DAI 2023) (Pp. 610-618). Atlantis Press.
[19] Iwendi, C., Ponnan, S., Munirathinam, R., Srinivasan, K., & Chang, C. Y. (2019). An Efficient And Unique TF/IDF Algorithmic Model-Based Data Analysis For Handling Applications With Big Data Streaming. Electronics, 8(11), 1331.
[20] Mishra, R. K., & Urolagin, S. (2019). A Sentiment Analysis-Based Hotel Recommendation Using TF-IDF Approach. In 2019 International Conference On Computational Intelligence And Knowledge Economy (ICCIKE) (Pp. 811-815). IEEE.
[21] Bounabi, M., El Moutaouakil, K., & Satori, K. (2019). Text Classification Using Fuzzy TF-IDF And Machine Learning Models. In Proceedings Of The 4th International Conference On Big Data And Internet Of Things (Pp. 1-6).
[22] Yuan, H., Tang, Y., Sun, W., & Liu, L. (2020). A Detection Method For Android Application Security Based On TF-IDF And Machine Learning. Plos One, 15(9), E0238694.
[23] Du, W., Ge, C., Yao, S., Chen, N., & Xu, L. (2023). Applicability Analysis And Ensemble Application Of BERT With TF-IDF, Textrank, MMR, And LDA For Topic Classification Based On Flood-Related VGI. ISPRS International Journal Of Geo-Information, 12(6), 240.
[24] Kirana, Y. D., & Al Faraby, S. (2021). Sentiment Analysis Of Beauty Product Reviews Using The K-Nearest Neighbor (KNN) And TF-IDF Methods With Chi-Square Feature Selection. Journal Of Data Science And Its Applications, 4(1), 31-42.
[25] Poulami, B., Dr. Dinabandhu, B., & Koustav, S. (2024)." Using Machine Learning Algorithms With TF-IDF To Generate Legal Petitions ." International Journal Of Engineering Research & Technology (IJERT): 1-4.
[26] He, C., & Hua, C. (2023). Research On User Profile Combined With Collaborative Filtering Recommendation Algorithm For Intelligent Tourism. Academic Journal Of Science And Technology, 7(1), 63-69.
[27] Popoola, G., Abdullah, K. K., Fuhnwi, G. S., & Agbaje, J. (2024). Sentiment Analysis Of Financial News Data Using TF-IDF And Machine Learning Algorithms. In 2024 IEEE 3rd International Conference On AI In Cybersecurity (ICAIC) (Pp. 1-6). IEEE
[28] Zhu, Z., Liang, J., Li, D., Yu, H., & Liu, G. (2019). Hot Topic Detection Based On A Refined TF-IDF Algorithm. IEEE Access, 7, 26996-27007.
[29] Alammary, A. S. (2021). Arabic Questions Classification Using Modified TF-IDF. IEEE Access, 9, 95109-95122.