Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th June 2025. Vol. 4. No. 3; e-ISSN: 2808-4519

Utilization of TF-IDF Weighting in Song Search System Based on Spotify Lyrics

Eviriawan[1]*, Nelti Juliana Sahera[2], Hikma[3], Syaban Barokah Nur Ilahi[4]

[1,2,3,4] Informatics Engineering, Universitas Halu Oleo
eviriawan052004@gmail.com [1], neltijuliana@gmail.com [2], hikmaaaa@gmail.com [3], syabanbarokahnurilahi@gmail.com [4]*

Abstract

In the rapidly developing digital era, the need for an efficient information retrieval system is increasing. Spotify, as one of the largest music streaming platforms, faces challenges in providing a fast and accurate song search system. Improving user experience in searching for song titles based on lyrics is the main focus in developing a search system on the music streaming platform. like Spotify. Study This explore use method weighting using TF-IDF (Term Frequency- Inverse Document Frequency) to optimize the search for song titles through lyrics. By applying TF-IDF, system can assess and weighting words in lyrics based on the frequency in One song and its uniqueness in gathering song data in overall. As for the data that used in this study totaling 30 entries. The methods used include system design, preprocessing (data cleaning, tokenization, filtering, and stemming), and TF-IDF weighting. The test results show that this approach significantly improves the relevance and accuracy of search results, making it easier for users to find the appropriate song title. with lyrics Which they remember. System Which proposed This expected can repair quality search services on Spotify and provide a more satisfying experience for users.

Keywords: *relevance, search system, song lyrics, song title, spotify, TF-IDF*

1. Introduction

Song is a type the voice that has rhythm and become one of the entertainment media known wide between modern society today this . With Thus , the song become form expression a sound that is rhythmic and becomes one choice favorite entertainment between today's society. Over time, songs have undergone many changes, both in terms of music and lyrics. In terms of music, technological developments have brought innovations in composition and sound processing, as well as more and more blends between genres that produce new musical colors. Today, the themes of song lyrics have expanded beyond traditional topics such as love, friendship, and life. Song lyrics are now increasingly diverse, raising various issues such as freedom, phenomena that are currently popular with the public, and using expressions that are currently trending among the public.[1].

Listening to songs has become a habit in daily activities with various purposes such as motivating listeners to exercise. [2], or to relieve stress and can affect the emotions of the listener. In a study conducted by Situmorang showed that the activity of listening to music can affect a person's thinking function and mental power in completing tasks.
In addition to functioning as a means of entertainment among modern society, music also continues to develop with many works being created, providing society with a variety of song choices according to their preferences. Factors that influence a person's choice of song can vary, such as musical genre, vocal artist, launch period, and song text content. This creates the need for a mechanism that can facilitate the public in identifying musical compositions based on these elements, especially song search platforms that use lyrics as a search basis. The presence of this kind of technology is expected to make it easier for the public to identify the title of a musical composition that matches the lyric fragment they want to listen to.
Thus, this study aims to help people find songs on Spotify more easily by implementing TF-IDF weighting. With this method, users only need to write a few words of the lyrics of the song they remember, and the search system will be able to display the song title that matches the lyrics. The application of TF-IDF aims to improve the accuracy and relevance of search results, allowing users to find the songs they are looking for faster, even if they only remember a small part of the lyrics. Thus, this lyrics-based song search system not only meets the needs of people to find songs based on their memory of the lyrics, but also improves the overall music listening experience.
In previous research by Tirtana [3], on the Creation of a Job Search System Using TF-IDF with an average search per keyword *of* 0.9458. research conducted by Arif and Ipam [4]on the Final Project Title Similarity Search System Using the TF-IDF Method, with the results of the TF-IDF trial on 384 final project titles, 99 titles were found to have similarities based on keywords. The TF-IDF method can be a solution for lecturers and students in determining relevant final project titles. Research by Pribadi and Munir[5], on the Application of Graph Node Coloring with the Welch Powell Algorithm and Depth First Search Algorithm on the Distribution of Turns to Play Angklung. As well as research by Adhimas and Arif [6], on Recommendation Systems Using Indonesian Song Content-Based Filtering

and Cosine Similarity Methods. The results of the study indicate that the system can present appropriate and appropriate song recommendations based on the lyrics.
Based on the description above, the use of the TF-IDF method has good results in the context of information retrieval. TF-IDF is a technique in *text mining* that analyzes data and recognizes important keywords in documents. One of the advantages of the TF-IDF method is its ability to handle text searches by considering the intensity of the appearance of terms in a manuscript and the number of text files containing the terminology in question, which helps in determining the relevance of the words.[7]. This method has a high level of accuracy and relatively low errors in finding interrelated information. Thus, this study aims to make it easier for people to search for songs on Spotify by simply entering a few words from the lyrics they remember using the TF-IDF method.

2. Theoretical Basis

2.1. Song

A song is a combination of musical elements composed of various tones and texts that have meaning. Song lyrics describe a person's feelings about something they have seen, heard, or experienced. To express this experience, the songwriter or poet plays with diction and style of language to create charm and characteristics in the lyrics or poems they create.[8].
As time progresses, the number of songs available continues to grow rapidly. Various genres and types of songs emerge. The song itself is created from a harmonious blend of musical elements and lyrical elements which are a form of mass communication.

2.2. Lyrics

Song lyrics reflect the expression of a person's feelings towards everything in the form of auditory, visual, or personal experiences. The expression of these feelings is then manifested in language that forms its own uniqueness and interest [9]. In addition, the emotions contained in the lyrics are often the main consideration for users when searching for songs. Song lyrics can also be considered as poetry composed in simple language but containing deep meaning.[10].

2.3. TF-IDF method

TF-IDF is one of the classic techniques for keyword extraction that functions to evaluate the significance of a term in a text. This approach has two main components, namely Term Frequency (TF) and Inverse Document Frequency (IDF). TF shows how often a word appears in a document, where the higher the frequency, the more important the word is. While IDF refers to the document where a word is in a corpus. In this corpus, words with high frequency are considered less representative and have a low level of importance. Conversely, words with low frequency are considered more representative and have a high level of importance [9].
The TF-IDF calculation process can be done through several stages described in equations 1 to 3. The formula for calculating term frequency is as follows:

$$(tf_{t,d})tf_j = 1 + \log tf_j \quad (1)$$

Information :
tf_j = Number of occurrences of term (j)

Formula calculation *document frequency*:

$$idf_j = \log\left(\frac{N}{df_j}\right) \quad (2)$$

Formula calculation *Term Frequency (TF)-inverse document frequency (IDF):*

$$W_{t,d} = tf_j * idf_j \quad (3)$$

Information:
$W_{t,d}$*Term* weight (t) in document (d)
idf_j*= Inverse document frequency* in *term* (j)
tf_j*= Term frequency* in *term* (j)

2.4. Text Mining

Text mining refers to the process of analyzing textual data to uncover hidden knowledge from unstructured sources such as electronic documents, office files, and other text snippets. This process aims to group and categorize text, and can also be used in summarization processes. [9][11].

3. Research Methods

3.1. Stages Study

Stages carried out in research This use TF-IDF method for make it easier public in look for A songs on Spotify with user only need write some lyrics words from the song they remember. This study uses the TF-IDF method, where the stages start from data collection, pre-processing, and TF-IDF weighting. The stages or framework in this study can be seen in Figure 1.

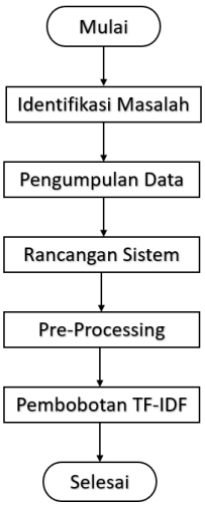

Fig. 1: Research Flow

3.2. Data Collection

Data or sample is component crucial to a study scientific, where the type of data is used play a role in influence determination approach proper analysis use solve problem. In this study, data was collected manually from the Spotify application and includes three columns, namely song name, artist name, and song lyrics. The number of samples analyzed in study This reach 30 units of data.

3.3. Design System

At this stage, the system is designed and developed using the Python programming language for a song search application based on lyrics on Spotify. The system architecture implemented is *the Front-end* (User Interface) and *Back-end architecture* . The front-end is responsible for providing a user interface to enter lyrics words and display the search results for song names and artist names.[12]. Meanwhile, *the back-end* system contains business logic, search algorithms, and interacts with the database. The algorithm uses text processing techniques such as TF-IDF to match lyrics words with song data stored in the Spotify database. This database stores important metadata such as song names, artist names, and lyrics to ensure accurate and relevant search results for users. [13]. With this architecture, the song search application on Spotify can provide an efficient user experience and make it easier for them to find songs according to their preferences based on the desired lyrics.

3.4. Pre-processing

Stage Data Pre -Processing is fundamental steps in processing information that works For ensure that sample in condition processed and have good quality before used in modeling. This process includes a series of systematic procedures designed to improve the quality of raw data, addressing challenges such as inconsistencies, missing values, or data format inconsistencies. [14][15]. In the context of Text Mining, this stage plays an important role in reducing data complexity and speeding up computation time by focusing only on relevant terms. Therefore, this pre-processing stage consists of four steps, namely data cleaning, tokenization, stopword removal (filtering), and stemming.

3.4.1. Cleaning

Data cleaning is stage beginning in the pre-processing process in research This process aims to remove non - textual elements such as numbers, web links, and symbols that do not contribute to the content of the document, such as (0-10,!@#$%&*_+-={}[]:;”’/?<>.) [16].

3.4.2. Tokenization

One of the steps in the *pre-processing process* that aims to break down the document text into separate words. This process is usually done by using spaces as a reference, because spaces function as separators between words in a sentence. [8].

3.4.3. Filtering (Filtering Process)

Filtering (filtering process), in this stage, irrelevant terms are eliminated from the tokenization output. Words that are considered unimportant are called stopwords. Stopwords generally reach a significant number of occurrences in the text, and their removal will not affect the meaning of the sentence. For example, the words "yang", " ke ", "di", " dari ". [8].

3.4.4. Stemming

at this stage, the stemming process is carried out by changing each keyword into its basic form. This process is carried out after going through the filtering stage. For example, the word "menggunakan" will be returned to its basic form by removing the prefix me- and the suffix -kan, so that it becomes "guna.". [8].

3.5. TF-IDF weighting

In research conducted by Baeza-Yates and Ribeiro-Neto (1999), argued that system TF-IDF weighting is formed by two components main, namely:

3.5.1. Term Frequency (TF)

Term Frequency (TF) measures frequency emergence a word (fi) in document dj, relative against other words (fl) which are more dominant in the same document. In mathematical, concept This can formulated as following:

$$f_{i,j} = \frac{freq_{i,j}}{max_i\, freq_{i,j}} \tag{4}$$

3.5.2. Inverse Document Frequency (IDF)

Inverse Document Frequency (IDF) is used for see how much seldom or how much general A the term fi appears in gathering documents. The reason for using IDF is because words that appear too often in almost all documents are usually not very helpful in distinguishing which documents are relevant and which are not. If the total number of documents in a collection is written as N, and ni is the number of documents containing the term fi, then the IDF value for that term can be calculated using the following formula:

$$idf_i = log\, \frac{N}{n_i} \tag{5}$$

Information:
IDF = *Inverse Document Frequency*
N = number sentences containing the term (t)
ni = number of occurrences of the word (term) in dj

The weight for each word in a document is determined through a combination of Term Frequency (TF) and Inverse Document Frequency (IDF). By considering these two factors, the TF-IDF weighting can be formulated as follows:

$$w_{i,j} = f_{i,j} \times log\, \frac{N}{n_i} \tag{6}$$

$$W_{i,j} = tf_{i,j} \times idf_j \tag{7}$$

Information :
$w_{i,j}$= Weight value of word j of document i
$tf_{i,j}$= *Term frequency* , namely the number of occurrences of the word tj in the document Di
df_j= *Document frequency* , namely the number of documents containing tj
IDF $_j$ = log ! !" where d is the number of all documents in the collection. IDFj is *the inverse document frequency* .

In this method, determining the weight of a word in a document is done by multiplying *the Term Frequency* (TF) and *Inverse Document Frequency* (IDF) values. The weight reflects how often the word appears in the document (TF), while also considering how many other documents also contain the word (IDF). The more often a term appears in a particular document, the greater its weight, while the weight will be smaller if the term is spread across many documents. [17].
After that, the sorting process is carried out based on the cumulative *W value* of each sentence. The three sentences with the highest *W value* will be selected as the main summary and will be used as the output of the automatic text summarization process.[18] .

4. Results and Discussion

4.4.1. Data Collection Results

In the Data Collection stage, the process of taking data from the Spotify application is carried out to build a song classification model using the TF-IDF method. This process aims to ensure data diversity, completeness, and accuracy in filling in order to avoid missing values. Data is collected manually from the Spotify application and includes three main columns: song name, artist name, and song lyrics. Figure 2 shows the song dataset used in this study.

	Nama Lagu	Nama Artis	Lirik Lagu
1	Mati-Matian	Mahalini	Kita adalah dua insan penuh cinta Di awal tercipta kisah kita Manis tuturmu buatmu ……
2	Sial	Mahalini	Sampai saat ini tak terpikir olehku Aku pernah beri rasa pada orang sepertimu Seandainya ……
3	Sisa Rasa	Mahalini	Ha-ah-ah-ah Melihatmu bahagia, satu hal yang terindah Anup'rah cinta yang pernah ……
4	Bohong Hati	Mahalini	Aku tersiksa Melihat semuanya berubah Mengapa Kau tak mau tahu Bagaimana hati ini ……
5	Melawan Restu	Mahalini	Hm-mm Indah semua cerita Yang t'lah terlewati dalam satu cinta Kita yang pernah ……
6	Bawa Dia Kembali	Mahalini	Malam ini aku menanti Kedatanganmu mengisi sepiku Lama terasa waktu bergulir ……
7	Kisah Sempurna	Mahalini	Ha-ah, ha-ah-ah Tenggelam, jiwaku dalam angan Tersesat, hilang, dan tak tahu arah Ku ……
8	Aku Yang Salah	Mahalini	Ku tak mengerti Betapa bodoh diri Membuarkanmu pergi Jauh dari hati Ku coba mengobati ……
9	Ini lagiuku	Mahalini	Apalah arti hidupku bila tanpamu Apalah arti cintaku bila bukan kamu Semua perjuanganku ……
10	Janji Kita	Mahalini	Di dalam hati ini Hanya ada satu cinta Dan itu cuma ada di kamu Kamu Dan kamu cinta ……
11	Jodoh Pasti Bertemu	Afgan	Andai engkau tahu Betapa ku mencinta Selalu menjadikanmu Isi dalam doaku Ku tahu, tak ……
12	Bawalah Cintaku	Afgan	Sumpah tak ada lagi Kesempatanku untuk bisa bersamamu Kini ku tau bagaimana cara ku……
13	Untukmu Aku Bertahan	Afgan	Tenanglah, kekasihku. Kutahu hatimu menangis, namun beranilah 'tuk percaya bahwa ……
14	Terima Kasih Cinta	Afgan	Tersadar di dalam sepiku setelah jauh melangkah, cahaya kasihmu menuntunku kembali ……
15	Putar Waktu	Mahalini	Waktu ke waktu kulewati, melihat diriku tak kecil lagi, tuan dan putri mulai menghakimi ……
16	Padamu Kubersujud	Afgan	Ku menatap dalam kelam, tiada yang bisa kulihat selain hanya nama-Mu, ya Allah. Esok ……
17	Ku Dengannya Kau Dengan Dia	Afgan	Awalnya ku tak bermaksud apapun saat ku kenal dirimu. Kita hanya saling bercerita tentang…
18	Ada	Afgan	Ku tahu kisah kita sedang diuji waktu, seperti tak pernah terbayang kembali satu. Namun ……
19	Wajahmu Mengalihkan Dunisku	Afgan	Ketika kau lewati bumi tempatku berdiri, kedua mata ini tak berkedip menatapi. Pesona ……
20	Sadis	Afgan	Terlalu sadis caramu menjadikan diriku pelampiasan cintamu agar dia kembali padamu……
21	Yang Kutahu Cinta Itu Indah	Afgan	Tak pernah aku membayangkannya, bila insan sedang patah hati. Kali ini kurasakan ……
22	Cinta 2 Hati	Afgan	Tak kusangka dirimu hadir di hidupku, menyapaku dengan sentuhan kasihmu. Ku sesali ……
23	Pencari Jalanmu	Afgan	T'lah banyak yang ku lewati, jalan hitam di dunia, tak terhitung salah menodai. Masihkah ……
24	Bunga Terakhir	Afgan	Kaulah yang pertama menjadi cinta, tinggallah kenangan berakhir lewat bunga. Seluruh ……
25	Pesan Cinta	Afgan	Merah pipi ini saat ku lihat dirinya. Mungkinkah ini yang dinamakan cinta? Malu hati ini ……
26	Kunci Hati	Afgan	Teringat pada saat itu, tertegun lamunanku melihatmu. Tulus senyumanmu sejenak sudut ……
27	Pendam	Afgan	Setelah kujalani kisah kita, kini aku telah menyadari, kau dan aku takkan bisa, walau kita ……
28	Setia Menunggu	Afgan	Tuhan yang tahu betapa ku simpan rasaku, meski ini tak mungkin ku bersamamu. Sejak ……
29	Kumohon	Afgan	Setiap hariku, mohon agar kau senantiasa memberiku ketenangan dalam hati, kekuatan ……
30	Bukan Cinta Biasa	Afgan	Kali ini kusadari, aku telah jatuh cinta dari hatiku terdalam. Sungguh, aku cinta padamu……

Fig. 2: Song Dataset

4.4.2. Design Results System

The following are the results of the system design regarding song searches based on lyrics on the Spotify application using the TF-IDF method:

a. Home Page View

Figure 3 shows the home page that users will first see when they open the application.

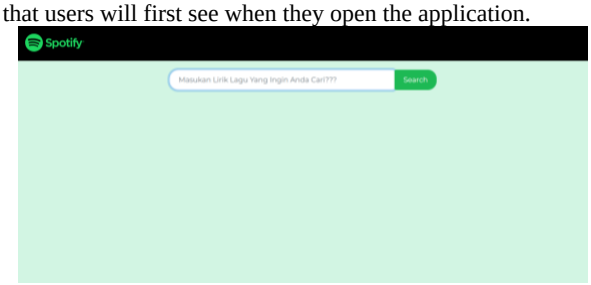

Fig. 3: Home Page

b. Appearance Song Search

In figure 4 is a song lyrics search page to display a list of songs that users want to listen to. For the search results, it will display information in the form of song titles, artist names, and weighting results.

Fig. 4: Song Search Page View

4.4.3. Pre-Processing Results

Preprocessing is a data cleaning stage that aims to remove irrelevant characters in text analysis. There are four main steps in this process, namely data *cleaning* , *tokenization*, *filtering* , and *stemming* .

a. Data Cleaning Stage Results

Figure 5 shows the results of the data cleaning process, where the text has been filtered from numbers, links (URLs), and various characters that have no direct connection to the content or meaning of the document, such as symbols (0-9, !@#$%&*_+-={}[]:;”'/?<>.).

Cleaned_text
Kita adalah dua insan penuh cinta Di awal tercipta kisah kita Manis .....
Sampai saat ini tak terpikir olehku Aku pernah beri rasa pada orang……..
Haahahah Melihatmu bahagia satu hal yang terindah Anugrah cinta ……
Aku tersiksa Melihat semuanya berubah Mengapa Kau tak mau tahu…...
Hmmm Indah semua cerita Yang tlah terlewati dalam satu cinta Kita…...
……
……
Teringat pada saat itu tertegun lamunanku melihatmu Tulus ……
Setelah kujalani kisah kita kini aku telah menyadari kau dan aku…….
Tuhan yang tahu betapa ku simpan rasaku meski ini tak mungkin……
Setiap hariku mohon agar kau senantiasa memberiku ketenangan ……
Kali ini kusadari aku telah jatuh cinta dari hatiku terdalam Sungguh……

Fig. 5: Results of stage *cleaning* data

b. Stage Results Tokenization

After stage *cleaning* data, tokenization process is carried out, namely break lyrics song into separate words, as shown in Figure 6.

Tokenized_text
[‘Kita’, ‘adalah’, ‘dua’, ‘insan’, ‘penuh’, ‘cinta’, ‘Di’, ‘awal’, ‘tercipta’,….]
[‘Sampai’, ‘saat’, ‘ini’, ‘tak’, ‘terpikir’, ‘olehku’, ‘Aku’,‘pernah’, ‘beri’,…]
[‘Haahahah’, ‘Melihatmu’, ‘bahagia’, ‘satu’, ‘hal’, ‘yang’, ‘terindah’,……]
[‘Aku’, ‘tersiksa’, ‘Melihat’ ‘semuanya’, ‘berubah’, ‘Mengapa’, ‘Kau’, ….]
[‘Hmmm’, ‘Indah’, ‘semua’, ‘cerita’, ‘Yang’, ‘tlah’,‘terlewati’,‘dalam’,…..]
……
……
[‘Teringat’, ‘pada’, ‘saat’, ‘itu’, ‘tertegun’, ‘lamunanku’, ‘melihatmu’,……]
[‘Setelah’, ‘kujalani’, ‘kisah’, ‘kita’, ‘kini’, ‘aku’, ‘telah’, ‘menyadari’ ,…..]
[‘Tuhan’, ‘yang’, ‘tahu’, ‘betapa’, ‘ku’, ‘simpan’,‘rasaku’,‘meski’,‘ini’ ,….]
[‘Setiap’, ‘hariku’, ‘mohon’, ‘agar, ‘kau’, ‘senantiasa’, ‘memberiku’,……..]
[‘Kali’, ‘ini’, ‘kusadari’, ‘aku’, ‘telah’,‘jatuh’, ‘cinta’,‘dari’,‘hatiku’,……..]

Fig. 6: Results of stage tokenization

c. *Stage Results Filtering*

After going through the tokenization process, the next step is filtering. At this stage, the system filters and removes words that are considered to have no major influence on the analysis, or are considered less relevant to the main context, as seen in Figure 7.

Filtered_text
dua insan penuh cinta
tak terpikir olehku Aku menyadari
Melihatmu bahagia satu hal tak
Aku tersiksa Melihat semuanya berubah
Indah cerita terlewati
……
……...
Teringat tertegun lamunanku
Setelah kujalani kisah kini aku menyadari
Tuhan tahu betapa kusimpan rasaku
Setiap hariku mohon kausenantiasa
Kali kusadari aku jatuh cinta

Fig. 7: Results of the filtering stage

d. *Stemming Stage Results*

The next stage is the stemming process, where the words that have been processed in the previous stage are converted to their basic form by removing the affixes. The results of the stemming process can be seen in Figure 8.

Stemmed_text
dua insan penuh cinta
tak pikir oleh aku sadar
lihat bahagia satu hal tak
aku siksa lihat semua ubah
indah cerita lewat
……
……
ingat tegun lamun
telah jalan kisah kini aku sadar
tuhan tahu betapa simpan rasa
tiap hari mohon senantiasa
kali sadar aku jatuh cinta

Fig. 8: Stage results *stemming*

4.4.4. TF-IDF Weighting Results

a. TF Calculation

The following table contains the keywords used in TF-IDF calculations.

Table 1: Keywords

keywords (kk): hati, cinta
D1 : hati ada satu cinta
D2 : kali sadar aku jatuh cinta
D3 : tenang kasih tahu hati menang
D4 : kau pertama jadi cinta tinggal
D5 : telahjalan kisah kini aku sadar
D6 : tuhan tahu betapa simpan rasa

Next, after determining the keywords that will be searched for in each document, the following are the results of the tf calculation, which can be seen in Figure 9.

		tf							
token	kk	D1	D2	D3	D4	D5	D6	df	D/df
hati	1	1	0	1	0	0	0	3	2
cinta	1	1	1	0	1	0	0	4	1.5
menang	0	0	0	1	0	0	0	1	6
tuhan	0	0	0	0	0	0	1	1	6
rasa	0	0	0	0	0	0	1	1	6
kisah	0	0	0	0	0	1	0	1	6
tenang	0	0	0	1	0	0	0	1	6

Fig. 9: TF calculation results

b. IDF Calculation

Formula *idf* :

$$idf = (\frac{1}{df}) \quad (8)$$

After to obtain results TF calculation, steps next is calculate IDF for each term of use determine its weight. The results of the IDF calculation can seen in Figure 10.

token	IDF (log D/df)
hati	0.301029996
cinta	-1.345449135
menang	0.77815125
tuhan	0.77815125
rasa	0.77815125
kisah	0.77815125
tenang	0.77815125

Fig. 10: IDF Calculation Results

c. *TF-IDF calculation*

Formula weighting *tf- idf* :

$$W_{t,d} = tf_{t,d} \times idf_t \quad (9)$$

Information:

W = weight of the dth document against the tth word

After the tf and idf values are successfully obtained, the next step is to enter them into the *tf-idf weighting calculation* to determine the weight of the association of a term in the document. The results of this TF-IDF calculation can observed in Figure 11.

	W						
token	kk	D1	D2	D3	D4	D5	D6
hati	0.30103	0.30103	0	0.301029996	0	0	0
cinta	-1.3454	-1.3454	-1.3454	0	-1.345	0	0
menang	0	0	0	0.77815125	0	0	0
tuhan	0	0	0	0	0	0	0.77815125
rasa	0	0	0	0	0	0	0.77815125
kisah	0	0	0	0	0	0.77815	0
tenang	0	0	0	0.77815125	0	0	0

Fig. 11: TF-IDF Calculation Results

d. *Application Results*

Figure 12 shows an application system that allows song searches using input in the form of lyrics from the user, where the search keywords can consist of several words that are part of the song lyrics.

Fig. 12: Search Results Page

The image above show search output song featuring title songs, artist identity, and values weighting for every Songs. The song with the highest weight value is displayed at the top, indicating that the song has the highest level of relevance to the keyword entered by the user in the search.

5. Conclusion

Based on the research above, it can be concluded that this research has successfully implemented the TF-IDF method for searching songs based on lyrics on the Spotify application. By using 30 song data entries, this system is designed with an efficient *front-end* and *back-end architecture* . The pre-processing stages including data cleaning, tokenization, filtering, and *stemming* have succeeded in increasing search accuracy. The test results show that the use of TF-IDF significantly increases the relevance and accuracy of search results, making it easier for users to find songs based on lyrics that users remember. This implementation aims to improve the quality of the search experience for Spotify users and make it more effective and provide higher satisfaction.

In further research and development for a lyrics-based song search system, the author provides several recommendations, namely, it is recommended to increase the amount of song data used so that the search model is more comprehensive and able to handle a wider

variety of lyrics, integrate *machine learning* or *deep learning techniques* to improve search accuracy and handle synonyms or variations of words in song lyrics, develop additional features such as searching by genre, release year, or song popularity to enrich the user experience and conduct testing by involving more users to get more varied *feedback and improve the application's user experience* (UX).

References

[1] G. Widianto Aldiansyah, P. Pandu Adikara, and R. Cahya Wihandika, "Cross Language Song Recommendation Based on Lyrics Using Word2VEC," *Journal of Information Technology and Computer Science Development* , vol. 3, no. 8, pp. 8036–8041, 2019, [Online]. Available: http://j-ptiik.ub.ac.id

[2] TF Keita Nakamura, "Music recommendation system using lyric network," *Global Conference on Consumer Electronics* , 2017.

[3] A. Tirtana, A. Zulkarnain, YD Listio, and S. Malang, "Creating a Job Search System Using TF-IDF," *Asian Information Technology Scientific Journal* , vol. 13, no. 2, 2019.

[4] IFA Arif Amrulloh, "Final Project Title Similarity Search System Using TF-IDF Method," *CoreIT Journal: Journal of Computer Science and Information Technology Research Results* , vol. 7, no. 2, p. 58, Dec. 2021, doi: 10.24014/coreit.v7i2.15338.

[5] F. Pribadi and R. Munir, "Implementation of Graph Vertex Coloring with Welch Powell Algorithm and Depth First Search Algorithm on Angklung Playing Turn Distribution," *CoreIT Journal: Journal of Computer Science and Information Technology Research Results* , vol. 7, no. 2, p. 58, Dec. 2021, doi: 10.24014/coreit.v7i2.15338.

[6] ANR Adhimas Yanto Timur, " Indonesian Song Recommendation System Using Content-Based Filtering and Cosine Similarity Methods," *Journal of Informatics and Applied Electrical Engineering* , vol. 13, no. 1, Jan. 2025, doi: 10.23960/jitet.v13i1.5949.

[7] I. Tri Hapsari, B. Satria Andoko, and C. Rahmad, " Information Retrieval Application for Research Report Document Search," *Polinema Informatics Journal* , vol. 1, no. 3, pp. 2407–070, 2015.

[8] F. Sarito Sinaga and B. Rahayudi, "Song Lyrics Emotion Classification using Improved K-Nearest Neighbor with Feature Selection and BM25," *Journal of Information Technology and Computer Science Development* , vol. 3, no. 6, pp. 5697–5702, 2019, [Online]. Available: http://j-ptiik.ub.ac.id

[9] MA Lestari, PP Adikara, and S. Adinugroho, "Song Recommendation Based on Lyrics and Song Genre Using Word Embedding Method (Word2Vec)," *Journal of Information Technology and Computer Science Development* , vol. 3, no. 8, pp. 2548–964, 2019, [Online]. Available: http://j-ptiik.ub.ac.id

[10] IP Sari, F. Febriyanti, TA Ujung, and FL Barus, "Analysis of Connotative Meaning in the Lyrics of the Song Bertaut by Nadin Amizah," *Diksa: Indonesian Language and Literature Education* , vol. 7, no. 1, pp. 22–32, Jun. 2021, doi: 10.33369/diksa.v7i1.15891.

[11] B. Hakim, " Sentiment Analysis of Text Data Preprocessing in Data Mining with Using Machine Learning," *JBASE - Journal of Business and Audit Information Systems* , vol. 4, no. 2, Aug. 2021, doi : 10.30813/ jbase.v 4i2.3000.

[12] NP Yunita and P. Correspondence , " Application Hadith Search Using Vector Space Model With Weighting Tf-Idf Dan Confix -Stripping Stemmer Hadith Search Application Using The Vector Space Model With Tf-Idf Weighting And Confix -Stripping Stemmer," *Journal Technology Information and Knowledge Computer (JTIIK)* , vol. 10, no. 3, pp. 665–676, 2023, doi : 10.25126/jtiik.2023106736.

[13] M. Noveanto, H. Sastypratiwi, and H. Muhardi, "Test of Emotion Classification Accuracy in Indonesian Song Lyrics," *Journal of Information Systems and Technology (JustIN)* , vol. 10, no. 3, p. 311, Nov. 2022, doi: 10.26418/justin.v10i3.56804.

[14] Q. Qi, DJ Hessen, T. Deoskar , and PGM Van Der Heijden, "A comparison of latent semantic analysis and correspondence analysis of document-term matrices," *Nat Lang Eng* , vol. 8, no. 10, May 2023, doi : 10.1017/S1351324923000244.

[15] SAN Alexandropoulos, SB Kotsiantis , and MN Vrahatis , "Data preprocessing in predictive data mining," *Knowledge Engineering Review* , vol. 34, 2019, doi : 10.1017/S026988891800036X.

[16] F. Rahutomo, F. Rozi, PN Mirzha, T. Informatika, and PN Malang, " Evaluation of the Implementation of the Improved K-Nearest Neighbor Algorithm in Categorizing Indonesian Song Lyrics According to Age," *JIP (Polinema Informatics Journal)* , 2020.

[17] F. Amin, " Implementation of Search Engine Using Vector Space Model Method," *Engineering Dynamics* , pp. 45–58, Jan. 2011.

[18] LW Astuti, A. Rachmat C., and Y. Lukito, " Implementation of Naïve Bayes Algorithm Using Isear for Classification of Emotions in English Song Lyrics," *Jurnal Informatika* , vol. 14, no. 1, May 2017, doi: 10.9744/informatika.14.1.16-21.