Journal of Information Systems and Informatics
Vol. 5, No. 4, December 2023 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v5i4.566 Published By DRPM-UBD

Machine Learning-Based E-Archive for Archives Management of South Sumatra Province

Toni Tri Atmojo[1], Yesi Novaria Kunang[2]

[1,2]Magister Teknik Informatika, Universitas Bina Darma, Palembang Indonesia
Email: [1]toni.triatmojo@binadarma.ac.id, [2]yesinovariakunang@binadarma.ac.id

Abstract

Archives play a crucial role in institutional operations, yet efficiently retrieving specific information from them can be challenging. This research addresses this issue by developing an information retrieval system that incorporates advanced methods to enhance search efficiency. The system employs the TF-IDF (Term Frequency-Inverse Document Frequency) formula, which assesses the significance of a word within a document set, and the BM25 method, a sophisticated algorithm for ranking documents based on their relevance to the input query. Both methods undergo a preprocessing stage, enabling the system to calculate the relevance of each document to the given query accurately. The effectiveness of this system is evaluated using key performance metrics: precision (accuracy), recall (completeness), and the F1 Score (the harmonic means of precision and recall, representing the best value). Testing with various keywords revealed that the BM25 method yielded impressive results, achieving an average precision of 0.75, recall of 0.6, and an F1 Score of 0.6665. In contrast, the TF-IDF method scored lower, with a precision of 0.33, recall of 0.2, and an F1 Score of 0.2500. The system was tested using a dataset of 350 documents.

Keywords: Information Retrieval, TF-IDF, BM25, Archives

1. INTRODUCTION

Along with current technology that is growing rapidly and increasing use as a source of online information providers from all over the world in finding information with the help of search engines. The increasing number of scattered articles and the growth of data size make it more difficult for users to find suitable articles, while users want to get information quickly and accurately [1]. Information retrieval aims to produce documents that are relevant to user needs from a collection of information automatically based on keywords. One of these data sources is archives. With the development of ICT today, it has an impact on archive management which can be done electronically which is the process of converting archives from paper sheets into electronic sheets [2].

Archives are very important in an institution, one of which is the South Sumatra Provincial Archives Office which acts as a coordinator and carries out guidance

tasks in the field of Archives to the Archives Office in regencies or cities in South Sumatra. In searching for information, users can do it by reading the documents needed, but the technique is less efficient to get the information needed. Machine Learning techniques are part of artificial intelligence that aims to understand or recognize the structure of data and convert the data into a mode l[3]. In recent years, several Machine Learning algorithms have begun to develop a lot and the more data, the algorithm will adjust itself to work better. Machine learning is widely researched and used to solve various problems and its algorithms are divided into three categories including supervised learning, unsupervised learning, and reinforcement learning [4].

In Information Retrieval (IR) there are methods that function to find documents based on the query needed by the user. The methods used in this research are TF IDF and BM25. There are several stages in this research, the first stage is searching for archive data in the form of text which is a source of information that is useful for classifying archive categories and proceeding to the preprocessing stage which is the second stage that must be done in ranking documents. Preprocessing is done to extract documents into a collection of terms. TF-IDF consists of 2 components, namely in this study the term t that appears in the document, for example the title in the archive, and the recurrence of the term searched in the document, divided by the number of recurrences in all documents. TF measures the frequency of occurrence of a term in a particular document. And IDF measures how important a document is. If the term you are looking for is found in the document, then the idf value will be high after that the document ranking process is continued based on the input query. Documents are sorted based on the highest function value with the Best Macth-25 (BM25) method against all documents based on the keywords and queries we are looking for [5]. In searching documents based on the entered query so that the results of each method are obtained. in this study will compare several techniques used in Information Retrieval (IR) in evaluating the archive classification system by testing the classification results obtained including the accuracy value (Precision), the truth value (Recall) and the best value (F1 Score).

Based on the problems and reasons that have been described and supported by several studies that have been conducted by Faradila Puspa Wardani (2019) with the title Query Expansion on the Information Retrieval System for Indonesian Language Journal Documents Using the BM25 Method. The results of query expansion testing on the Indonesian language journal document information retrieval system using the BM25 method, namely producing an average increase in the Precesion@K value of 0.309. The addition of the number of words in the initial query using query expansion also affects the resulting Precision@K value [6]. This research will focus on searching photo archives in the form of queries. Thus, the system built is expected to help in finding fast and accurate information.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2. METHODS

This research has stages in searching documents using TF-IDF calculation and BM25 Method. The following are the techniques used in information retrieval (IR) on Machine Learning-Based E-Archives as shown in Figure 1.

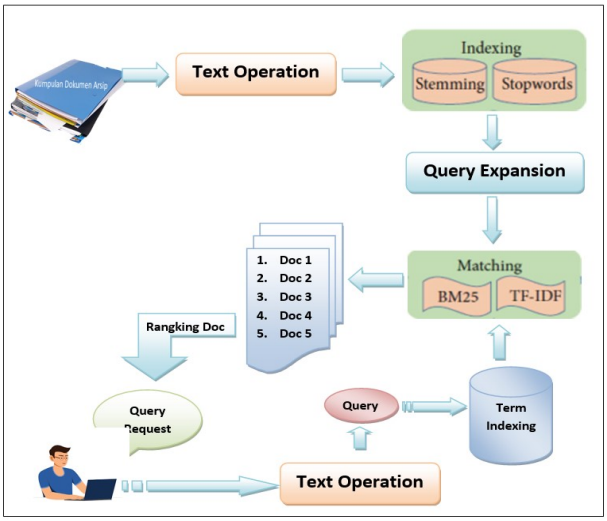


Figure 1. Techniques used in information retrieval (IR) on Machine Learning-Based E-Archives.

1. Text Operations which include the selection of words in queries and documents in transforming documents or queries into terms index (index of words).
2. Indexing, in indexing here is a preprocessing stage that is in a collection of archive documents that have been prepared in the initial stage. The preprocessing includes stemming and Stopword. The stemming process is the process of removing affixes on words that have gone through the stopword removal process. After going through the stemming process, the words have become basic words, these basic words are called a collection of terms. The resulting terms include document terms and query terms. The stopword removal process is a process that serves to eliminate words that have been obtained in the tokenizing process that do not have important meanings that are adjusted to the adjusted word dictionary [7].
3. Query Formulation which gives weight to the index of query words.
4. Matching is the process of matching several documents and queries by giving weights. The techniques used in this research are TF-IDF and BM25. The system will check the query terms contained in each document, then the

system will calculate the document weight using the TF-IDF and BM25 equations.
5. Ranking, the system will rank and search for documents that are relevant to the query and sort the documents that have a greater value based on their suitability to the query.

2.1 Methods BM25 (Binary Independen Model)

BM25 is a ranking system used to sort the results of matches (similarity) against all training documents, based on the keywords (queries) sought. BM25 method is the best method in the best match class, because this method is effective and has accuracy in sorting documents based on the query that users are looking for [8]. BM25 score is linear because it is influenced by several factors, namely: frequency of occurrence of query terms in documents or Term Frequency (TF), frequency of the number of documents or Inverse Document Frequency (IDF) and document length [9].

2.2 TF-IDF

2.2.1 Term Frequency (TF)
The most widely applied approach in local weighting is term frequency (tf). This factor expresses the number of occurrences of a word in a document. The more often a word appears in a document, the more important it is.

2.2.2 Inverse Documen Frequency (IDF)
TF-IDF (Term Frequency Inverse Document Frequency) method is a way to give weight to the relationship of a word (term) to a document. This method combines two concepts for weight calculation, namely, the frequency of occurrence of a word in a particular document and the inverse frequency of documents containing the word [10].

2.3 Machine Learning

Machine learning is a set of computer algorithms used to optimize the performance of a computer or system based on existing sample data. The main capability of machine learning is the modification and adaptation of decisions in response to changes [11]. The uses of machine learning include the following:
a. Classification is a machine learning method used to predict the value/class of an individual in a population.
b. Similarity matching is a machine learning method used to identify similarities between individuals based on existing data.
c. Clustering is a machine learning method used to group individuals in the same group based on their similarities [12].

In this research, the use of machine learning method refers to the third point.

2.4 Evaluation

Classification results can be tested using a test method where the accuracy of the system created will be measured[13]. Precision is the value of accuracy between the information requested and the answer given by the system[14], Recall The value of the system's correctness in making predictions is understanding, and F1 Score by definition, F1 Score is the harmonic mean of Precision and Recall. The best value of F1 Score is 1.0 and the worst value is 0. Representatively, if F1 Score has a good score, it indicates that our classification model has good precision and Recall. To conclude, we will calculate the Precision, Recall and F1 Score using the previous data [15].

3. RESULTS AND DISCUSSION

3.1 Implementation

This sub chapter explains the stages of application of the information retrieval system with TF-IDF and BM25 calculations. The stages carried out include Preprocessing Text preprocessing consists of several processes, namely tokenization, filtering, stopword removal, and stemming. Furthermore, the TF-IDF weight calculation process is one of the methods used to calculate the similarity between documents by means of TF-IDF weighting. After the calculation of TF-IDF, the next stage in this research is continued using the BM25 method, which is a method in the information retrieval system that is used to rank the results of the relevance of documents to the query to be searched. Ringkasan yang dihasilkan pada sistem peringkasan teks otomatis perlu diuji dan dievaluasi agar mengetahui akurasi dan kesesuaian hasil ringkasan yang dihasilkan. Pengujian tersebut dilakukan bertujuan untuk mengetahui pengaruh tingkat akurasi sistem dengan menggunakan nilai Precision, Recall, dan F1 Score.

3.2 Data Testing Process

In the testing process there are 350 documents for training, documents in the form of archives in the form of text in which there are titles and descriptions of these titles. This is done to observe the differences that occur if the number of documents used for training is different. In the data testing process in this study requires the Python programming language by importing packages which can be seen in Figure 2.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

```
!pip install lxml
import pandas as pd
import numpy as np
import nltk
import string
import re
import matplotlib.pyplot as plt
from urllib.request import urlopen
from bs4 import BeautifulSoup
from tqdm import tqdm
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
%matplotlib inline
from nltk.corpus import stopwords
nltk.download("stopwords")
!pip install Sastrawi
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
```

Figure 2. Importing packages.

```
df = pd.read_csv('data350test.csv')
df = df[:1000]
df
```

	kategori	title	description
0	jembatan	jembatan ampera	jembatan ampera dibangun pada tahun 1962 denga...
1	jembatan	lambang eksotisme kota palembang	jembatan ampera merupakan salah satu jembatan ...
2	jembatan	mengulik sejarah jembatan ampera palembang yan...	jembatan ampera palembang merupakan penghubung...
3	jembatan	ikon wong palembang di atas sungai musi	awalnya jembatan tersebut diberi nama jembatan...
4	jembatan	jembatan ampera kemegahan simbol kota palembang	keistimewaan jembatan ampera adalah dulunya pa...
...	...	...	...
345	pariwisata	Kampung Kapitan	Merupakan sebuah kawasan cagar budaya yang ter...
346	pariwisata	Taman Kambang Iwak	Taman Kambang Iwak adalah taman indah nan rind...
347	pariwisata	Wisata Pondok Bambu di Palembang	Pondok bambu dibuat pada akhir 2021 awalnya ka...
348	ibadah	Pura Kahyangan Swarna Dwipa Jakabaring	Salah satu rumah ibadah di Kompleks Jakabaring...
349	ibadah	Klenteng Hok Tjing Rio	Klenteng Hok Tjing Rio atau lebih dikenal Klen...

Figure 3. Dataset

Tabel 1. Number of query archives by category

No	Category (Indonesia)	Number of Archives
1	adat	35
2	ibadah	37
3	instansi	7
4	jembatan	12
5	kuliner	16
6	makam	15
7	musium	17
8	pariwisata	73
9	pasar	20
10	rumah sakit	8
11	sejarah palembang	79
12	sungai	5
13	transportasi	26
	Number of documents	350

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

In this research, document archives are divided into 13 categories where each category has different archives. Where the category is used as one of the queries in searching for archives. For example, the query "tourism" is inputted then the results come out all archives related to tourism in Palembang. Before doing the data preprocessing stage, the next step is to read the dataset used in this research in pyton programming, which can be seen in Figure 3. The data above is obtained from various sources. Among them are from the archives of the Archives Office and social media in the form of photos of South Sumatra in the past then from the photos are converted into digital archives in the form of Indonesian text in which there are titles and descriptions of these photos put into files with csv format.

```
len_description = []
for x in df.description:
    len_description.append(len(x))
df['description_length'] = len_description

len_title = []
for x in df.title:
    len_title.append(len(x))
df['title_length'] = len_title
df.describe(include='all')
```

	kategori	title	description	description_length	title_length
count	350	350	350	350.000000	350.000000
unique	14	325	349	NaN	NaN
top	pariwisata	Masjid Agung Palembang	benteng kuto besak berada di dekat jembatan am...	NaN	NaN
freq	79	4	2	NaN	NaN
mean	NaN	NaN	NaN	469.617143	28.442857
std	NaN	NaN	NaN	302.168916	24.112342
min	NaN	NaN	NaN	8.000000	3.000000
25%	NaN	NaN	NaN	234.250000	17.000000
50%	NaN	NaN	NaN	399.500000	24.000000
75%	NaN	NaN	NaN	657.750000	35.000000
max	NaN	NaN	NaN	1731.000000	386.000000

Figure 4. Document Length

In Figure 4 is the description length and title length of the archive. For numerical data the index consists of statistical count, mean, std, min, and max. and percentage of 25% to 75%. The most frequent number or top value is with the tourism category with a frequency of 79, in the title there is the Great Mosque of Palembang with a frequency of 4.

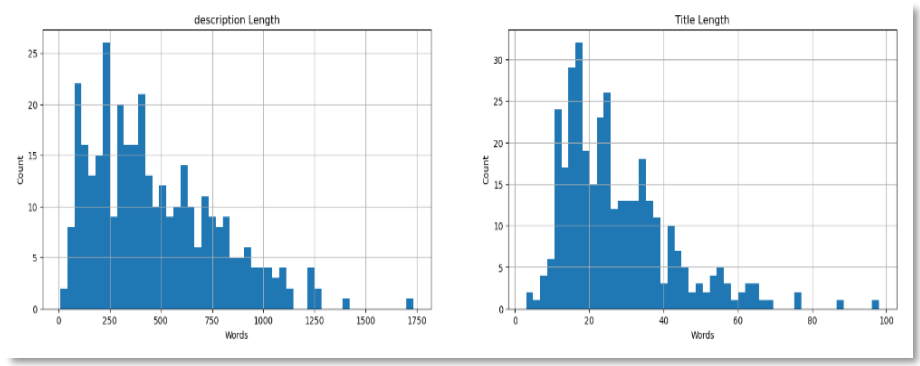


Figure 5. Visualization Chart of archive description length and title

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

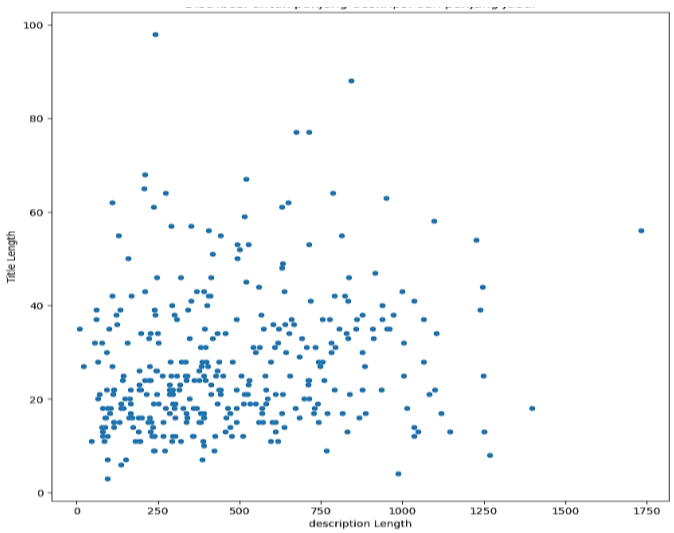


Figure 6. Length distribution of archive descriptions and titles

Figures 5 and 6 are the graphical display and distribution of the length of a description and the length of the archive title in this study with 350 documents in the form of data visualization using the Matplotlib library.

3.3 TF-IDF Value Calculation

TF-IDF is a method to calculate the weight of each word that is most commonly used in information retrieval. This method is also known to be efficient, easy and has accurate results. This method will calculate the Term Frequency (TF) and Inverse Document Frequency (IDF) values for each token (word) in each document in the corpus. This method will calculate the weight of each token t in document d with Equation 1.

$$W_{dt} = tf_{dt} * IDF_t \quad (1)$$

Where:
d = document d
t = word t of keywords
w = document weight d of the word t
tf = number of words searched for in a document
IDF = Inversed Document Frequency IDF value is obtained from
IDF = log2 (D/df)
Where:
D = total documents
Df = many documents containing the searched word
Examples of TF IDF calculation results can be seen in the table below with sample data of 5 archive docum.

Find the result of TF (Term Frequency) normalization on document 1

$$tf_{t.d} = \frac{frequency\ of\ occurrence\ of\ term\ t\ in\ document\ d}{Total\ terms\ in\ document\ d}$$
$$tf_{t.d} = \frac{1}{3} = \underline{\mathbf{0{,}333}}$$

Find the result of IDF (Inverse Document Frequency) on document 1

$$idf_t = \frac{1}{dft}$$
$$idf_t = \frac{1}{dft}$$
$$idf_t = log\ (\frac{n}{dft})$$
$$idf_t = log\ (\frac{5}{4}) = \underline{\mathbf{0{,}097}}$$

df = document frequency term t

N = number of documents in the dataset

Search TF-IDF result on document 1

$$TfIdf = TF * IDF$$
$$TfIdf = 0{,}333 * 0{,}097 = \underline{\mathbf{0.032}}$$

The calculation results as shown in Figure 7.

TERM	TF (Term Frequency)					TF Normalisasi (Term Occurrence "t" divided by document length "d")					DF (number of documents that contain term)	IDF(n/df)	TF-IDF (TF*IDF)				
	d1	d2	d3	d4	d5	d1	d2	d3	d4	d5			d1	d2	d3	d4	d5
ampera	1	1	1	1	0	0,333	0,333	0,333	0,143	0	4	0,097	0,032	0,03	0,03	0,01	0
jaman	1	0	0	0	0	0,333	0	0	0	0	1	0,699	0,233	0	0	0	0
dulu	1	0	0	0	0	0,333	0	0	0	0	1	0,699	0,233	0	0	0	0
jembatan	0	1	1	2	0	0	0,333	0,333	0,286	0	3	0,222	0	0,07	0,07	0,06	0
1970	0	1	0	0	0	0	0,333	0	0	0	1	0,699	0	0,23	0	0	0
1971	0	0	1	0	0	0	0	0,333	0	0	1	0,699	0	0	0,23	0	0
linimasa	0	0	0	1	0	0	0	0	0,143	0	1	0,699	0	0	0	0,1	0
sejarah	0	0	0	1	0	0	0	0	0,143	0	1	0,699	0	0	0	0,1	0
wong	0	0	0	1	0	0	0	0	0,143	0	1	0,699	0	0	0	0,1	0
kito	0	0	0	1	0	0	0	0	0,143	0	1	0,699	0	0	0	0,1	0
kota	0	0	0	0	1	0	0	0	0	0,333	1	0,699	0	0	0	0	0,23
palembang	0	0	0	0	1	0	0	0	0	0,333	1	0,699	0	0	0	0	0,23
2019	0	0	0	0	1	0	0	0	0	0,333	1	0,699	0	0	0	0	0,23
Document Length	3	3	3	7	3												

Figure 7. TF-IDF calculation of sample 5 documents

After testing, the results obtained from the trial can be seen in Figure 8 and Figure 9 is the TF-IDF formula entered into Python Programming with 350 data.

p-ISSN: 2656-5935 http://journal-isi.org/index.php/isi e-ISSN: 2656-4882

```python
tfidf_vectorizer = TfidfVectorizer()
tfidf_documents_vectorized = tfidf_vectorizer.fit_transform(df_list)
tfidf_vocabulary = tfidf_vectorizer.get_feature_names_out()
tfidf_dataframe = pd.DataFrame(tfidf_documents_vectorized.toarray(), columns=tfidf_vocabulary)
```

```python
def TF_IDF_df(df):
    dfs = (df > 0).sum(axis=0)
    N = df.shape[0]
    idfs = np.log(N/dfs)
    TF_IDF_score = np.array(idfs*df)
    return pd.DataFrame(TF_IDF_score, columns=tfidf_vocabulary)

#TF/IDF model
tf_idf_df = TF_IDF_df(tfidf_dataframe)
display(tf_idf_df[:5])
```

Figure 8. TF-IDF formula in Python Programming

ampera	amperadulunya	amperanya	ampuh	amtenar	amur	an	anak	anaknya	anang	and	anda	andal	andalan	andalas	aneka	anggap
0.389168	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.591781	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.649631	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.360482	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.722325	0.605746	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.436834	0.00000
0.400933	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.464063	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.474513	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.325157	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.264598	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00000
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.63869

Figure 9. TF-IDF value display of 350 documents

Figure 9 is the result of calculating the weight of each word using the TF-IDF method. This method will calculate the Term Frequency (TF) and Inverse Document Frequency (IDF) values on each token (word) in each document in the corpus.

3.4 BM25 Value Calculation

BM25 is a ranking system used to sort the results of matches (similarity) against all training documents, based on the keywords (queries) sought. The BM25 method is the best method in the best match class, because this method is effective and has accuracy in sorting documents based on the query that users are looking for. The BM25 calculation formula can be seen in Equation (2).

$$BM25=\sum_{i=1}^{n} IDF(qi).\frac{f(qi.D).(k1+1)}{f(qi.D)+k1.\left((1\text{-}b)+b.\frac{|D|}{avgdl}\right)} \tag{2}$$

Description:
(qi, D) = Number of term frequencies that appear in document D
$|D|$ = Number of sentences in the document D
$avgdl$ = Average document length in the collection or corpus
k1 = 1.2
b = 0.75

With the inverse document frequency equation shown in Equation (3).

$$IDF(qi) = log_{10}\left(\frac{N - df(qi) + 0.5}{df(qi) + 0.5}\right) \quad (2)$$

Description:
(qi) = Number of documents containing term q
N = Number of documents in the collection

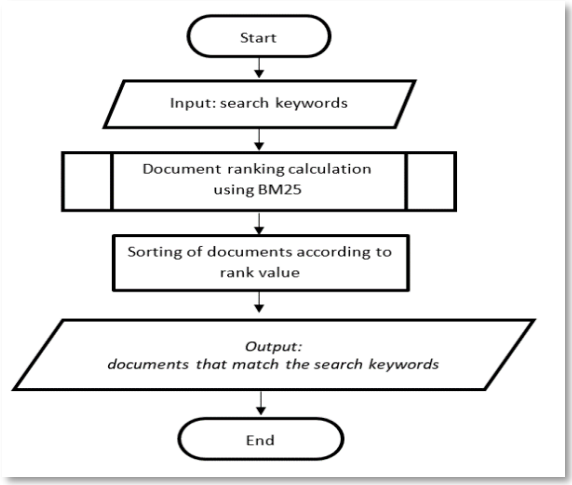


Figure 10. Archive Data Search Flowchart

In searching archive documents, you must input search keywords, then the input will be processed in three processes as follows:

1. The process of searching documents that have search keywords using pyton programming. This process produces a collection of documents that contain the search keyword.
2. The next process is to provide a ranking value for the set of documents obtained in the first process. The ranking value for each document is based on the document's relevance to the search keywords. Documents that are more relevant to the search keywords have a greater ranking value than those that are less relevant.
3. The next process is to sort the documents based on the ranking values obtained in the second process. Sorting is done by sorting ascending (sorting from high

to low value rank). The largest rank means that the document is most suitable for the keywords searched. Documents displayed as search results are documents that have been sorted based on the ranking of document suitability with search keywords.

After testing, the results obtained from the trial can be seen in Figure 11, which is the BM25 formula entered into the Python Programming.

```
def BM25_IDF_df(df, k, b):

    dfs = (df > 0).sum(axis=0)
    N = df.shape[0]
    idfs = -np.log(dfs / N)

    k_1 = k
    b = b
    dls = df.sum(axis=1)
    avgdl = np.mean(dls)

    numerator = np.array((k_1 + 1) * df)
    denominator = np.array(k_1 *((1 - b) + b * (dls / avgdl))).reshape(N,1) + np.array(df)

    BM25_tf = numerator / denominator

    idfs = np.array(idfs)

    BM25_score = BM25_tf * idfs
    return pd.DataFrame(BM25_score, columns=vocabulary)
```

```
#Default model
bm25_df = BM25_IDF_df(dataframe, k=1.2, b=0.75)

# a dataframe with BM25-idf weights
display(bm25_df[:10])

#Specific model
bm25_df_specific = BM25_IDF_df(dataframe, k=1, b=0.5)

# a dataframe with BM25-idf weights
display(bm25_df_specific[:10])
```

Figure 11. BM25 formula in Python Programming

ampera	amperadulunya	amperanya	ampuh	amtenar	amur	anak	anaknya	anang	anda	andal	andalan	andalas	aneka	anggap	anggaran	a
3.224827	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
3.596164	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
3.842749	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
3.226519	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
3.883600	4.219116	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	3.427852	0.0	0.0	
3.329816	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
3.561732	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.002603	0.0	0.0	0.0	0.000000	0.0	0.0	
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	
0.000000	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.0	0.0	

Figure 12. Display of BM25 value results from 350 documents

Figure 12 is the result of calculating the weight of each word using the BM25 method. BM25 score is linear because it is influenced by several factors, namely: frequency of occurrence of query terms in documents or Term Frequency (TF); frequency of the number of documents or Inverse Document Frequency (IDF); and document length.

Journal of Information Systems and Informatics
Vol. 5, No. 4, December 2023
p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

3.5 BM-25 Method Document Ranking

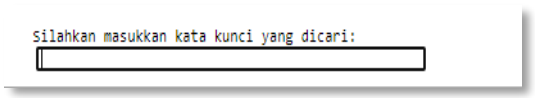
Based on the results obtained, the user enters the query as a testing document, then preprocessing and continued ranking documents with BM25. Further results are sorted based on documents that have the highest match value. Testing in this study was carried out between queries from users and training data that had been processed previously. The training process uses 350 data archives. Figure 13 is the code in ranking documents in Python Programming.

```
df5=df.description
#Contoh imput query
inp = input("\nSilahkan masukkan kata kunci: \n")
query2 = inp

q_terms = query2.split()
q_terms

#def retrieve_ranking(query, dfa):
doc_scores = retrieve_ranking(query2,bm25_df)
print('NILAI RANKING BM25')
print(doc_scores)
print('\n')
print('NILAI RANKING BM25 dengan id 10 terbesar')
print(*doc_scores[:10], sep = "\n")
print('\n')
res_list = [x[0] for x in doc_scores[:10]]
print('NILAI RANKING BM25 dengan id 10 termirip')
print(res_list)
print('\n')
print('NILAI RANKING BM25 dengan Title 10 termirip')
print(df1[res_list])

print('NILAI RANKING BM25 (description) 10 termirip')
print(df5[res_list])
```



Gambar 13. Display to search for a document

Figure 13 is the display of the system to search for a document that the user wants. When the user enters a query, the results will come out where the results that appear are the BM25 ranking value of all documents. This ranking includes the BM25 ranking value of ID, Title and description as shown in Figure 14 and 15.

Figure. 14 BM25 Rank Value Output with 10 largest ID and descriptions

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

NO	ID DOC	NILAI BM25	TITLE	DESKRIPSI
1	108	4.960414771739761	Jembatan Kertapati Palembang 14 Agustus 1948	Sebelum adanya Jembatan Ampera ditahun 1960an Jembatan Ogan adalah satu-satunya jembatan yang menghubungkan wilayah Palembang dengan daerah Uluan Pada masa penjajahan orang Belanda di Palembang menyebut jembatan itu dengan sebutan Ogan Brug atau Jembatan Ogan karena lokasi pembuatan Jembatan ini melintas di atas Sungai Ogan Kemudian penyebutan nama Jembatan Ogan ini berubah menjadi Wilhelmina Brug atau Jembatan Wilhelmina yang merujuk nama Ratu Belanda pada saat itu yang berkuasa di negeri Belanda Jembatan Ogan saat ini dikenal dengan Jembatan Kertapati
2	256	4.820876909185179	Tiga Jembatan megah di Palembang	View Tiga Jembatan megah di Palembang dengan design yg berbeda dalam Satu Frame Sungai musi memiliki Jembatan dari Hulu sampai Hilir sekitar 43 jembatan gantung 19 buah Jembatan besi baja (kendaraan) 1 buah jembatan kereta api (di Kota Tebing Tinggi)
3	3	4.700282354928654	ikon wong palembang di atas sungai musi	awalnya jembatan tersebut diberi nama jembatan bung karno sebagai ungkapan terima kasih kepada presiden soekarno yang telah mendukung dan berjuang untuk mewujudkan pembangunan jembatan yang menghubungkan palembang ilir dengan palembang ulu
4	93	4.5967878781009945	Jembatan Ampera dan Sungai Musi	Tak akan lengkap rasanya jika datang ke Palembang tetapi tidak menyempatkan diri ke Jembatan Ampera Ada kemegahan jembatan dan juga luasnya sungai musi yang begitu ikonik Jembatan yang bersejarah ini ada di atas sungai musi dan merupakan singkatan dari Amanat Penderitaan Rakyat Jembatan yang sudah dibangun di tahun 1962 Jembatan ini ada di Jalan Lintas Timur Palembang Jembatan yang menghubungkan Seberang Ulu dan juga Seberang ilir yang terpisah karena Sungai Musi Sebaiknya berkunjung di malam hari karena membuat para wisatawan bisa menikmati keindahannya
5	1	4.56248447216035	lambang eksotisme kota palembang	jembatan ampera merupakan salah satu jembatan bersejarah di indonesia yang terletak di kota palembang sumatera selatan jembatan megah yang memiliki panjang 1117 meter ini membentang membelah garis keindahan perairan sungai musi menghubungkan wilayah seberang ulu dan ilir kota palembang jembatan ampera mulai dibangun pada masa kepemimpinan presiden soekarno tepatnya pada tahun 1962 dan diresmikan pada tanggal 10 november 1965 bersamaan dengan peringatan hari pahlawan jembatan ampera sempat dinamakan dengan jembatan soekarno hal ini sebagai simbol ungkapan terima kasih masyakarat provinsi sumatera selatan kepada presiden soekarno atas peranan dan dedikasi beliau yang telah merealisasikan cita-cita masyarakat sumatera selatan khususnya palembang namun kemudian karena persoalan politik yang terjadi di tanah air pada tahun 1966 nama jembatan seokarno secara resmi diubah menjadi jembatan ampera (amanat penderitaan rakyat) sebagai sebuah simbol kemerdekaan dari amanat penderitaan rakyat palembang
6	2	4.561650363525118	mengulik sejarah jembatan ampera palembang yang bakal dipasang lift	jembatan ampera palembang merupakan penghubung wilayah hulu dan hilir jembatan ini sudah ada sejak puluhan tahun lalu dan menjadi ikon kota pempek jembatan ampera beberapa kali mendapat revitalisasi dan terbaru bakal dipasang lift atau tangga otomatis untuk membawa pengunjung ke menara jembatan dengan ketinggian 50 meter namun tak banyak yang tahu jika sebelum bernama ampera atau amanat penderitaan rakyat jembatan ini pernah diberi nama jembatan soekarno sebagai bentuk penghormatan kepada presiden indonesia pertama
7	8	4.482035231398483	sejarah jembatan ampera palembang	jembatan amanat penderitaan ini adalah jembatan yang hanya ada di palembang situs jembatan ini memiliki panjang 1177 meter tinggi 63 meter dan lebar 22 meter di atasnya terdapat beberapa buah menara yang jaraknya 75 meter antar menara ide pembangunan jembatan yang kini menjadi tempat wisata palembang ini pertama kali dicetuskan sekitar tahun 1906 lalu tujuannya adalah untuk menghubungkan dua wilayah daratan kemudian pada tahun 1924 gagasan tersebut muncul kembali sayangnya usulan tersebut belum juga terealisasi hingga kemudian setelah kemerdekaan tepatnya tahun 1956 usulan pembuatan jembatan muncul kembali dengan hanya bermodalkan dana rp3000000 saja pemerintah setempat mulai membangun jembatan ini melalui tangan tangan panitia pembangunan jembatan ini dibangun pada tahun 1957 proses pembangunan jembatan ini berlangsung kurang lebih selama 3 tahun dan mendapat dukungan dari tenaga ahli dari jepang
8	5	4.424250902396177	infrastruktur ikonik kota palembang	berlibur ke palembang tidak afdol rasanya jika tidak mengunjungi jembatan ampera ya jembatan yang menjadi ikon dari ibukota sumatera selatan ini berada di jalur lintas sumatera yang membentang diatas sungai musi jembatan ampera dibangun sebagai penghubung dua kawasan yakni seberang ilir dan seberang ulu tak ayal jembatan ini menjadi akses primadona warga dalam beraktifitas hal ini bisa dibuktikan dengan pemandangan jembatan ampera yang tidak pernah sepi dilintasi kendaraan selain berfungsi sebagai jalur utama jembatan ampera juga menjadi destinasi wajib bagi wisatawan yang berkunjung ke kota berjuluk bumi sriwijaya tersebut lho disana sobat trac bisa melakukan berbagai aktivitas seru seperti berswa fotomencicipi aneka kulinerhingga menikmati keindahannya ketika malam ampera merupakan akronim dari amanat penderitaan rakyat yang tersemat sebagai nama pada jembatan ini namunsebelum dikenal sebagai jembatan amperadulunya jembatan ini punya nama lain
9	7	4.2672627513612795	menakutkan kisah misteri jembatan ampera bikin merinding	jembatan yang menjadi salah satu ikon kota palembang ini ternyata menyimpan banyak cerita misterius yang telah diturunkan sejak lama seperti keberadaan makhluk gaib dan tempat orang bunuh diri kisah misterius jembatan ampera ini sebenarnya sudah banyak diketahui masyarakat setempat namun bagi anda yang berada di luar kota palembang mungkin cerita misteri jembatan ampera ini jarang diketahui banyak orang
10	4	4.241073907116916	jembatan ampera kemegahan simbol kota palembang	keistimewaan jembatan ampera adalah dulunya pada agian tengah jembatan ampera bisa di angkat yang dimaksudkan agar kapal kapal yang melewati sungai musi tidak terbentur dengan badan jembatan pemberat masing masing bandul mencapai 500 ton dengan kecepatan pengangkatan bandul 10 meter per menit pada saat bagian atas jembatan dapat diangkat kapal dengan ukuran lebar 60 meter dan dengan tinggi 4450 centimeter dapat melalui sungai musi kecepatan membuka jembatan sekitar 10 meter per menit dan dibutuhkan waktu sekitar 30 menit untuk membuka jembatan secara penuh kini jembatan ampera sudah tidak dibuka kembali selain sudah tidak dilintasi perahu besar waktu yang lama untuk membuka jembatan akan menganggu arus lalu lintas yang ada di atasnya saat fungsi terbuka jembatan tidak digunakan lagi maka bandul seberat 500 ton yang ada di kedua menara jembatan diturunkan hal tersebut dilakukan demi pertimbangan keamanan

Figure 15. BM25 Rank Value with 10 largest IDs and descriptions (Indonesia)

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Figure 15 is the search results displayed in order based on the results of the BM25 ranking function with the keyword "ampera bridge" from 350 archived documents. The results obtained with the keywords above for the top order are sequence id 24 with the title Jembatan Kertapati Palembang 14 Agustus 1948 with a BM25 value of 4.960414771739761. This system is set to display a minimum of 10 relevant documents.

3.3.4 Evaluation

After the archive document search is completed, an evaluation of this research is carried out. Classification results can be tested using a test method where the accuracy level of the system created will be measured. In Table 4 is the results of Precision, Recall and F1 score as many as 350 archive documents in Python Programming using the keyword " ampera bridge ".

Table. 4 Precision, Recal and F1 Score Results

Model	Precision	Recall	F1 Score
TF/IDF	0.33	0.2	0.250000
BM25	0.75	0.6	0.666667
BM25_Specific	0.50	0.4	0.444444

Table 4 is the result of testing with the keyword "ampera bridge" where the keyword is used to search for archives of 350 documents, so that the results obtained are archived documents that are relevant to the keywords searched. The results tested on the BM25 method obtained an average result of Precision 0.75, Recall 0.6 and F1 Score 0.666665 while on TF-IDF obtained an average result of Precision 0.33, Recall 0.2 and F1 Score 0.250000.

The testing graph of Precision, Recall and F1 Score values can be seen in Figure 16 shows that the Precision, Recall and F1 Score values with these two testing methods, the BM25 method for the archive search system is included in the good category.

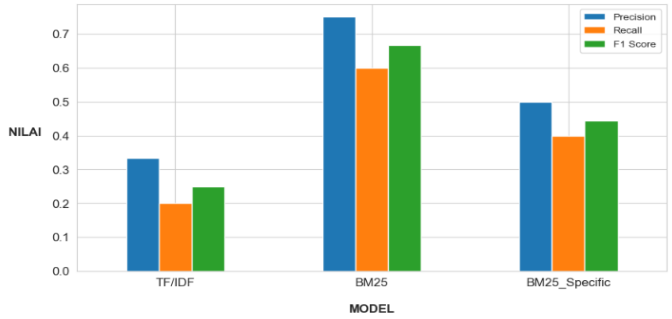


Figure 165. Graph Visualization of Precision, Recall and F1 Score Values

4. CONCLUSION

After a thorough process encompassing data collection, implementation, and testing, the study yields several key conclusions. The BM25 algorithm, as proposed in this research, proves effective for archive searching. This efficacy extends from the initial data pre-processing stage through to document ranking, effectively determining the similarity between documents and queries. The algorithm's performance was rigorously evaluated using the metrics of Precision, Recall, and the F1 Score, specifically with the keyword 'ampera bridge' across a dataset of 350 documents.

The test results for various keywords indicate that the BM25 method performs well, achieving a Precision of 0.500, Recall of 0.4, and an F1 Score of 0.444. In comparison, the TF-IDF method yielded lower scores, with a Precision of 0.333, Recall of 0.2, and an F1 Score of 0.25000. These results place the BM25 method in a favorable position for archive searching, demonstrating its reliability and efficiency. It's important to note that the optimal F1 Score is 1.0, representing perfect precision and recall, while the lowest possible score is 0, indicating no accuracy in the search results. Based on these evaluations, the BM25 method clearly stands out as a robust and effective tool for enhancing archive search systems.

REFERENCES

[1] R. R. Baihaqi, "Temu Kembali Informasi pada Berita Olahraga Berbahasa Indonesia dengan Metode BM25 dan Seleksi Fitur Term Frequency (TF)," J. Pengemb. *Teknol. Inf. dan Ilmu Komput.*, vol. 4, no. 11, pp. 4200–4206, 2020.

[2] J. Sistem, A. Cucus, Y. Aprilinda, I. Sistem, and I. Presensi, "768-1474-1-Sm," 2018.

[3] A. Roihan, P. A. Sunarya, and A. S. Rafika, "Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper," *IJCIT (Indonesian J. Comput. Inf. Technol.*, vol. 5, no. 1, pp. 75–82, 2020, doi: 10.31294/ijcit.v5i1.7951.

[4] M. Ula, A. Faridhatul Ulva, and Mauliza, "Implementasi Machine Learning Dengan Model Case Based Reasoning Dalam Mendagnosa Gizi Buruk Pada Anak," *J. Inform. Kaputama*, vol. 5, no. 2, pp. 333–339, 2021.

[5] A. I. Kadhim, "Term Weighting for Feature Extraction on Twitter: A Comparison between BM25 and TF-IDF," *2019 Int. Conf. Adv. Sci. Eng. ICOASE 2019*, pp. 124–128, 2019, doi: 10.1109/ICOASE.2019.8723825.

[6] "Faradila Puspa Wardani (1).pdf." 2018.

[7] W. Faradila Puspa, "Query Expansion Pada Sistem Temu Kembali Informasi Dokumen Jurnal Berbahasa Indonesia Menggunakan Metode BM25," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 3, pp. 2619–2625, 2019.

[8] A. I. B. Pranata and M. Indriati, "Klasifikasi Dokumen pada Laporan Kepolisian dengan Menggunakan Metode BM25 dan Improved K-Nearest Neighbor (IKNN)," *Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 5, pp. 4434–4438, 2019.

[9] B. Herwijayanti, D. E. Ratnawati, and L. Muflikhah, "Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity," Pengemb. *Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 1, pp. 306–312, 2018.

[10] R. R. A. Siregar, F. A. Sinaga, and R. Arianto, "Aplikasi Penentuan Dosen Penguji Skripsi Menggunakan Metode TF-IDF dan Vector Space Model," Comput. *J. Comput. Sci. Inf. Syst.*, vol. 1, no. 2, p. 171, 2017, doi: 10.24912/computatio.v1i2.1014.

[11] H. K. Pambudi, P. G. A. Kusuma, F. Yulianti, and K. A. Julian, "Prediksi Status Pengiriman Barang Menggunakan Metode Machine Learning," *J. Ilm. Teknol. Infomasi Terap.*, vol. 6, no. 2, pp. 100–109, 2020, doi: 10.33197/jitter.vol6.iss2.2020.396.

[12] N. L. P. C. Savitri, R. A. Rahman, R. Venyutzky, and N. A. Rakhmawati, "Analisis Klasifikasi Sentimen Terhadap Sekolah Daring pada Twitter Menggunakan Supervised Machine Learning," *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 1, pp. 47–58, 2021, doi: 10.28932/jutisi.v7i1.3216.

[13] R. Sistem and E. J. Evaluasi, "JURNAL RESTI Klasifikasi Citra Burung Lovebird Menggunakan Decision Tree dengan," *J. Resti*, vol. 5, no. 10, pp. 688–696, 2021.

[14] M. Martin and L. Nilawati, "Recall dan Precision Pada Sistem Temu Kembali Informasi Online Public Access Catalogue (OPAC) di Perpustakaan," *Paradig. - J. Komput. dan Inform.*, vol. 21, no. 1, pp. 77–84, 2019, doi: 10.31294/p.v21i1.5064.

[15] C. H. Yutika, A. Adiwijaya, and S. Al Faraby, "Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes," *J. Media Inform. Budidarma*, vol. 5, no. 2, p. 422, 2021, doi: 10.30865/mib.v5i2.2845.