ISSN 2460-9056
http://socj.telkomuniversity.ac.id/ijoict/

Intl. Journal on ICT
Vol. 8, No. 2, Dec 2022. pp. 53-65
doi: 10.21108/ijoict.v8i2.681

Comparison of Term Weighting Methods in Sentiment Analysis of the New State Capital of Indonesia with the SVM Method

Muhammad Kiko Aulia Reiki [1*], Yuliant Sibaroni [2], Erwin Budi Setiawan [3]

[123]*School of Computing, Telkom University*
Jl. Telekomunikasi No. 1 Terusan Buahbatu, Bandung, Jawa Barat, Indonesia 40257

[1]muhammadkiko@students.telkomuniversity.ac.id, [2]yuliant@telkomuniversity.ac.id,
[3]erwinbudisetiawan@telkomuniversity.ac.id,

Abstract

The relocation of the State Capital to "Nusantara", which was inaugurated with the enactment of UU No. 3 of 2022, is a significant project that has sparked polemics among Indonesian citizens. Many people expressed their opinions and thoughts regarding the relocation of the State Capital on Twitter. This tendency of public opinion needs to be identified with sentiment analysis. In sentiment analysis, term weighting is an essential component to obtain optimal accuracy. Various people are trying to modify the existing term weighting to increase the performance and accuracy of the model. One of them is *icf-based* or *tf-bin.icf*, which combines inverse category frequency (ICF) and relevance frequency (RF). This study compares the *tf-idf*, *tf-rf*, and *tf-bin.icf* term weighting with the SVM classification method on the new State Capital of Indonesia topic. The *tf-idf* weighting results are still the best compared to the *tf-bin.icf* and *tf-rf* term weights, with an accuracy score of 88.0%, a 1,3% difference with *tf-bin.icf* term weighting.

Keywords: Indonesian state capital, Nusantara, Twitter, SVM, term weighting, sentiment

I. INTRODUCTION

The use of social media nowadays is widespread. Social media has become the primary means for human beings to access all forms of communication [1]. Social media is a new medium for us to express opinions and share unique perspectives and thoughts on various trending issues and topics. One of the popular social media for communicating using text is Twitter. Twitter is a micro-blogging type of social media with a maximum of 280 characters. With this, Twitter can create a public space where people can gather, discuss, and demand accountability, leading to positive social change [2]. One of the hot topics is the relocation of the Indonesian State Capital, the "Nusantara".

The discourse on relocating the capital seems to be a routine agenda in each presidential period. However, under the government of Jokowi - Ma'ruf, it was finally realized with the enactment of Law No. 3 of 2022 concerning the State Capital [3]. This then became a polemic amid Indonesian society. Many people expressed their opinions and views regarding the transfer of the Indonesian State Capital on social media Twitter. Simple access to Twitter in the delivery of opinions could provide an opportunity for the assessment and evaluation of the new Indonesian state capital. To generate information from existing opinion data, sentiment analysis was done to separate opinions into positive or negative sentiment classes and determines which factors are frequently discussed in those opinions [4].

Received on 09 Dec 2022. Revised on 26 Dec 2022. Accepted and Published on 31 Dec 2022.

Sentiment analysis, commonly called opinion mining, is a field of study that analyses opinions, judgments, and emotions about services, products, and even issues or topics [5]. The opinion itself has become essential in making a decision [6]. Sentiment analysis can be done using various methods, including Naïve Bayes, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and so on. One of the studies by Arsi P. et al. [7] performing sentiment analysis using the SVM method and the TFIDF feature weighting resulted in an accuracy of 96.68%. The primary stages in conducting sentiment analysis include data collection, data labeling, preprocessing, adding weighting features, classification, and then conducting analysis.

Term weighting aims to give weight to each word in a document or sentence. The relevance of any terms in these models changes depending on the text. As a result, adding a weight (value) to each phrase is critical for representing text documents [8]. Some feature or term weights often used are *tf*, *tf-idf*, *tf-rf*, etc. Not a few parties are trying to modify the existing weighting features to increase the performance and accuracy of the model. Research by Deqing Wang et al. [9] introduces inverse category frequency into the term weighting scheme. It proposes a weighting scheme based on *tf-icf* and *icf-based* or *tf-bin.icf*, which combines inverse category frequency (ICF) and relevance frequency (RF). The results reveal that the *tf-bin.icf* weighting scheme performs better than the other seven weighting schemes on binary classification task.

Departing from this background, this study will compare term weighting, including *tf-idf*, *tf-rf*, and *tf-bin.icf*. The New State Capital of Indonesia was chosen in this sentiment analysis because it is a trending topic and is widely discussed by the public with various opinions. One of the studies by Zhiejie Liu [10] proved that the SVM classification method is superior to other methods, especially in political fields or topics, with an accuracy of 97.31% and an F1 score of 93.78%. The study implemented the SVM classification with linear kernel and K-fold cross-validation, along with the Confusion Matrix evaluation method for measuring accuracy. SVM's Linear Kernel is often used when the data is linearly separable[11] which is, very suitable for this binary classification task.

In this article, related literature studies will be discussed as well as studies that are the reference for research in Chapter 2. Then the system built in this research can be seen in Chapter 3. The results of the research that has been carried out are listed in Chapter 4. As well as the conclusions from the research can be see Chapter 5.

II. LITERATURE REVIEW

The Twitter social media platform is the most popular choice for text classification due to the relatively large number of characters allowed (280) and its use globally to express opinions on certain issues [12]. Opinions produced by users are called tweets, with emotions from each author. This emotion needs to be analyzed further to get a good understanding of the opinions expressed.

Sentiment analysis is a field that analyses the sequence of words conveyed to understand better the emotions of the opinions expressed [13]. Opinion is an essential part of social life. Whenever humans need to make a decision, humans tend to know the opinion of others first. In the real world, companies and associations are always looking for customer or public feedback on their products and services.[5].

Research related to sentiment analysis regarding the relocation of the state capital has been carried out using various classification methods. Amar [14] researched sentiment analysis on the decision to move the Indonesian state capital using the Naïve Bayes classification method. The stages of the research activity consist of data collection, feature selection with *tf*, and then classification. This research produces an average accuracy value of 89.86%. However, this research was conducted before the decision on the new state capital of Indonesia.

Meanwhile, Arsi P et al. [7] also performed a sentiment analysis on moving the capital city of Indonesia using the Support Vector Machine classification method. Feature weighting is also done using *tf-idf*. The test results reveal that the SVM method has an accuracy of 96.68% with the k-fold cross-validation method. In this study, a sentiment with negative words cannot be optimally determined.

Domeniconi et al. [15] researched to compare the weighting of *tf-idf* and other supervised variants of *tf-idf* on text classification. The results show a tight rivalry between *tf.rf-based* and *tf.idfec-based* methods: the top results obtained with various datasets and algorithms, with variable amounts of feature selection, are fairly comparable, although with minor variances. When there are a lot of features, *tf-rf* appears to be more stable. In addition to the weighting of these features, Wang D. et al. [9] introduce *tf-icf* and *icf-based* or *tf-bin.icf* into the feature weighting scheme. The research was also carried out using the SVM method. As a result, *tf-bin.icf* is superior in the binary classification task, with an F1 score reaching 73.6%.

Based on related research, the SVM classification method can be relied upon for sentiment analysis or text classification. Especially on political topics, as explained by Zhiejie Liu [10] in his research. So, this study uses the SVM classification method, which is used as a model to compare feature weighting, which can increase the model's accuracy, namely *tf-idf*, *tf-rf*, and *tf-bin.icf* in text classification with two categories, either it is positive or negative. A comparison of these feature weighting has never been done before.

III. RESEARCH METHOD

The classification model developed in this study aims to compare various term weighting variations, namely *tf-idf*, *tf-rf*, and *tf-bin.icf*. While the classification method used is SVM. In general, the system design to be built is shown in Figure 1.

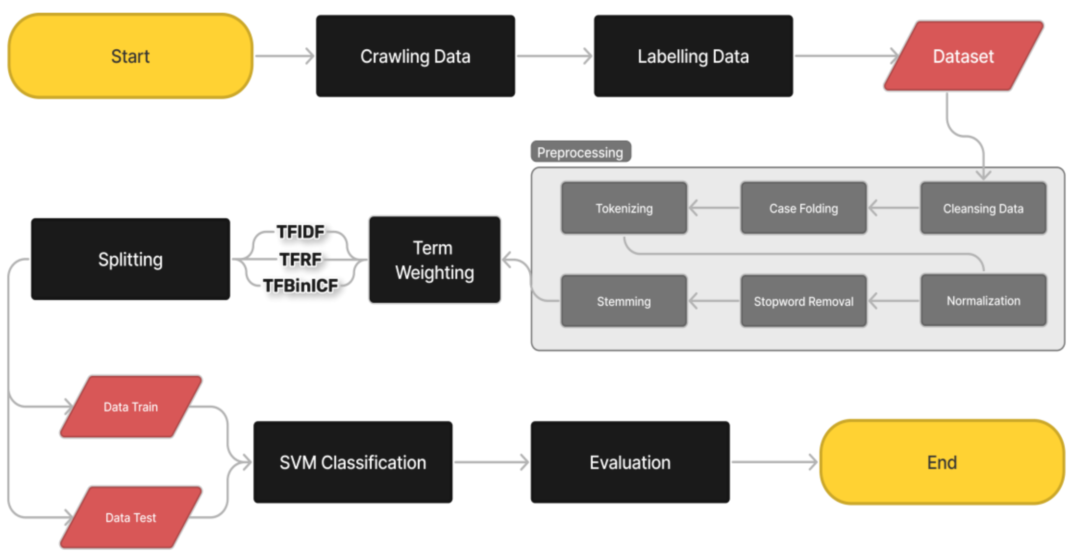

Fig 1. Flowchart of the built system

A. Crawling Data

In this study, the dataset is taken from Twitter from 1 October 2021 to 28 February 2022 using the keywords "ibu kota negara" and "ikn". The process of crawling the data is done using a library in the python programming language, namely Social Network Scrap or *snscrape*. This crawling data produces a dataset in Comma Separated Value (CSV) format, which contains tweets regarding the opinions of the Indonesian people about the decision of the new state capital.

B. Labelling Data

The results of the crawling data that have been collected then go through the manual labeling stage where the data is divided into 2 classes/labels namely positive and negative. Labeling is carried out by 3 people with the aim of reducing subjectivity in labeling. Labeling is done by paying attention to the words contained in the tweets data that has been taken, if in the tweets data there are harsh words or words that are inappropriate to eat they will be labeled negative or 0, if the data does not contain harsh words or has words which have a positive meaning, then it is given a positive label or 1.

The word harsh has the meaning of swearing, insulting, slurring, ridiculing, and others. While sentences that are inappropriate are sentences that contain provocations that have bad intuition about an issue or negative sentences that can make things worse. The example of labeliing data process can be seen in Table I.

TABLE I
LABELLING DATA

Tweet	**Label**
Pembangunan IKN berdasarkan UU yang berlaku. Bernama NUSANTARA, Ibu Kota Negara mengusung Visi "Kota Dunia Untuk Semua" #IndonesiaMaju	Positive
PKB Dukung Nusantara nama ibu kota baru IKN Sejahterakan Bangsa	Positive
Jangan sampai ibu kota baru jadi ibu tiri yang kerjanya menyengsarakan rakyat dengan hutang APBN yang semakin menggunung #UUIKNMendzalimiRakyat	Negative

C. Preprocessing

When the data has been collected and labeled, the data then enters the next stage which is preprocessing. Preprocessing is done to clean data from things that are not needed and to facilitate the processing of data that will be used for the classification process. In this study, the dataset that has been collected goes through these several steps.

1) *Cleansing Data*
Cleansing Data is a process of cleaning or removing things, such as punctuation, numbers, URLs, and words that are considered not important. An example of data that will be deleted through the cleansing process is shown seen in Table II. Most of the cleansing was done with the *regex* library on python.

TABLE II
CLEANSING DATA

<table>
<tr><th>Cleansing</th></tr>
<tr><td>(.) , (,) , (?) , (!) , (;) , (:) , (-) , (--) , (=), (‘..’) , (“..”) , (/) , ((..)) , ([..]) , (`) , (~), (@) , (#) , ($) , (^) , (&) , (*) , (_) , (+), ({..}) . (|) , (>) , (<) , (1,2,3,4,5,6,7,8,9)</td></tr>
</table>

2) *Case Folding*
In this process, words that have uppercase are changed to lowercase. This process is done to avoid duplication, which is differentiated from uppercase/lowercase (case-sensitive).

3) *Tokenizing*
Tokenization is dividing or breaking a sentence previously separated by a space into the words that compose it. This tokenization is an essential process for classification.

4) *Normalization*
At the normalization stage, abbreviated words, wrong words in writing, and informal words are changed into standard terms according to KBBI writing. The dictionary used is a KBBI word dictionary that has been manually made. An example of normalization can be seen in Table III.

TABLE III
NORMALIZATION

Before	After
[proyek, ikn, **yg**, **akaan** ,membuka, peluang, besar, para, oligarki, **utk**, bermain, curang, securangnya, tolak, uu, ikn]	[proyek, ikn, **yang**, **akan**, membuka, peluang, besar, para, oligarki, **untuk**, bermain, curang, securangnya, tolak, uu, ikn]

5) *Stop Word Removal*
Stop Word Removal is the stage of removing conjunctions that often appear. These words usually have a function but do not have a meaning and do not give weight to an opinion or sentence. An example of stop word can be seen in Table IV. Stop words are obtained from Natural Language Toolkit (nltk) library and extended with customized Indonesian stop words from GitHub.

TABLE IV
STOP WORD REMOVAL

Stop Word
adalah, adanya, adapun, aku, anda, antara, apa, berapa, berlalu, cara, jika, sebagaimana, sebut, terus, ungkap, untuk, yaitu, yang

6) *Stemming*
Stemming is the stage of cleaning affixes, including prefixes, suffixes or a combination of both. With stemming, words with the same base word will be considered to have the same token. This helps in increasing data processing performance. The results of the stemming data can be seen in Table V.

TABLE V
STEMMING

Tweet	Stemming
[proyek, ikn, membuka, peluang, oligarki, bermain, curang, securangnya, tolak, uu, ikn]	[proyek, ikn, buka, peluang, oligarki, main, curang, curang, tolak, uu, ikn]

D. Term Weighting

Data that has passed the preprocessing stage will be given a weight or value in this process. This weighting is done to measure the size of the influence of a word in a document. In addition, this weighting is carried out using various methods to compare and determine the best term weighting method. Weighting calculation is done per tweet. The term weighting raised in this study is as follows.

1) *Term Frequency – Inverse Document Frequency*
Term Frequency is a measurement of how many times a term is present in a document. While Inverse Document Frequency is the total number of documents that contain certain words by giving low weights to words that appear frequently and high weights to words that rarely appear [16]. Hence, Term Frequency – Inverse Document Frequency, or hereinafter referred to as *tf-idf* is a term weighting method

that merges Term Frequency (TF) and Inverse Document Frequency (IDF). The weight of the *tf-idf* value can be calculated by equation [17]:

$$tf_{td}idf_t = tf_{td} \times \log\left(\frac{N}{df_t}\right) \quad (1)$$

Where:

$tf_{td}idf_t$: Weight of word *t*.
tf_{td} : Number of occurrences of the word *t* in a document.
N : Number of total documents.
df_t : Number of documents containing the word *t*.

The *tf-idf* and SVM method along with 5-Fold Validation delivers rather acceptable accuracy of 84.7%, precision of 84.9%, recall of 84.7%, and f-measure of 84.8%. The effect of *tf-idf* on model performance measurement is not significant, although it is better [18].

2) *Term Frequency – Relevance Frequency*
Relevance Frequency or so-called *rf* is a method developed by *tf-idf* in which this method calculates the relevance of documents based on the frequency of occurrence of words in related categories. The *tf-rf* value can be calculated by equation [15]:

$$tf_{td}rf = tf_{td} \times \log\left(2 + \frac{b}{\max(1, c)}\right) \quad (2)$$

Where :

$tf_{td}rf$: Weight of word *t*.
tf_{td} : Number of occurrences of the word *t* in a document.
b : Number of documents that contain the word *t.*
c : Number of documents that do not contain the word *t.*

In research [15] a comparison of the weighting features was carried out using the SVM classification method on MDSD books data. The *tf-rf* accuracy results were 81.00%, a difference of 1.02% from *tf-idf* and *tf-idfec* with an accuracy of 80.02%.

3) *Inverse Category Frequency – Binary Classification*
Inverse Category Frequency or hereinafter referred to as *icf* has been generally used in text classification. One of them is the research of Quan et al. which uses the variation of *icf* as one of the factors in the categorization of questions. In *icf*, the fewer categories of a term appear, the greater the distinguishing power of the term contributes to the categorization of the text. This supports rare terms and biases towards popular terms in that category. Therefore, Wang D. et al. introduce *icf* to the term weighting scheme and proposes 2 approaches, namely *tf-icf* and *icf-based* supervised term weighting. Calculation of *tf-icf* and *icf-based* can be seen in the following equation [9]

$$tf.icf(t_i, d_j) = tf(t_i, d_j) \times \log\left(1 + \frac{|C|}{cf(t_i)}\right) \quad (3)$$

$$icf - based(t_i, d_j) = tf(t_i, d_j) \times \log\left(2 + \frac{a}{max(1, c)} \times \frac{|C|}{cf(t_i)}\right) \quad (4)$$

Where |C| denotes the number of categories in the data, and *cf* indicates the number of classes in which the term is involved. The *icf-based* or *tf-bin.icf* weighting has three main factors. First, the *tf* factor as the raw term frequency. Second, the *rf* calculates the distribution of terms between two categories. Third, the *icf* measures the distribution of terms in terms between categories. The *cf* information can be obtained and stored when analyzing training data so that it can be used to calculate term weights after the training data is divided into positive and negative categories. Wang D. et al.'s research [9] revealed that *icf-based* is superior in binary categorization compared to other term weights.

E. Support Vector Machine

The Support Vector Machine (SVM) is a tool that can be used to make predictions in classification and regression cases [19]. The main goal of this SVM is to discover the best hyperplane by expanding the distance between classes. The hyperplane is a term used for separators between classes [20].

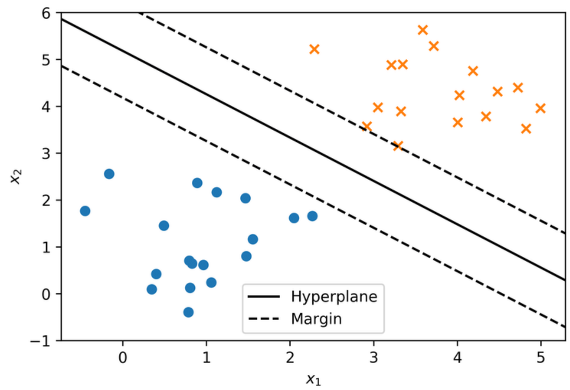

Fig. 2. SVM Illustration

Figure 2 shows that the hyperplane lies in the midst of the two classes, with the distance between the hyperplane and the data items varying from the neighboring (outermost) class. The outer data item closest to the hyperplane is referred to as the support vector in SVM. Because their locations are virtually overlapping with other classes, support vectors are the most challenging objects to categorize. This is what SVM calculates in order to determine the best hyperplane.

The core concept of SVM is a linear classifier that was later modified to operate on non-linear issues. By utilizing the kernel scheme idea in a high-dimensional workspace [21]. Linear Kernel is the most basic kernel. This kernel proves to be the best and most frequently used function for text classification problems because most of these types of classification problems can be separated linearly. The linear kernel is used when the data can be divided into separate categories using a single straight line. It is frequently used, particularly when the dataset has a large number of features. An example of this is text classification, where each individual letter is considered a feature. Therefore, the linear kernel is often utilized in text classification tasks. The Linear Kernel formula can be seen as follows:

$$F(x, x_j) = sum(x \cdot x_j) \quad (5)$$

Where x and x_j represents the data to be classified, and $(\cdot)$ shows the dot product of the two values. SVM is more attractive theoretically and in practice [22]. In a comparative study of SVM with other classification methods, the SVM classification method is superior to different classification methods [10].

F. Validation and Evaluation

Measuring the performance of the classification model that has been made is done by validation and evaluation. Validation was carried out using the K-Fold cross-validation method. K-Fold cross-validation is a cross-validation model carried out by dividing the data and doing iterations/repetitions of as many as *k* values. While the evaluation was carried out using the Confusion Matrix (see Fig. 3). The Confusion Matrix contains a table with 4 different combinations of predicted and true values.

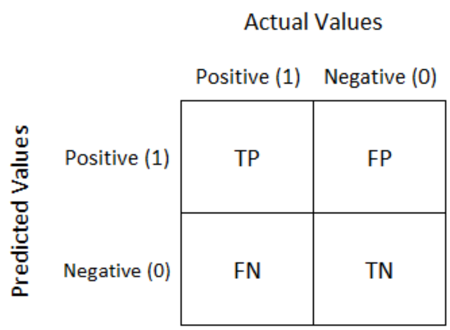

Fig. 3. Confusion Matrix

True Positive (TP) is a condition with a positive value predicted to be true. Meanwhile, True Negative (TN) is a condition with a negative value predicted to be true. Furthermore, False Positive (FP) is a condition that is negative but is predicted to be true. Meanwhile, False Negative (FN) is a state that is positive and which is expected to be wrong [23]. From this Confusion Matrix, the performance of the classification model can be calculated, including the:

1) *Accuracy*

Accuracy is the ratio of data classified correctly compared to all data. The equation can calculate the accuracy value:

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{6}$$

2) *Precision*

Precision is the comparison between positive values that are correctly predicted and all data that are expected to be positive, whether true or not. The equation can calculate precision:

$$precision = \frac{TP}{TP+FP} \tag{7}$$

3) *Recall*

Recall is the ratio of data correctly classified as positive compared to all existing positive data. The equation can calculate recall:

$$recall = \frac{TP}{TP+FN} \tag{8}$$

4) *F1 Score*

F1 Score is a performance metric considering both precision and recall. The equation calculates F1 Score:

$$f1\ score = \frac{2*(recall*precision)}{(recall+precision)} \tag{9}$$

IV. RESULTS AND DISCUSSION

The data used in this study are 5892 Indonesian language tweet data with the topic of the new State Capital with the keywords "national capital" and "ikn" which have been crawled previously using the *snscrape* library. Data has been labeled with two categories namely positive and negative. Data must be labeled with two categories because it relates to the term weighting used, namely the *tf-bin.icf* where its use is specific to binary classification or two-category classification [9]. The data distribution in this research can be seen in Table VI.

TABLE VI
DATA DISTRIBUTION

Sentiment	Number of Data
Positive	3750
Negative	2142

A. Testing Scenario and Test Result

Model testing was performed on each term weighting used with the same dataset and classification method, which is SVM. In other words, the model was tested three times with different term weighting namely *tf-idf*, *tf-rf*, and *tf-bin.icf*. In each test, K-Fold was carried out to measure how often the classifier is predicting the *right* class, or in other words, how *accurate* is the classifier in selecting positive items and the negative items.

1) *Testing Scenario with TF-IDF Term Weighting Feature*

The first scenario is carried out by measuring the performance of the *tf-idf* feature weighting in the SVM classification model. Data that has gone through the preprocessing stage then enter the weighting stage. Then the data splitting was carried out using the 10-Fold cross-validation method to determine the best accuracy of the several splitting iterations carried out. Iteration is carried out 9 times, from $k = 2$ to $k = 10$. The variations of the train and test data use a combination of 10 parts with a train and test ratio of 9:1. The results can be seen in Table VII.

TABLE VII
10-FOLD EXPERIMENT ON TF-IDF TERM WEIGHTING

k	Average Accuracy	Max Accuracy
2	87,1%	87,1%
3	87,3%	87,6%
4	87,7%	88,9%
5	87,8%	88,5%
6	87,8%	88,7%
7	**88,0%**	**89,2%**
8	87,9%	88,7%
9	87,9%	89,0%
10	87,9%	89,3%

Based on these experiments, the value of $k = 7$ is the best value to obtain the best accuracy. As a result, the combination of *tf-idf* weighting with the SVM classification method obtained 1676 data TP, 466 FP, 243 FN, and 3507 data TN as shown in Fig.4. Hence, we got an accuracy value of 88.0% and an f1-score value of 90.8%.

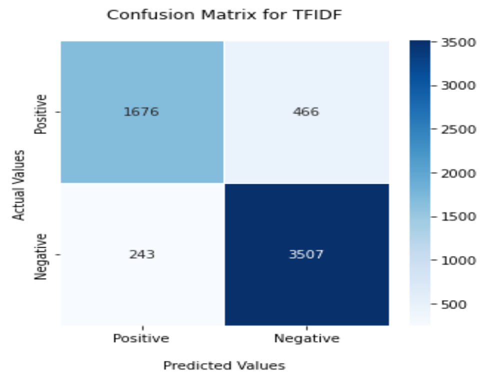

Fig. 4. Classification Results *tf-idf* with SVM

2) Testing Scenario with TF-RF Term Weighting Feature

The second scenario is carried out by measuring the performance of the *tf-rf* feature weighting in the SVM classification model. As in the first test, data that has gone through the preprocessing stage enters the weighting stage. The data splitting was also carried out using the 10-fold cross-validation method to determine the best accuracy of the several splitting iterations. Iteration is also carried out as in the first scenario. The results can be seen in Table VIII.

TABLE VIII
10-FOLD EXPERIMENT ON TF-RF TERM WEIGHTING

k	**Average Accuracy**	**Max Accuracy**
2	84,6%	85,5%
3	85,3%	86,2%
4	85,6%	87,1%
5	85,7%	86,7%
6	85,7%	87,3%
7	85,9%	86,9%
8	85,7%	87,1%
9	86,0%	87,6%
10	**85,9%**	**88,1%**

The combination of tf-rf weighting with the SVM classification method receives 1488 data TP, 654 FP, 173 FN, and 3577 data TN as shown in Figure 5. Hence, we got an accuracy value of 86.0% and an f1-score value of 89.6%.

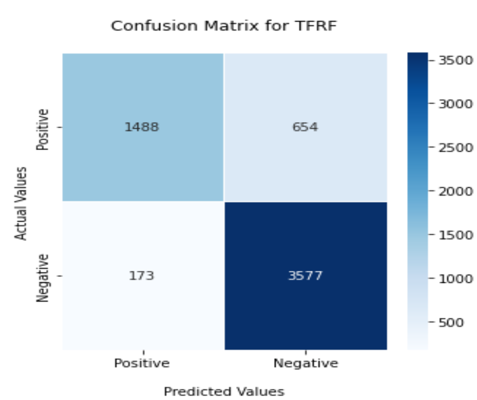

Fig. 5. Classification Results *tf-rf* with SVM

3) *Testing Scenario with TF-BinICF Term Weighting Feature*

The third scenario is carried out by measuring the performance of the *tf-bin.icf* feature weighting in the SVM classification model. As in the first and second tests, the data that has gone through the preprocessing stage then enters the weighting stage. The data splitting was also carried out using the K-fold cross-validation method to determine the best accuracy of the several splitting iterations. Iterations are also carried out as in the first and second scenarios. The results can be seen in Table IX.

TABLE IX
10-FOLD EXPERIMENT ON TF-BINICF TERM WEIGHTING

k	Average Accuracy	Max Accuracy
2	86,7%	87,0%
3	86,8%	87,3%
4	87,3%	88,1%
5	87,1%	88,2%
6	87,2%	88,8%
7	87,3%	88,4%
8	87,3%	88,7%
9	87,3%	88,7%
10	**87,4%**	**90,2%**

Based on these experiments, *k = 10* is the best value to obtain the best accuracy. As a result, the combination of *tf-bin.icf* weighting with the SVM classification method received 1609 data TP, 533 FP, 213 FN, and 3537 data TN as shown in Fig.6. Hence, we got an accuracy value of 87,3% and an f1-score value of 90.5%.

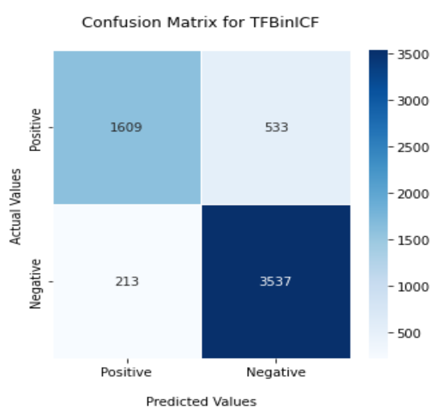

Fig. 6. Classification Results *tf-bin.icf* with SVM

Based on the results of testing the three feature weighting methods that have been tested using the SVM classification method and using the K-fold cross-validation, it can be seen that the *tf-bin.icf* feature weighting can compete even more superior to the more popular term weighting method, namely *tf-idf* and *tf-rf*. A comparison of the accuracy value and f1-score of the feature weights that have been tested can be seen in table 10. The accuracy score measures how many predictions made by a model are correct. However, this score should not be the only factor used to compare models, as other metrics such as precision and recall should also be considered. Precision measures the proportion of positive predictions that are actually correct, while recall measures the proportion of actual positive classes that were correctly predicted. The F1 score is a balance between precision and recall and is a better overall measure of a model's performance. As shown in table 10, *tf-bin.icf* and *tf-rf* have higher recall scores but lower precision scores, leading to lower F1 scores.

TABLE X
ACCURACY COMPARISON

Metrics	Term Weighting		
	tf-idf	***tf-bin.icf***	***tf-rf***
Accuracy	**88,0%**	87,3%	86,0%
Precision	**88,3%**	86,9%	84,5%
Recall	**93,5%**	94,3%	95,4%
F1 Score	**90,8%**	90,5%	89,6%

V. CONCLUSION

From the comparison of feature weighting accuracy in Table X, the accuracy and f1 score of each weighting is not much different. However, because the test was carried out with K-fold validation, the test results were the best among the several possible tests that occurred. The *tf-idf* weighting has proven to be the most optimal feature weighting with the SVM classification method on sentiment analysis for new state capitals. Although, *tf-bin.icf* is still comparable to *tf-idf* with a difference of 1,3% accuracy score and 0,3% f1 score.

In further research, other classification methods can be tested or use other SVM kernel variants. Further research can also be carried out for other sentiment analysis topics and with multi-class categories. This research also has not paid attention to negation sentences in tweets, so tweets with negation words cannot be determined properly. This can be the main focus of further research.

VI. REFERENCES

[1] M. Safiullah, P. Pathak, S. Singh, and A. Anshul, “Social media as an upcoming tool for political marketing effectiveness,” *Asia Pacific Management Review*, vol. 22, no. 1, pp. 10–15, Mar. 2017, doi: 10.1016/j.apmrv.2016.10.007.

[2] D. Neu, G. Saxton, A. Rahaman, and J. Everett, “Twitter and social accountability: Reactions to the Panama Papers,” *Critical Perspectives on Accounting*, vol. 61, pp. 38–53, Jun. 2019, doi: 10.1016/j.cpa.2019.04.003.

[3] Presiden Republik Indonesia, “Undang-undang Republik Indonesia Nomor 3 Tahun 2022 Tentang Ibu Kota Negara,” 2022.

[4] A. Giachanou and F. Crestani, “Like it or not: A survey of Twitter sentiment analysis methods,” *ACM Computing Surveys*, vol. 49, no. 2. Association for Computing Machinery, Jun. 01, 2016. doi: 10.1145/2938640.

[5] B. Liu, “Sentiment Analysis and Opinion Mining,” Morgan & Claypool Publishers, 2012.

[6] B. Liu, *Sentiment Analysis : Mining Opinions, Sentiments, and Emotions*. 2015. doi: doi:10.1162/COLIr00259595.

[7] P. Arsi and R. Waluyo, “Analisis Sentimen Wacana Pemindahan Ibu Kota Indonesia Menggunakan Algoritma Support Vector Machine (SVM),” vol. 8, no. 1, pp. 147–156, 2021, doi: 10.25126/jtiik.202183944.

[8] A. Alsaeedi, “A survey of term weighting schemes for text classification,” 2020. doi: https://doi.org/10.1504/IJDMMM.2020.106741.

[9] D. Wang and H. Zhang, “Inverse-Category-Frequency based Supervised Term Weighting Schemes for Text Categorization,” XXX-XXX, 2010. doi: https://doi.org/10.48550/arXiv.1012.2609.

[10] Z. Liu, X. Lv, K. Liu, and S. Shi, “Study on SVM compared with the other text classification methods,” in *2nd International Workshop on Education Technology and Computer Science, ETCS 2010*, 2010, vol. 1, pp. 219–222. doi: 10.1109/ETCS.2010.248.

[11] A. P. Deepak and S. Chouhan, “SVM Kernel Functions for Classification,” 2013. doi: 10.1109/ICAdTE.2013.6524743.

[12] K. Sailunaz and R. Alhajj, “Emotion and sentiment analysis from Twitter text,” *J Comput Sci*, vol. 36, Sep. 2019, doi: 10.1016/j.jocs.2019.05.009.

[13] P. Gandhi, S. Bhatia, and N. Alkhaldi, “Sentiment Analysis Using Deep Learning,” IET, 2021, pp. 204–211.

[14] A. P. Natasuwarna STMIK Pontianak Jurusan Sistem Informasi, “Analisis Sentimen Keputusan Pemindahan Ibu Kota Negara Menggunakan Klasifikasi Naive Bayes,” *SEMINAR NASIONAL SISTEM INFORMASI dan TEKNIK INFORMATIKA*, pp. 47–54, 2019.

[15] G. Domeniconi, G. Moro, R. Pasolini, and C. Sartori, “A Comparison of Term Weighting Schemes for Text Classification and Sentiment Analysis with a Supervised Variant of tf.idf,” *Communications in Computer and Information Science*, vol. 584, p. v, 2016, doi: 10.1007/978-3-319-30162-4.

[16] E. Uwiragiye and K. L. Rhinehardt, “TFIDF-Random Forest: Prediction of Aptamer-Protein Interacting Pairs,” *IEEE/ACM Trans Comput Biol Bioinform*, 2021, doi: 10.1109/TCBB.2021.3098709.

[17] B. Trstenjak, S. Mikac, and D. Donko, “KNN with TF-IDF based framework for text categorization,” in *Procedia Engineering*, 2014, vol. 69, pp. 1356–1364. doi: 10.1016/j.proeng.2014.03.129.

[18] S. Fransiska and A. Irham Gufroni, “Sentiment Analysis Provider by.U on Google Play Store Reviews with TF-IDF and Support Vector Machine (SVM) Method,” *Scientific Journal of Informatics*, vol. 7, no. 2, pp. 2407–7658, 2020, [Online]. Available: http://journal.unnes.ac.id/nju/index.php/sji

[19] S. Styawati and K. Mustofa, “A Support Vector Machine-Firefly Algorithm for Movie Opinion Data Classification,” *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 13, no. 3, p. 219, Jul. 2019, doi: 10.22146/ijccs.41302.

[20] A. Kowalczyk, “Support vector machines succinctly,” *Syncfusion Inc*, 2017.

[21] S. Patnaik and X. Li, “Proceedings of International Conference on Computer Science and Information Technology,” 2013. [Online]. Available: http://www.springer.com/series/11156

[22] F. Colas and P. Brazdil, “Comparison of SVM and Some Older Classification Algorithms in Text Classification Tasks,” 2006.

[23] E. Haddi, X. Liu, and Y. Shi, “The role of text pre-processing in sentiment analysis,” in *Procedia Computer Science*, 2013, vol. 17, pp. 26–32. doi: 10.1016/j.procs.2013.05.005.