Accredited by National Journal Accreditation (ARJUNA) Managed by
Ministry of Research, Technology, and Higher Education, Republic Indonesia with Second Grade (Peringkat 2, Sinta 2)
Since year 2017 to 2021 according to the decree No. 10/E/KPT/2019

Published online on the journal's webpage: **http://jurnal.iaii.or.id**

RESTI JOURNAL
(System Engineering and Information Technology)
Vol. 4 No. 5 (2020) 846 - 853 ISSN Electronic Media: 2580-0760

Sentiment Analysis on KAI Twitter Post Using Multiclass Support Vector Machine (SVM)

Dhina Nur Fitriana[1], Yuliant Sibaroni[2]
[1,2]Informatics, School of Computing, Telkom University
[1]dhnnur@student.telkomuniversity.ac.id, [2]yuliant@telkomuniversity.ac.id

Abstract

Information in form of unstructured texts is increasing and becoming commonplace for its existence on the internet. This information is easily found and utilized by business people or companies through social media. One of them is Twitter. Twitter is ranked 6th as a social media that is widely accessed today. The use of Twitter has the disadvantage of unstructured and large data. Consequently, it is difficult for business people or companies to know opinion towards service with limited resources. To Make it easier for businesses know the public's sentiment for better service in the future, public sentiment on Twitter needs to be classified as positive, neutral, and negative. The Multiclass Support Vector Machine (SVM) method is a supervised learning classification method that handles three classes classification. This paper uses One Against All (OAA) approach as a method to determine the class. This paper contains the results of classifying OAA Multiclass SVM methods with five different weighting features unigram, bigram, trigram, unigram+ bigram, and word cloud for analyzing tweet data, finding the best accuracy and important feature when processed with large data. The highest accuracy is the unigram TF-IDF model combined with the OAA Multiclass SVM with gamma 0.7 is 80.59.

Keywords: Twitter Data, Sentiment Analysis, Text Classification, Term Frequency-Invers Document Frequency.

1. Introduction

Information in the form of unstructured text-based documents is increasing and becoming commonplace on the internet. This happens because of the increase in internet users every year[1]. This information is often easily found and utilized by businesses or companies through social media, one of them is Twitter. Twitter is ranked 6th as a social media that widely accessed today[2]. The use of Twitter in Indonesia has led many business people to use it as a communication media to complaints, questions, or suggestions on a given service so, that it will be better in the future.

The use of Twitter has the disadvantage of an unstructured and large amount of text data. Twitter contains complaints about facilities, questions relating to service, or appreciation of customer satisfaction. This makes it difficult for business people or companies to know public sentiment towards service with limited resources. There are weaknesses in research by Windasari et al.[3] which discusses the Twitter data classification of Gojek, namely classification only into positive and negative, while there are many found tweets that are neutral so that the data needs to be classified into neutral sentiment. To Make easier for businesses to know the public's response for better service in the future, public sentiment on Twitter needs to be classified as positive, neutral, and negative. *C*lassification in this research using the TF-IDF approach and machine learning method to facilitate the admin knows information/responses from customers.

Term Frequency Inverse Document Frequency (TF-IDF) is a method for giving weight to a word (term) in a document. TF-IDF features can be adapted to the form of data with machine learning methods to select the best and accurate features in the classification of tweet data. Previous research on weighting bigram features was investigated by Gleen et al. [4] which explains the TF-IDF method that integrates collocation as a feature. This study aims to overcome the weaknesses of Term Frequency-Inverse Document Frequency (TF-IDF) in dealing with single terms. The results show that there is a 10% increase in accuracy compared to TF-IDF without collocation integration.

In previous studies, according to Windasari et al.[3], Support Vector Machine method is a supervised learning method that more optimal than the Naïve Bayes method.

Accepted by editor: 30-07-2020 | Final Revision : 06-09-2020 | Online Publication : 30-10-2020

However, the Support Vector Machine method can only handle two classes or binary classifications. In this study, it is necessary to develop the SVM method to handle non-binary classifications case, namely the Multiclass Support Vector Machine (SVM) approach that handles the classification of more than two classes. There are two approaches to implementing the Multiclass Support Vector Machine method by combining several binary SVMs, namely One Against All (OAA) and One Against One (OAO). Multiclass SVM OAA approach in the research of Hejazi et al., Pratama et al. [5], and Mustakim et al. [6] has a better accuracy value than Multiclass SVM OAO. Based on the discussion above, this paper will classify Twitter data user sentences into positive, neutral and negative using the Multiclass Support Vector Machine (SVM) One Against All (OAA) classification using a radial basis function kernel with five TF-IDF feature weighting approaches namely unigram[3], bigram [4], trigram [7], unigram + bigram, and word cloud [8] to map people's sentiment into positive, negative, or neutral. From the combination of five different features will be found the best features for data classification. The Input of this research is a collection of data from the Twitter scrapper and the output of this research is the performance classification of community sentiments.

This research aims to determine the performance of the Multiclass Support Vector Machine (SVM) method for the classification of Twitter data and find out the best TF-IDF feature group seen from the accuracy values obtained. So that, it can find out information in the form of public sentiment in service facilities, questions, and complaints.

2. Research Method

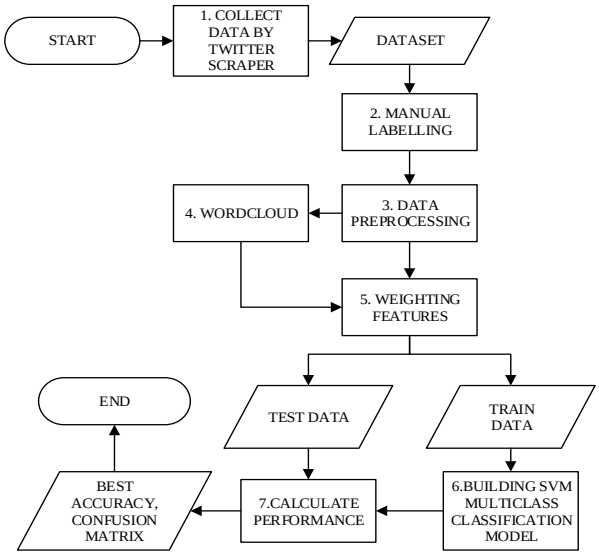

Figure 1 System Architecture

The classification system built in this research is a system that can classify public sentiments in the Indonesian language. In this study, sentiment analysis is based on the classification of community tweets on the Twitter account @KAI121 collected through Twitter scrapper. A collection of tweets is used as training data with a label and then testing with test data. The results of the performance of the Multiclass Support Vector Machine (SVM) method and five different TF-IDF features approaches recognize positive, negative, and neutral tweets to be the focus of research. The following system architecture is built in Figure 1.

2.1. Data set

Data collection executed through Twitter scrapper on the account @kai121 from January 2018 to January 2020 as many as 7000 data will be training data and test data by manually determining sentiment labels. Determination of labels is done by analyzing tweets and grouping sentences containing good words or appreciation such as good, cool, happy, etc. into positive classes, sentences containing questions of schedule or facility classify into neutral, and sentences that contain complaints and bad words classify into negative. Labeling is done by the author together with a partner consisting of three people. Furthermore, data preprocessing is executed. In Table 1, it is explained about an example of class labeling in tweet data:

Table 1 Examples of Tweets and Classes

No	*Tweet*	Class
1	Terimakasih @KAI121 perjalanan ku bersama JogloSemarkerto menuju Purwokerto menyenangkan	Positif
2	Prosedurnya bagaimana ?	Netral
3	Adminnya tidak profesional	Negatif

2.2 Preprocessing Data

At these steps, data preprocessing is performed on training data and test data to optimize data features that have the same meaning but have different writing so that it is easy to process. In the system architecture, there are five stages of preprocessing.

The first step is Case folding. Case folding is a step in data processing that aims to change or eliminate all capital letters in the document into lowercase letters [9]. Data that has been collected from Twitter then carried out the Case Folding process. In Figure 2 shows an example of Case Folding process.

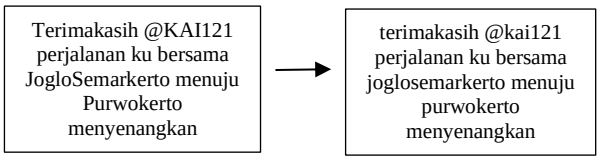

Figure 2 Case Folding

The second step is Remove Punctuation. Remove Punctuation is a step taken on a document to delete or eliminate some punctuation or numbers that have no relation to the document. Punctuation or numbers that do

not have a relationship will decrease the performance value of the classification process. In Figure 3 shows an example of Remove Punctuation process.

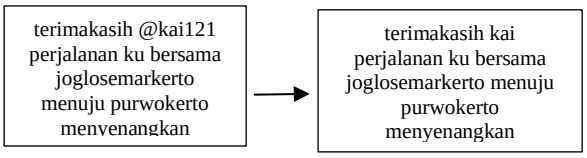

Figure 3 Remove Punctuation

The third step is normalization. Normalization is a step taken to change abbreviations, acronyms, or ambiguous words in documents. The normalization can handle unbalanced data. The normalization phase converts 530 words that researchers get by analyzing and turning them into standard words according to the KBBI. In Table 2 shows an example of some words in the normalization process.

Table 2 Normalization

No	Original Word	Normalization Word
1	aja	saja
2	aj	saja
3	gak	tidak
4	yg	yang
5	st	stasiun

The next step is stemming. Stemming is the process of removing the prefix and suffix in a word to get the root word of a document. The stemming process in this paper uses the Indonesian language Sastrawi Stemming library. Sastrawi stemming library applies the Nazief and Andriani algorithms. The results from the previous normalization steps are processed to carry out stemming. In Figure 4 shows an example of stemming process.

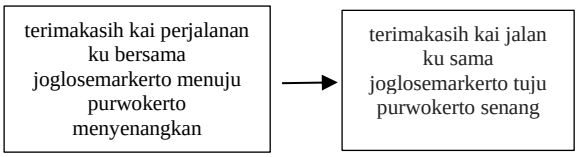

Figure 4 Stemming

The last step is Tokenization. Tokenization is the process of breaking a character sequence into several parts (words/phrases) called tokens[10]. In Figure 5

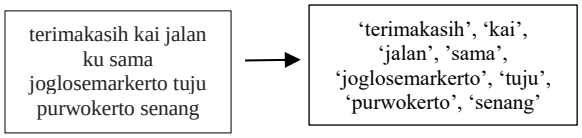

Figure 5 Tokenization

2.3 Wordcloud

Wordcloud comes as a direct and interesting method of visualizing text. Wordcloud is usually used in various contexts to provide an overview by filtering out text in the form of words with high-frequency values[9]. This research uses a word cloud as a technique to filtering words on each sentiment which will then be used as a feature during the TF-IDF process. The word cloud process is done in several steps. In the Figure 6, shows an illustration to get word cloud features. Table 3 shows the word cloud result.

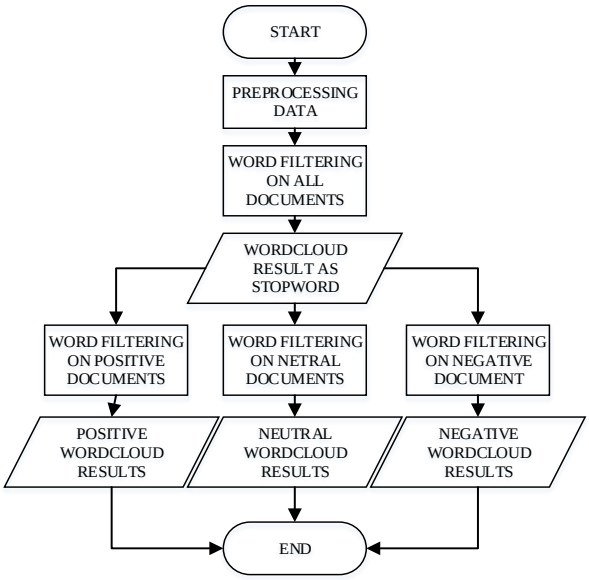

Figure 6 Steps of Wordcloud

Table 3 Wordcloud Result

No	Document	Wordcloud Data
1	All	kai access, ini, nya, sampai, dari, di, saya mau, beli tiket, tidak, baru, kalau, saya, terima kasih, tidak bisa, jalan, yang, haru, tapi, karena. kakak, juga, ke, kereta, admin kai, sama, ya admin, tuju, mau tanya, bagaimana ya, tiket kereta, kereta api, dan, di stasiun, atau, dengan, sekarang, buat, jadi, lagi, saja, ya, itu, sudah, ada, admin mau, apakah, tidak ada, jam, apa, untuk.
2	Positive	aman, guna, moga, eksekutif, bersih, naik, sangat, lebih, jalan, alhamdulillah, layan, ekonomi, malam, dapat, mantap, gerbong, kembali, terima kasih, sekali, suka, lokal, enak, bagus, masih, kursi, bisa, banget, tugas, bagus, makin, bikin, hari, banyak, semua, biar, seperti, tambah, nyaman, selalu, tumpang, bapak, haru, sedia, stasiun, aku, dong, makan, keren, pakai, thank
3	Neutral	dapat, habis, aku, berapa, sedia, aplikasi, terus, belum, masih, ktp, bisa, harga, gambir, haru, loket, hari, pasar senen, bagaimana, mesan, nanya, sore, jadwal, batal, bandung, lewat, apakah, bayar, mana, bagaimana cara, pada, tariff khusus, berangkat, mohon info, gerbong, jalan, tumpang, kenapa, naik, apakah bisa, pesan, kursi, lokal, pakai, kapan, buka, malangekonomi, seperti, Jakarta, tanya
4	Negative	gerbong, terus, apakah, aplikasi, lalu, coba, jalan, tolong, lebih, masih, tanya, seperti, bayar, cek, kali, dapat, tumpang, pesan, banyak, belum, selalu, lokal, malah, benar, padahal, kenapa, hari, mohon, bisa, sih, tadi, masuk, telat, bagaimana, jadwal, pakai, harga, naik, nih, pas, berangkat, error, kursi, lama, semua, gabisa, harus, muncul, saat, banget.

2.4 Weighting Features

Weighting Features in this study play an important role in text classification because it can affect the accuracy of classification. Feature weighting is based on vector space models where features are seen as points in N-dimensional space. Each point dimension represents one text feature. Feature extraction algorithms usually use a collection of keywords. Based on a collection of keywords that have been obtained, the feature weighting algorithm calculates the word weight in a text or document and then forms a digital vector [11]. Term Frequency Inverse Document Frequency (TF-IDF) is a method for giving the weight of a word (term) to a document. Term Frequency (TF) means the number of words that appear in a text, and IDF is the Inverse Document Frequency, an algorithm used to calculate the inverse value of the probability of finding words in a text [12]. The method applied in this study is the unigram[3], bigram[4], trigram[7], unigram + bigram model, and word cloud features. The next word-based n-gram model will be weighted for each word that forms a tweet sentence. Word cloud is a weighting feature that contains features that often appear in tweets to determine the effect on the classification process. Tweets that contain rare words have a higher weight than tweets that contain common words and have a greater effect on classification.

Determination of the weight value in the TF-IDF method based on the frequency of occurrence of terms in research data. This method can produce a large number of feature vectors in large text corpus that can potentially increase the opportunity to adjust the classification model. TF and IDF calculations can be seen in the eqution 1 and 2.

$$Wi = TF(\omega i, d)\ x\ IDF\ (\omega i) \quad (1)$$

$$IDF(\omega i) = \log\left(\frac{|D|}{DF(\omega i)}\right) \quad (2)$$

where Wi is the term word weight (ωi) in a document (d), TF is Term Frequency, the number of terms in one sentence, DF is Document Frequency, the number of terms/words in one document, | D | is the number of sentences in one document, IDF is Inverse Document Frequency. The largest IDF value appears when ωi only appears in one document. In the Table 4 shows an illustrating the weighting of unigram and bigram features in the following statement (for TI, T2, and T3 are from Table 1):

2.5 Multiclass Support Vector Machine (SVM)

Support Vector Machine (SVM) method is a supervised learning method for classifying linear and non-linear data. The workings of the SVM algorithm are to use non-linear mapping to convert training data to higher dimensions and find the most optimal separating hyperplane. Data on a hyperplane is called a support vector[13]. The Support Vector Machine method is a method that handles binary classification cases, for non-binary classifications cases such as positive, negative, and neutral classification a Multiclass Support Vector Machine (SVM) approach is needed that handles the classification of more than two classes. There are two approaches to implementing the Multiclass Support Vector Machine method by combining several binary SVMs namely One Against All (OAA) and One Against One (OAO) or combining optimization of all data.

Table 4 Unigram and Bigrams TF-IDF

Unigram	TF			df	D	idf	W		
	t1	t2	t3				t1	t2	t3
terima	1	0	0	1	3	0.48	0.48	0	0
kasih	1	0	0	1	3	0.48	0.48	0	0
kai	1	0	0	1	3	0.48	0.48	0	0
jalan	1	0	0	1	3	0.48	0.48	0	0
sama	1	0	0	1	3	0.48	0.48	0	0
Joglose-markerto	1	0	0	1	3	0.48	0.48	0	0
tuju	1	0	0	1	3	0.48	0.48	0	0
Purwo-kerto	1	0	0	1	3	0.48	0.48	0	0
senang	1	0	0	1	3	0.48	0.48	0	0
prosedur	0	1	0	1	3	0.48	0	0.48	0
Bagai-mana	0	1	0	1	3	0.48	0	0.48	0
admin	0	0	1	1	3	0.48	0	0	0.48
tidak	0	0	1	1	3	0.48	0	0	0.48
Profesi-onal	0	0	1	1	3	0.48	0	0	0.48
terima kasih	1	0	0	1	3	0.48	0.48	0	0
kasih kai	1	0	0	1	3	0.48	0.48	0	0
kai jalan	1	0	0	1	3	0.48	0.48	0	0
jalan sama	1	0	0	1	3	0.48	0.48	0	0
sama joglose-markerto	1	0	0	1	3	0.48	0.48	0	0
Joglose-markerto tuju	1	0	0	1	3	0.48	0.48	0	0
tuju purwo-kerto	1	0	0	1	3	0.48	0.48	0	0
Purwo-kerto senang	0	1	0	1	3	0.48	0	0.48	0
prosedur gimana	0	0	1	1	3	0.48	0	0	0.48
Admin tidak	0	0	1	1	3	0.48	0	0	0.48
Tidak profesio-nal									

The OAA approach solves Multiclass problems or more than two classes (N classes) with N decision boundaries. The resulting decision boundary is the result of the hyperplane of each i class with the rest of the class. The OAO approach solves Multiclass problems or more than two classes (N classes) with N (N-1) / 2 decision boundaries. The decision boundary is the hyperplane of each class with every other class. This study uses the OAA approach as a model to determine the right class.

The OAA approach has better performance than the OAO approach and also simpler than combining optimization from all data classes. The tweet classification process in this study is divided into two steps, training for model formation using the Multiclass Support Vector Machine method and the testing phase. Figure 7 shows an overview of the training and testing process.

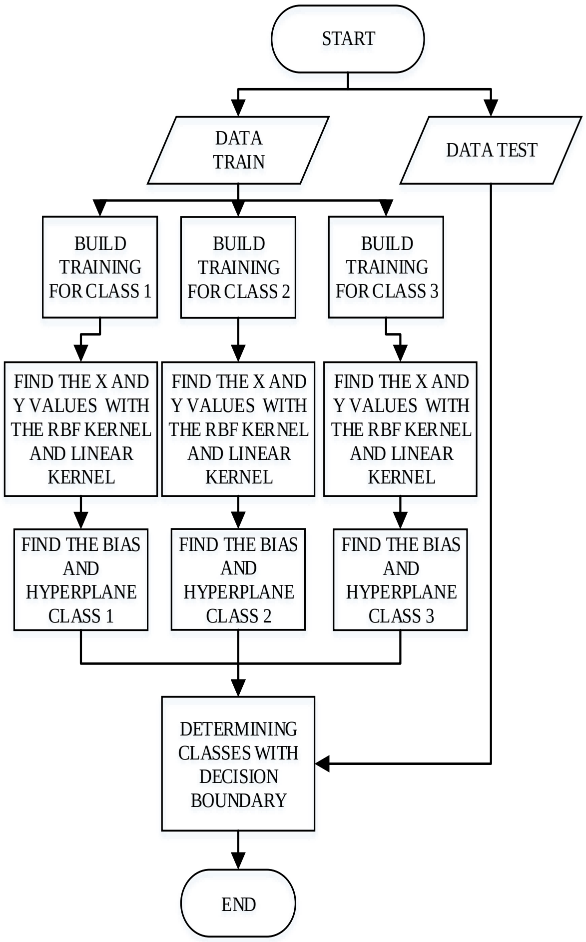

Figure 7 Training and Testing Process

In the Figure 7 there are some stages of training and testing process. First, the formulation (W) used is the duality of the Lagrange Multiplier, which has been modified by the x value of the RBF kernel. Calculate data K (x, xi) from old dimension features to get data with new high dimensional features. The kernel used is the RBF kernel. In the Table 5 shows a portion of the x-xi calculation:

Table 5 Calculate X-Xi

Train Data Class 1			Train Data Class 2			Train Data Class 3		
x1-x1	x1-x2	x1-x3	x2-x1	x2-x2	x2-x3	x3-x1	x3-x2	x3-x3
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	0.48	0.48	-0.48	0	0	-0.48	0	0
0	-0.48	0	0.48	0	0.48	0	-0.48	0
0	-0.48	0	0.48	0	0.48	0	-0.48	0
0	0	-0.48	0	0	-0.48	0.48	0.48	0
0	0	-0.48	0	0	-0.48	0.48	0.48	0
0	0	-0.48	0	0	-0.48	0.48	0.48	0

After getting it, the next step is to do a calculation to get the vector length. Table 6 shows the results of the vector length calculation.

Table 6 Calculate Vector Length

No	Vector Length	Result
1	$\lVert x1-x1 \rVert$	0
2	$\lVert x1-x2 \rVert$	2.502
3	$\lVert x1-x3 \rVert$	2.730
4	$\lVert x2-x1 \rVert$	2.502
5	$\lVert x2-x2 \rVert$	0
6	$\lVert x2-x3 \rVert$	1.137
7	$\lVert x3-x1 \rVert$	2.730
8	$\lVert x3-x2 \rVert$	1.137
9	$\lVert x3-x2 \rVert$	0

Next, the results of the vector length are entered into the RBF kernel. The gamma value used is 0.5. Table 7 shows the calculation of the RBF kernel at the training steps.

Table 7 Calculate RBF Kernel

No	Kernel	Result
1	K(1,1)	$\exp(-\gamma \lVert x1-x1 \rVert^2)$ $\exp((-0.5)(0)^2) = 1$
2	K(1,2)	$\exp(-\gamma \lVert x1-x2 \rVert^2)$ $\exp((-0.5)(2.502)^2) = 0.043$
3	K(1,3)	$\exp(-\gamma \lVert x1-x3 \rVert^2)$ $\exp((-0.5)(2.730)^2) = 0.024$
4	K(2,1)	$\exp(-\gamma \lVert x2-x1 \rVert^2)$ $\exp((-0.5)(2.502)^2) = 0.043$
5	K(2,2)	$\exp(-\gamma \lVert x2-x2 \rVert^2)$ $\exp((-0.5)(0)^2) = 1$
6	K(2,3)	$\exp(-\gamma \lVert x2-x3 \rVert^2)$ $\exp((-0.5)(1.137)^2) = 0.524$
7	K(3,1)	$\exp(-\gamma \lVert x3-x1 \rVert^2)$ $\exp((-0.5)(2.730)^2) = 0.024$
8	K(3,2)	$\exp(-\gamma \lVert x3-x2 \rVert^2)$ $\exp((-0.5)(1.137)^2) = 0.524$
9	K(3,3)	$\exp(-\gamma \lVert x3-x3 \rVert^2)$ $\exp((-0.5)(0)^2) = 1$

After calculating the kernel, the next step is to calculate the value of y. The value of y is obtained from the label or class value that has been given.In the Table 8 shows the value of y.

Table 8 The Value of Y

Y of Training Class 1			Y of Training Class 2			Y of Training Class 3		
y1	y2	y3	y1	y2	y3	y1	y2	y3
1	-1	-1	-1	1	-1	-1	-1	1

The next step is to do the y calculation using the linear kernel calculation by the equation (3).

$$\sum yiyi^T \tag{3}$$

The value of y is the value of the label given. Table 3.9 shows the y value for the class 1 training stage.

Table 9 Value of Y in Class 1 Training

y1	y2	y3
-1	1	1

Then the next step is to find the value of a. The process of getting the value of a begins by converting each statement to a vector value (support vector) with equation (4).

$$\begin{cases} \sqrt{x^2+y^2} > 2 \rightarrow \begin{pmatrix} 4-y+|x-y| \\ 4-x+|x-y| \end{pmatrix} \\ \sqrt{x^2+y^2} \leq 2 \rightarrow \begin{pmatrix} x \\ y \end{pmatrix} \end{cases} \quad (4)$$

As an example, the calculation in the first statement. The equation (5) shows calculation process.

$$\sqrt{1^2+-1^2} = \sqrt{2} \text{ -> } \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad (5)$$

After that, each support vector is given a bias value 1 to get the optimal perpendicular distance and help get the value of b or hyperplane. Then multiply each sentence using the equation (6).

$$\sum_{i=1,j=1}^{n} a_i S_i^T S_j, \quad (6)$$

For example, the calculation in the first statement is,

$$a_1 \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}^T * \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} = 3\ a_1$$

After calculating all statements. Then find the *ai* parameter using the equation (7).

$$\sum_{i=1,j=1}^{n} a_i S_i^T S_j = yi \quad (7)$$

So the shape can be seen as follows:
3 *a1* + 2.002 *a2* + 2.001 *a3* = 1
2.002 *a1* + 3 *a2* + 2.270 *a3* = -1
2.001 a1 + 2.270 *a2* + 3 *a3* = -1
so we get the values a1, a2 and a3 are as follows:

a1 = -418.077 *a2*= 1427.4 *a3*= -802.122

After the parameter ai is obtained, then enter the equation (8).

$$w = \sum_{i=1,j=1}^{n} a_i s_i \quad (8)$$

Then equation (9) is used to get the values of w and b.

$$y = wx + b \quad (9)$$

so we get the result of calculation,

$$W = -418.077 \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} + 1427.4 \begin{bmatrix} 0.043 \\ 1 \\ 1 \end{bmatrix} + -802.122 \begin{bmatrix} 0.024 \\ 1 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} -418.077 \\ 418.077 \\ -418.077 \end{bmatrix} + \begin{bmatrix} 61.378 \\ 1427.4 \\ 1427.4 \end{bmatrix} + \begin{bmatrix} -19.251 \\ -802.122 \\ -802.122 \end{bmatrix}$$

$$= \begin{bmatrix} -375.95 \\ 1043.355 \\ 207.201 \end{bmatrix}$$

W1 = $\begin{bmatrix} -375.95 \\ 1043.355 \end{bmatrix}$, B1 = 207.201.

The steps for finding second and third hyperplanes are the same as determining the first hyperplane. The result of the second and third hyperplane values there are,

W2 = $\begin{bmatrix} -989.111 \\ 2647.599 \end{bmatrix}$, B2 = 207.201

W3 = $\begin{bmatrix} 64.198 \\ 1811.445 \end{bmatrix}$, B3 = 207.201.

After getting the first until third hyperplane values, the next step is to determine the class of test data into a positive, neutral, or negative class. For example, the test data has a support vector value (120,112.2) then at the test steps the vector value is substituted into the equation (10):

$$kelas\ x = \arg\max([w^1]^T.\varphi(x) + b^1, [w^2]^T.\varphi(x) + b^2, [w^3]^T.\varphi(x) + b^3 \quad (10)$$

$$kelas\ x = \arg\max(\begin{bmatrix} -375.95 \\ 1043.355 \end{bmatrix}^T . \begin{bmatrix} 120.112 \\ 2 \end{bmatrix} + 207.201, \begin{bmatrix} -989.111 \\ 2647.599 \end{bmatrix}^T . \begin{bmatrix} 120.112 \\ 2 \end{bmatrix} + 207.201, \begin{bmatrix} 64.198 \\ 1811.445 \end{bmatrix} . \begin{bmatrix} 120.112 \\ 2 \end{bmatrix} + 207.201)$$

= arg max (-42862.189, 113301.701, 10281.043)

The largest hyperplane value is 113301.70. The hyperplane is a value of class 2, it means the test data is included in the neutral class.

2.6. Classification Performance

Classification system performance illustrates how good the system in classifying data. The Confusion Matrix is one of the methods used to measure the performance of a classification method. The confusion matrix contains a comparison between the results of the classification made by the system with the actual classification[14]. Test data enter into the confusion matrix will produce accuracy values. In the Table 10 shows the Confusion Matrix of this study.

Table 10 Confusion Matrix

Class	Classified as Positive	Classified as Negative	Classified as Neutral
Positive	*True Positive (TP)*	*False Negative (FNe)*	*False Neutral (FNt)*
Negative	*False Positive (FP)*	*True Negative (TNe)*	*False Neutral (FNt)*
Neutral	*False Positive (FP)*	*False Negative (FNe)*	*True Neutral (TNt)*

Based on the value of True Negative (TNe), True Neutral (TNt), False Neutral (FNt), False Positive (FP), False Negative (FNe), and True Positive (TP) accuracy values can be obtained. The accuracy value describes how accurately the system can classify data correctly. The accuracy value can be obtained by the equation (11):

$$Accuracy = \frac{TP+TNe+TNt}{TP+TNe+TNt+FP+FNe+FNt} * 100\% \quad (11)$$

Which TP is the amount of positive data classified correctly, TNt is the amount of neutral data classified correctly, TNe is the amount of negative data classified correctly, FP is the amount of positive data but classified incorrectly by the system, FNe the amount of neutral data but classified incorrectly by the system. FNe the amount of negative data but classified incorrectly by the system.

3. Result and Discussion

This section will describe by displaying a table of the number of features of each scheme, the accuracy value, and the confusion matrix of the best accuracy with ratio 90:10 between the train data and the test data. Testing phase on 7000 tweets obtained from the @KAI121 account using the Multiclass Support Vector Machine (SVM) method using the RBF kernel with gamma parameter intervals of 0.4 to 0.9 and TF-IDF weighting with five different features. From the five features, which have the highest accuracy value used as an important feature.

Five features used in the feature weighting process have a different number of features that can affect the classification results. Table 11 shows the number of features of each scheme/scenario.

Table 11 Number of Features each Scheme

Feature	Unigam	Bigram	Trigram	Unigram+ Bigram	Word cloud
Total	7130	50655	77565	57758	116

3.1. Accuracy Results

Table 12 Accuracy Result

No	Feature	Gamma						Average
		0.9	0.8	0.7	0.6	0.5	0.4	
1	Unigram	80.31	80.45	80.59	80.59	79.88	80.17	80.33
2	Bigram	52.6	52.54	52.56	52.12	52.54	52.83	52.53
3	Trigram	53.54	53.82	53.82	53.68	53.54	52.83	53.54
4	Unigram+ Bigram	72.37	75.07	76.20	77.62	77.76	77.76	76.13
5	Word cloud	61.04	61.75	62.18	64.45	66.43	69.26	64.19
Average		63.97	64.72	65.07	65.69	66.03	66.57	

Based on the classification system developed in this study, the TF-IDF unigram system has the best results than the other four features in Table 12. Confusion matrix maps classify correctly and incorrectly classified test data. Table 13 shows illustration of data classified or predicted correctly or incorrectly from the best accuracy, the unigram TF-IDF model and the Multiclass Support Vector Machine (SVM) in detail.

Table 13 Confusion Matrix

Class	Classified as Positive	Classified as Neutral	Classified as Negative
Positive	35	24	13
Neutral	0	365	37
Negative	13	37	166

From Table 12, Each scheme has different values of gamma for the best accuracy. Based on Figure 8, it can be seen that the smaller gamma interval, the accuracy value tends to increase.

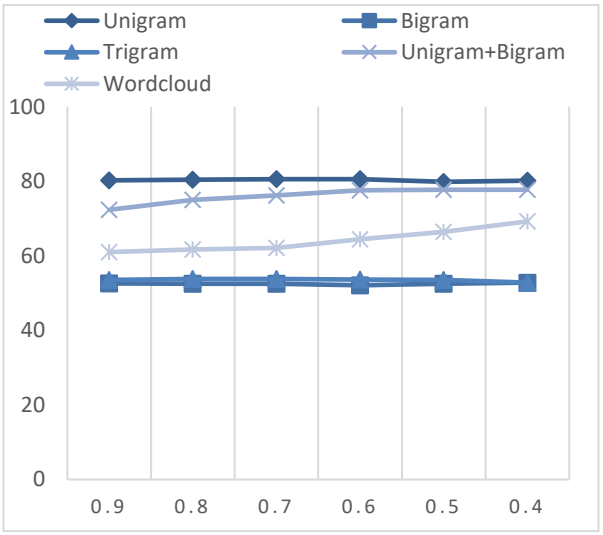

Figure 8 Visualization of The Effect of Gamma

3.2. Discussion

Based on the results of this study, the researcher can describe some analyzes. Table 11 shows the number of features used in the TF-IDF weighting. The unigram feature with the number of features 7103 at the feature weighting stage is very suitable to be combined with the SVM multiclass method in this study, as evidenced by the highest accuracy value obtained among other features. Based on the number of unigram features produced, it can be said that the unigram features are the most efficient and effective in this study. The trigram feature is very inappropriate, inefficient, and ineffective combined with the Multiclass SVM method in this research data sets because it is seen from the highest number of features, namely 77565 with the lowest accuracy value. The use of the word cloud feature in the dataset is not better because many words classify into two classes, it is difficult to classify them correctly. It is better if the features used do not fall into two classes at once.

Table 12 contains the accuracy results of the unigram, bigram, trigram, unigram + bigram, and word cloud features. The Unigram feature has the highest average accuracy value of 80.33 compared to the other four

models, namely Bigram of 52.53, Trigram of 53.54, Unigram + Bigram of 76.13, and Word cloud of 70.33. The highest accuracy value comes from testing the TF-IDF unigram model combined with the Multiclass Support Vector Machine (SVM) classification method with a gamma parameter value of 0.7, namely 80.59. The gamma used can affect the classification results, the smaller the gamma value used, the accuracy results tend to increase. According to the visualization in Figure 8 shows the visualization of the accuracy value with the gamma value variable.

Table 13 shows the configuration matrix of the best accuracy. The best accuracy is obtained from testing the Multiclass Support Vector Machine (SVM) method with 0.7 gamma parameters and Unigram's TF-IDF weighting. Based on Table 13, it can be seen that 35 data are classified as correctly positive, 365 data are correctly classified as neutral, and 166 data are correctly classified as negative.

4. Conclusion

Millions of Twitter users post their opinions on their tweets. Business can use this information to their advantages, but it takes a lot of time. Therefore, there is a need of sentiment analysis that predicted tweet sentiment with TF-IDF and machine learning method. Based on applying five different TF-IDF feature weighting approaches with the Multiclass Support Vector Machine (SVM) method to classify @KAI121 account tweet data for sentiment analysis, researchers can conclude that The highest accuracy results obtained using the SVM OAA multiclass method in analyzing sentiments were obtained at a ratio of 90:10 using the unigram scheme, TF-IDF weighting, and gamma 0.7 parameter values, which amounted to 80.59. An important feature in this study is the unigram feature because it represents a unique feature and produces a high accuracy value. The gamma used can affect the classification results, the smaller the gamma value used, the accuracy tends to increase. Based on the research results, account @kai121 has 11% positive sentiment, 58% neutral sentiment, and 31% negative sentiment. PT. KAI is expected to improve its services to users of railroad transportation services due to the positive sentiment rate which is the value of train user satisfaction is still below average.

Based on the information above, there are some suggestions for further research. It is recommended to do manual labeling to many linguists so that the data used is more valid. In further research, to improve the accuracy of testing can be done by adding the amount of the previous data set and adding vocabulary to the list of normalization so that the dataset is even more balanced than before. Conduct sentiment analysis using different classification methods and weighting features.

References

[1] Number of internet users in Indonesia 2023 | Statista.” [Online]. Available: https://www.statista.com/statistics/254456/number-of-internet-users-in-indonesia/. [Accessed: 17-Sep-2019].

[2] “Indonesia Digital 2019 : Media Sosial - Websindo.” [Online]. Available: https://websindo.com/indonesia-digital-2019-media-sosial/. [Accessed: 17-Sep-2019].

[3] I. P. Windasari, F. N. Uzzi, and K. I. Satoto, “Sentiment analysis on Twitter posts: An analysis of positive or negative opinion on GoJek,” *Proc. - 2017 4th Int. Conf. Inf. Technol. Comput. Electr. Eng. ICITACEE 2017*, vol. 2018-Janua, pp. 266–269, 2018.

[4] G. A. Dalaorao, A. M. Sison, and R. P. Medina, “Integrating Collocation as TF-IDF Enhancement to Improve Classification Accuracy,” *TSSA 2019 - 13th Int. Conf. Telecommun. Syst. Serv. Appl. Proc.*, pp. 282–285, 2019.

[5] M. L. Pratama, “Studi Komparasi Metode Multiclass Support Vector Machine Untuk Masalah Analisis Sentimen Pada Twitter,” *Fmipa Ui*, 2014.

[6] A. Mustakim, I. Santoso, and A. A. Zahra, “Pengenalan Ekspresi Wajah Manusia Menggunakan Tapis Gabor 2-D Dan Support Vector Machine (Svm),” *Transient*, vol. 6, no. 3, p. 232, 2017.

[7] D. De Clercq, Z. Wen, and Q. Song, “Innovation hotspots in food waste treatment, biogas, and anaerobic digestion technology: A natural language processing approach,” *Sci. Total Environ.*, vol. 673, pp. 402–413, 2019.

[8] F. Heimerl, S. Lohmann, S. Lange, and T. Ertl, “Word cloud explorer: Text analytics based on word clouds,” *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, pp. 1833–1842, 2014.

[9] A. M. Pravina, I. Cholissodin, and P. P. Adikara, “Analisis Sentimen Tentang Opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM),” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 3, no. 3, pp. 2789–2797, 2019.

[10] M. Allahyari *et al.*, “A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques,” 2017.

[11] H. Liang, X. Sun, Y. Sun, and Y. Gao, “Text feature extraction based on deep learning: a review,” *Eurasip J. Wirel. Commun. Netw.*, vol. 2017, no. 1, pp. 1–12, 2017.

[12] S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani, “A novel text mining approach based on TF-IDF and support vector machine for news classification,” *Proc. 2nd IEEE Int. Conf. Eng. Technol. ICETECH 2016*, no. March, pp. 112–116, 2016.

[13] “Digital library - Perpustakaan Pusat Unikom - Knowledge Center - WELCOME | Powered by GDL4.2 | ELIB UNIKOM.” [Online]. Available: https://elib.unikom.ac.id/gdl.php?mod=browse&op=read&id=jbptunikompp-gdl-citrawatii-35966&newtheme=gray&newtheme=green. [Accessed: 15-Dec-2019].

[14] “Mengukur Kinerja Algoritma Klasifikasi dengan Confusion Matrix – Achmatim.Net.” [Online]. Available: https://achmatim.net/2017/03/19/mengukur-kinerja-algoritma-klasifikasi-dengan-confusion-matrix/. [Accessed: 13-Nov-2019].