Study The SVM Kernel For Classification Of Covid-19 Vaccine Data On Twitter

Styawati[1,*], Andi Nurkholis[2], Syahirul Alim[3], Nadiya Safitri[4]

[1]Faculty of Engineering and Computer science, Information Systems, Universitas Teknokrat Indonesia, Bandarlampung, Lampung

[2,3]Faculty of Engineering and Computer science, Informatics, Universitas Teknokrat Indonesia, Bandarlampung, Lampung

[4]Faculty of Engineering and Computer science, Computer Engineering, Universitas Teknokrat Indonesia, Bandarlampung, Lampung

Email: [1*]styawati@teknokrat.ac.id, [2]andinh@teknokrat.ac.id, [3]syahirul_alim@teknokrat.ac.id
[4]nadiya_safitri@teknokrat.ac.id

Abstrak−Perkembangan Covid-19 di Indonesia yang begitu pesat pada tahun 2020 menyebabkan pemerintah mewajibkan kepada seluruh rakyat Indonesia untuk melakukan vaksinasi Covid-19. Respon masyarakat terkait kebijakan tersebut, ada yang setuju dan ada juga yang tidak setuju. Respon tersebut banyak dituangkan pada media sosial, salah satunya yaitu Twitter. Media sosial Twitter berada diperingkat ke 5 dalam kategori media sosial yang paling banyak digunakan dengan persentase pengguna sebesar 56%. Hal ini menunjukkan adanya peluang sumber data yang sangat besar yang dapat dimanfaatkan untuk mengetahui sentiment positif dan negatif masyarakat terkait kebijakan pemerintah Indonesia tentang vaksinasi Covid-19. Metode yang digunakan untuk melakukan klasifikasi pada penelitian ini yaitu Support Vector Machine dengan berbagai kernel. Kernel yang digunakan adalah Linear, RBF, Polynomial, dan Sigmoid. Hasil klasifikasi menggunakan kernel tersebut adalah kernel RBF menghasilkan akurasi 88,8%, kernel Linear menghasilkan akurasi 88,3%, kernel Sigmoid menghasilkan akurasi 87% dan kernel Polynomial menghasilkan akurasi 85,5%. Berdasarkan proses klasifikasi yang telah dilakukan, akurasi tertinggi dihasilkan oleh kernel RBF dan akurasi terendah dihasilkan oleh kernel Polynomial.

Kata Kunci: Covid-19, SVM, Kernel, Media SosialTwitter, Vaksin

Abstract−The rapid development of Covid-19 in Indonesia in 2020 caused the government too blige all Indonesian people to carry out the Covid-19 vaccination. The public's response to the policy, some agree and some disagree. The response is widely pouredon social media, one of whichis Twitter. Social media Twitter is ranked 5th in the category of the most used social media with a user percentage of 56%. This shows that there is a very large opportunity for data sources that can be used to find out positive and negative public sentiment regarding the Indonesian government's policy regarding Covid-19 vaccination. The method used to classify in this research is Support Vector Machine with various kernels. The kernels used are Linear, RBF, Polynomial, and Sigmoid. The classification results using the kernel are that the RBF kernel produces an accuracy of 88.8%, the Linear kernel produces an accuracy of 88.3%, the Sigmoid kernel produces an accuracy of 87% and the Polynomial kernel produces an accuracy of 85.5%. Based on the classification process that has been carried out, the highest accuracy is generated by the RBF kernel and the lowest accuracy is generated by the Polynomial kernel.

Keywords: Covid-19, SVM, Kernel, Twitter Social Media, Vaccines

1. INTRODUCTION

Twitter is used by all groups to provide neutral, positive and even negative responses to a trending topic. These social activities are believed to greatly facilitate a person in discussing, doing business, commenting freely[1]. This shows that there is a huge opportunity for data sources that can be used to find out positive and negative public sentiments regarding Covid-19 vaccination. To find out positive and negative sentiments, you can use Text Mining techniques, such as SVM [2][3]. Support Vector Machine (SVM) is a supervised learning classification method that predicts classes based on models or patterns from the results of the training process[4]. Classification is done by looking for a hyperplane or dividing line that separates one class from another. In this case, the line serves to separate positive sentiment tweets from negative sentiment tweets. SVM searches for the maximum hyperlane value using the support vector and margin values. In addition, the success of SVM in classifying data is by selecting the right kernel. Kernel SVM is a measure of similarity that is used to separate non-linear data[5]. The kernels used in this research are Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid.

Sentiment analysis is increasingly advanced and has been widely studied by previous studies. The first study examines Sentiment Analysis on Twitter Posts: An analysis of Positive or Negative Opinion on GoJek. This study aims to propose a system that can detect public sentiment using user opinion tweets about online transportation services, especially GoJek using the SVM method. The results of the tests carried out prove the accuracy rate of the SVM method is 86%, the prediction error rate is 14%, the correct prediction rate for positive sentiment is 100%, and the correct prediction rate for negative sentiment is 67.44%[6]. Another research examines Sentiment Analysis of Teacher Room Applications Using Support Vector Machine and Naive Bayes Classifier Methods. The research discusses how the user sentiment towards the Ruangi Guru application reviews obtained from the Google Play Store with a certain time limit. The analysis uses SVM and Naïve Bayes methods as well as aims to compare which method has the best accuracy. The results of the comparison of methods in the classification process also prove that the SVM method has the best level of accuracy[6].

Based on the explanation that has been done, the research aims to classify Twitter data related to COVID-19 vaccination.In addition, another goal is to see the performance of the Support Vector Machine method and compare the kernels contained in the SVM, so that the best kernel is known based on the accuracy value.

Previous research examines the four kernels in SVM. The kernels compared are Linear, RBF, Polynomial, and Sigmoid kernels. The data used is 4000 tweets about the issue of omnibus law in Indonesia. The results of this study are SVM with RBF kernel produces the highest accuracy of 96.20%, Linear kernel produces 95.73% accuracy, Sigmoid kernel produces 95.45% accuracy, and Polynomial kernel produces the lowest accuracy of 92.3% [7]. Research conducted by Soumya also examines the SVM kernel, but in this study the SVM kernel is compared only the Linear kernel and the RBF kernel. The Linear kernel produces an accuracy of 92.6%, while the RBF kernel produces an accuracy of 92.9. In this study, the RBF kernel got higher accuracy than the Linear kernel, although the difference was small. Another study comparing the performance of the SVM kernel in data classification was conducted by Nadira Putri Arthamevia. This study compared the accuracy of the SVM kernel with Linear, Sigmoid, Polynomial, and RBF kernels. The results of this study indicate that the Linear kernel with C=1, Gamma=Scale produces the highest accuracy, which is 87.03%. While the accuracy generated by the Sigmoid, Polynomial, and RBF kernels is 70.53%. This study also states that weighting with TF-IDF and preprocessing data using Stemming and not using Stopwords removal produces the best accuracy of 88.35%. Another study stated that the success of the SVM method classification depends on the soft margin coefficient C, as well as the parameter of the kernel function. Therefore, it takes the
right combination of SVM parameters to classify the data[8].

2. RESEARCH METHODS

2.1 Research Stage

In this study, there are stages carried out. These stages are shown in Figure 1

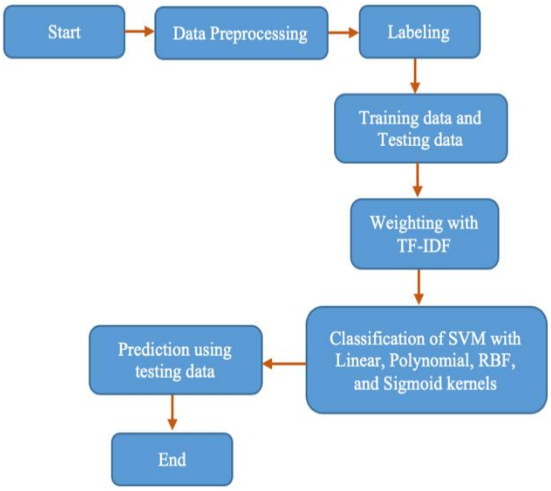

Start
Data Preprocessing
Labeling
Training data and Testing data
Weighting with TF-IDF
Classification of SVM with Linear, Polynomial, RBF, and Sigmoid kernels
Prediction using testing data
End

Figure 1. Research Stage

- Data Preprocessing
  In data preprocessing, it is necessary to clean the data with the aim that the data can be used at a later stage. The stages carried out in this research are cleansing, case folding, tokenizing, filtering, stemming [9].
- Labeling
  At this stage the labeling is done manually, assisted by Mr. Yusniar, an Indonesian language expert. The dataset is divided into 2 categories, namely positive and negative opinions.
- Data Partition

At this stage, 5000 data were divided into training andtest data with a percentage of 80% and 20%. The training data is used to train the model using the SVM algorithm, while the test data is used as a reference and determines the performance of the previously trained model[10].

- Data Weighting

The weighting stage gives weight to each word usingthe calculation of Term Frequency and Inverse Document[11]. To calculate the IDF, Equation 1. is used.

$$idf = \log D/df_i \qquad (1)$$

D = Amount of document

df_i= Occurrence of term from D, After getting the IDF, it can be calculated TF-IDF(w) for each term with Equation 2.

$$\boldsymbol{Wij} = \boldsymbol{tf}_{ij} + (\log \boldsymbol{D}/\boldsymbol{df}_i) \qquad (2)$$

$\boldsymbol{Wij}$ = The weight of the word/term $t_j$to the document d_i $\boldsymbol{tfij}$ = The number of occurrences of the word/term $t_j$in the document d_i

Example

Table 1. DATA TRAINING

Code	Text
D1	alhamdulillah tinggi antusiasme usaha bantu pemerintah sedia vaksin
D2	alhamdulillah vaksin pemerintah gratis tidak bayar lho vaksin ayo gaes
D3	ayo dukung pemerintah galak vaksin covid
D4	dukung perintah nolak vaksin
D5	vaksin covid pemerintah aman halal

The following are the results of the TF-IDF calculation which can be seen in Table 2.

Table 2. TF-IDF

Wdt = TF.IDF				
D1	**D2**	**D3**	**D4**	**D5**
0.39794	0.39794	0	0	0
0.69897	0	0	0	0
0.69897	0	0	0	0
0.69897	0	0	0	0
0.69897	0	0	0	0
0.09691	0.09691	0.09691	0	0.09691
0.69897	0	0	0	0
0	0	0	0	0
0	0.69897	0	0	0
0	0.69897	0	0	0

0	0.69897	0	0	0
0	0.39794	0.39794	0	0
0	0.69897	0	0	0
0	0	0.39794	0.39794	0
0	0	0.69897	0	0
0	0	0	0.69897	0
0	0	0	0.69897	0
0	0	0.39794	0	0.39794
0	0	0	0	0.69897
0	0	0	0	0.69897

- Classification

The results of the preprocessing data are then modeled to classify the data. The model is built with the Support Vector Machine Algorithm, implemented using the Python programming language on Jupyter Notebook. Then the calculation will be carried out with the Support Vector Machine algorithm to build a model on the Linear, Polynomial, Radial Basis Function (RBF) and Sigmoid kernel.

The SVM method allows computations for linear problems by applying mathematical transformations to the learning space using kernel functions[12]. The SVM method has a central concept in data classification: determining the best hyperplane to provide distance or separation between two predetermined classes[13]. SVM works to find the optimal hyperplane that provides a space or split between the two hyperplane classes with the maximum margin. The distance between the data points closest to the hyperplane is claimed as the margin. The support vector is the point closest to the hyperplane. The linear classification hyperplane can be denoted:

$$f(x)= w\ T\ x + bi \qquad (3)$$

so that the equation is obtained:

$$[(w\ T\ .xi) + b] > 1 \text{ untuki } y_i = +1i \qquad (4)$$

$$[(w\ T\ .xi) + b] < -1 \text{ untuki } y_i = -1i \qquad (5)$$

Where x_i = training data set, i = 1,2,......n and y_i = class label of x_i. To get the best hyperplane, one looks for a hyperplane located in the middle between the two class boundaries. This can be done by maximizing the margin or distance between two sets of objects from different classes.

- Evaluation

Evaluation is done using K-Fold Cross-validation to predict the failure rate. The training data is divided into several parts with the same ratio, then the error rate is calculated in parts, and then all the error rates are averaged to get the total error rate. This study used the 10-fold cross-validation test method, which will repeat the test 10 times. The measurement result is the average value of 10 times of testing due to the output of various extensive experiments and theoretical evidence. This shows that 10 fold cross-validation is the best choice to get accurate validation results.

- Testing

The testing stage is to measure the success of a system by comparing the implementation's output with the standard criteria set. In general, to evaluate the performance of sentiment analysis, a confusion matrix is used. Evaluation measurements are carried out based on the confusion matrix shown in Table 3.

TABLE 3. CONFUSION MATRIX

No	Aspect	Equation
1	Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
2	Precision	$\frac{TP}{TP+FP}$
3	Recall	$\frac{TP}{TP+FN}$
4	F1-Score	$2\times\frac{Precision\times Recall}{Precision+Recall}$

3. RESULTS AND DISCUSSIONS

3.1 Data *Crawling*

The data collectionused in this study is a collection of tweet data obtained using the Twitter API stream provided by Twitter using the python programming language. In this source, the keywords "Covid 19 Vaccination", "Covid Vaccines", and "Sinovac Covid Vaccines" are also written. By doing two crawling stages, the first is 3000 and the second is 2000, the total data obtained is 5000. All downloaded tweets are stored in a .csv documentforfurtherprocessing.Examples of crawled data can be seen in table 4.

Table 4. Example of crawled data

Di Bali adalayananVaksin Drive Thru ya? Denger-dengersihpertama di Asia Tenggara cuy.\n\n Grab Vaccine Center namanya, hasilkerjasamapemerintahdgn Grab & Good Doctor \n\nSmgpariwisata di Bali bisakembalisepertisediakala, yuhuu?#sukseskanvaksinasi #kitavscorona https://t.co/Ab0kKLuJr2
RT @triwul82: Vaksinasimandiri oleh perusahaanakhirnyadiizinkanpemerintah. Bahkanpemerintahmemberinyanamavaksinasi gotong
DPR Minta PemerintahDukungVaksin Covid-19 BuatanDalam Negeri #LengkapCepatBeritanya #BeritaTerkini #Berita #News #BeritaNasional . https://t.co/DhJlxoncn7

3.2 Preprocessing Data

Data preprocessing is the stage of preparing unstructured text data into structured data that is ready to be used for the next process. Preprocessing consists of five stages: cleansing, tokenization, case folding, removing stop words and stemming.

A. Cleansing

Cleansing is done to remove the delimiter comma (,), period (.), all punctuation marks, numbers in tweets and some typical components commonly found in tweets, namely username (@username), URL, HTML characters, and hashtag(#) because does not have any influence in the sentiment analysis process, then these components will be eliminated with the aim of reducing noise. The results of the data after going through the cleansing process can be seen in table 5.

Table 5. The results of the data after going through the cleansing process

Before	After
Di Bali adalayananVaksin Drive Thru ya? Denger-dengersihpertama di Asia Tenggara cuy.\n\n Grab Vaccine Center namanya, hasilkerjasamapemerintahdgn Grab & Good Doctor \n\nSmgpariwisata di Bali bisakembalisepertisediakala,	Di Bali adalayananVaksin Drive Thru yaDengerdengersihpertama di Asia Tenggara cuy Grab Vaccine Center namanyahasilkerjasamapemerintahdgn Grab Good Doctor Smgpariwisata di Bali bisakembalisepertisediakalayuhuusukseskanvaksinasikitavscorona

yuhuu?#sukseskanvaksinasi #kitavscorona https://t.co/Ab0kKLuJr2	
RT @triwul82: Vaksinasimandiri oleh perusahaanakhirnyadiizinkanpemerintah . Bahkanpemerintahmemberinyanamavaksinasi gotong	Vaksinasimandiri oleh perusahaanakhirnyadiizinkanpemerintahBahkanpemerintahmemberinyanamavaksinasi gotong
DPR Minta PemerintahDukungVaksin Covid-19 BuatanDalam Negeri #LengkapCepatBeritanya #BeritaTerkini #Berita #News #BeritaNasional . https://t.co/DhJlxoncn7	DPR Minta PemerintahDukungVaksin Covid BuatanDalam Negeri LengkapCepatBeritanyaBeritaTerkiniBerita News BeritaNasional

B. Case Folding

Case folding is done to change the entire size of the letters in the word into a form of the same letter size. Because not all tweets are consistent in the use of font size. Case folding is done by changing words into lower case or lowercase letters.

Table 6. The results of the data after going through the case folding process

Before	After
Di Bali adalayananVaksin Drive Thru yaDengerdengersihpertama di Asia Tenggara cuy Grab Vaccine Center namanyahasilkerjasamapemerintahdgn Grab Good Doctor Smgpariwisata di Bali bisakembalisepertisediakalayuhuusukseskanvaksinasikitavscorona	di baliadalayananvaksin drive thru yadengerdengersihpertama di asiatenggaracuy grab vaccine center namanyahasilkerjasamapemerintahdgn grab good doctor smgpariwisata di balibisakembalisepertisediakalayuhuusukseskanvaksinasi kitavscorona
Vaksinasimandiri oleh perusahaanakhirnyadiizinkanpemerintahBahkanpemerintahmemberinyanamavaksinasi gotong	vaksinasimandiri oleh perusahaanakhirnyadiizinkanpemerintahbahkanpemerintahmemberinyanamavaksinasi gotong
DPR Minta PemerintahDukungVaksin Covid BuatanDalam Negeri LengkapCepatBeritanyaBeritaTerkiniBerita News BeritaNasional	dprmintapemerintahdukungvaksin covid buatandalam negeri lengkapcepatberitanyaberitaterkiniberita news beritanasional

C. Tokenization

Tokenization is a process that is carried out to separate a row of words in a sentence, paragraph or page into tokens or single word pieces or termmed words. At the same time, tokenization also removes certain characters that are considered as punctuation marks.

Table 7. The results of the data after going through the tokenization process

Before	After
di baliadalayananvaksin drive thru yadengerdengersihpertama di asiatenggaracuy grab vaccine center namanyahasilkerjasamapemerintahdgn grab good doctor smgpariwisata di balibisakembalisepertisediakalayuhuusukseskanvaksinasikitavscorona	[di, bali, ada, layanan, vaksin, drive, thru, ya, denger, denger, sih, pertama, di, asia, tenggara, cuy, grab, vaccine, center, namanya, hasil, kerjasama, pemerintah, dgn, grab, good, doctor, smg, pariwisata, di, bali, bisa, kembali, seperti, sediakala, yuhuu, sukseskanvaksinasi, kitavscorona,]
vaksinasimandiri oleh perusahaanakhirnyadiizinkanpemerintahbahkanpemerintahmemberinyanamavaksinasi gotong	[vaksinasi, mandiri, oleh, perusahaan, akhirnya, diizinkan, pemerintah, bahkan, pemerintah, memberinya, nama, vaksinasi, gotong,]
dprmintapemerintahdukungvaksin covid buatandalam negeri lengkapcepatberitanyaberitaterkiniberita news beritanasional	[dpr, minta, pemerintah, dukung, vaksin, covid, 19, buatan, dalam, negeri, lengkapcepatberitanya, beritaterkini, berita, news, beritanasional,]

D. Removing stop words

Stopword stage, is the process of removing words that are considered unimportant and have no effect on the categorization process. Examples are conjunctions such as 'and', 'or', 'at', 'to', and so on.

Table 8. The results of the data after going through the removing stop words process

Before	After
[di, bali, ada, layanan, vaksin, drive, thru, ya, denger, denger, sih, pertama, di, asia, tenggara, cuy, grab, vaccine, center, namanya, hasil, kerjasama, pemerintah, dgn, grab, good, doctor, smg, pariwisata, di, bali, bisa, kembali, seperti, sediakala, yuhuu, sukseskanvaksinasi, kitavscorona,]	[bali, layanan, vaksin, drive, thru, ya, denger, denger, sih, asia, tenggara, cuy, grab, vaccine, center, namanya, hasil, kerjasama, pemerintah, dgn, grab, good, doctor, smg, pariwisata, bali, sediakala, yuhuu, sukseskanvaksinasi, kitavscorona,]
[vaksinasi, mandiri, oleh, perusahaan, akhirnya, diizinkan, pemerintah, bahkan, pemerintah, memberinya, nama, vaksinasi, gotong,]	[vaksinasi, mandiri, perusahaan, diizinkan, pemerintah, pemerintah, memberinya, nama, vaksinasi, gotong,]
[dpr, minta, pemerintah, dukung, vaksin, covid, buatan, dalam, negeri, lengkapcepatberitanya, beritaterkini, berita, news, beritanasional,]	[dpr, pemerintah, dukung, vaksin, covid, buatan, negeri, lengkapcepatberitanya, beritaterkini, berita, news, beritanasional,]

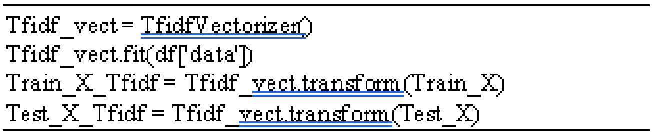
E. Stemming

Stemming stage, is the process of changing words with affixes into stems (basic words) from stop words.

Table 9. The results of the data after going through the stemming process

Before	After
balilayananvaksin drive thru yadengerdengersihasiatenggaracuy grab vaccine center namanyahasilkerjasamapemerintahdgn grab good doctor smgpariwisatabalisediakalayuhuusukseskanvaksinasikitavscorona	balilayanvaksin drive thru yadengerdengersihasiatenggaracuy grab vaccine center namahasilkerjasamaperintahdgn grab good doctor smgpariwisatabalsediakalayuhuusukseskanvaksinasikitavscorona
vaksinasimandiriperusahaandiizinkanpemerintahpemerintahmemberinyanamavaksinasi gotong	vaksinasimandiriusahaizinperintahperintahberinamavaksinasi gotong
dprpemerintahdukungvaksin covid buatan negeri lengkapcepatberitanyaberitaterkiniberita news beritanasional	dprperintahdukungvaksin covid buat negeri lengkapcepatberitanyaberitaterkiniberita news beritanasional

3.3 Implementation iTF-IDF



```
Tfidf_vect = TfidfVectorizer()
Tfidf_vect.fit(df['data'])
Train_X_Tfidf = Tfidf_vect.transform(Train_X)
Test_X_Tfidf = Tfidf_vect.transform(Test_X)
```

Figure 2. Source Code of TF-IDF

In figure 8, the first line of code serves to create the Tfidf_vect variable containing TfidfVectorizer(). The variable will be applied to the data frame with the 'data' column containing the tweet text in the second line. In the third and fourth lines, the code is useful for creating new variables, namely Train_X_Tfidf and Test_X_Tfidf, which contain data that has been transformed using TF-IDF [19].

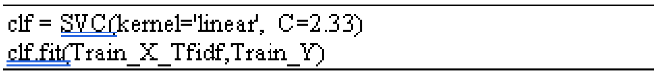
3.4 Modeling

At this stage, the Support Vector Machine (SVM) algorithm classification uses different kernels, namely: Linear, Polynomial, RBF, and Sigmoid. The SVM method requires a C parameter, and specifically, the RBF kernel requires a gamma parameter. Parameters C and Gamma refer to previous research [21]. Parameters C and gamma can be seen in Table X.

TABLE 10. CONFUSION MATRIX

C	Gamma
2.33	0.45
2.25	0.46
2.13	0.50
1.63	1.08

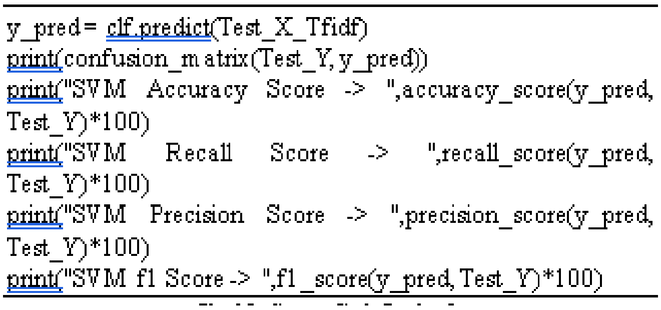
1. Linear



```
clf = SVC(kernel='linear', C=2.33)
clf.fit(Train_X_Tfidf,Train_Y)
```

Figure 3. Source Code of Linear Kernel

In Figure 3, the first line of the code functions to create a clf variable containing SVC (Support Vector Classifier) with a linear kernel parameter and C parameter obtained by finding the best parameter by experimenting with the result C=2.33.



```
y_pred= clf.predict(Test_X_Tfidf)
print(confusion_matrix(Test_Y,y_pred))
print("SVM Accuracy Score -> ",accuracy_score(y_pred,
Test_Y)*100)
print("SVM Recall Score -> ",recall_score(y_pred,
Test_Y)*100)
print("SVM Precision Score -> ",precision_score(y_pred,
Test_Y)*100)
print("SVM f1 Score -> ",f1_score(y_pred,Test_Y)*100)
```

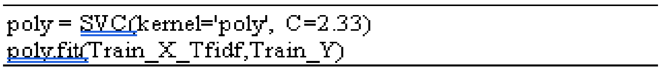
Figure 4. Source Code of Linear Kernel Prediction

In figure 4, the first line of code functions to create a variable for y_pred using testing data, then the second line displays a confusion matrix. The third line to the end of the line will display a report of the resultsofthesepredictions.

TABLE 11. Linear Result

SVM Accuracy Score	87.6
SVM Recall Score	84.2
SVM Precision Score	40.5
SVM F1 Score	54.7

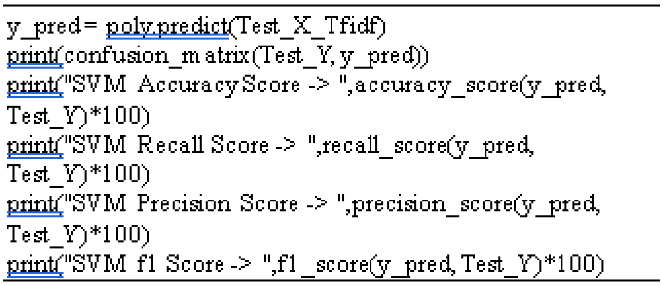
2. Polynomial



```
poly = SVC(kernel='poly', C=2.33)
poly.fit(Train_X_Tfidf,Train_Y)
```

Figure 5. Source Code of Polynomial Kernel

In figure 5, the first line of code functions to create a poly variable that contains SVC (Support Vector Classifier) with a kernel parameter, namely polynomial. Parameter C is obtained by finding the best parameter by experimenting with the result C=2.33.



```
y_pred= poly.predict(Test_X_Tfidf)
print(confusion_matrix(Test_Y, y_pred))
print("SVM AccuracyScore -> ",accuracy_score(y_pred,
Test_Y)*100)
print("SVM Recall Score -> ",recall_score(y_pred,
Test_Y)*100)
print("SVM Precision Score -> ",precision_score(y_pred,
Test_Y)*100)
print("SVM f1 Score -> ",f1_score(y_pred, Test_Y)*100)
```

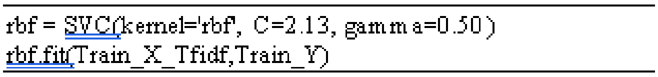
Figure 6. Source Code of Polynomial Prediction

In Figure 6, the first line of code is used to create a variable for y_pred using testing data. The second line will display the confusion matrix. The third line to the end of the line will display a report of the results of these predictions.

TABLE 12. Polynomial Result

SVM Accuracy Score	85.5
SVM Recall Score	81.25
SVM Precision Score	28.1
SVM F1 Score	41.7

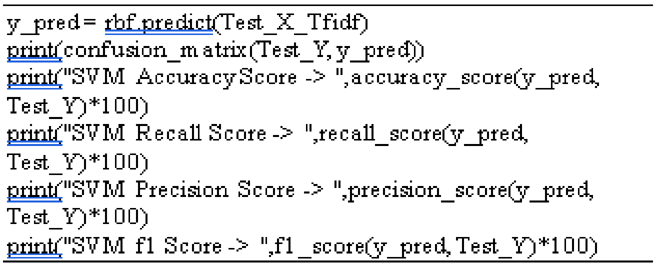
3. RBF



```
rbf = SVC(kernel='rbf', C=2.13, gamma=0.50)
rbf.fit(Train_X_Tfidf,Train_Y)
```

Figure 7. Source Code of RBF Kernel

In figure 7, the first line of the code functions to create a variable rbf that contains SVC(Support Vector Classifier) with the kernel parameter, namely RBF. Parameters C and gamma parameters were obtained by finding the best parameters by conducting experiments, namely C=2.13 & gamma=0.50.



```
y_pred= rbf.predict(Test_X_Tfidf)
print(confusion_matrix(Test_Y, y_pred))
print("SVM AccuracyScore -> ",accuracy_score(y_pred,
Test_Y)*100)
print("SVM Recall Score -> ",recall_score(y_pred,
Test_Y)*100)
print("SVM Precision Score -> ",precision_score(y_pred,
Test_Y)*100)
print("SVM f1 Score -> ",f1_score(y_pred, Test_Y)*100)
```

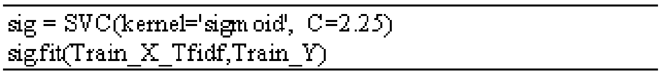
Figure 8. Source Code of RBF Kernel Prediction

In figure 8, the first line of code is used to create a variable for y_pred using testing data. The second line will display the confusion matrix. The third line to the end of the line will display a report of the results of these predictions.

TABLE 13. RBF Result

SVM Accuracy Score	88.8
SVM RecallScore	84.1
SVM Precision Score	48.6
SVM F1-Score	61.4

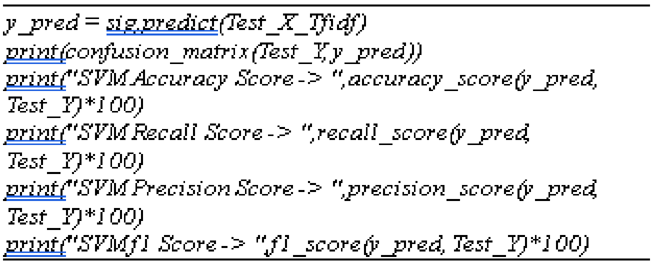
4. Sigmoid



```
sig = SVC(kernel='sigmoid',  C=2.25)
sig.fit(Train_X_Tfidf,Train_Y)
```

Figure 9. Source Code of Sigmoid Kernel

In Figure 9, the first line of code is useful for creating a sig variable that contains SVC (Support Vector Classifier) with the kernel parameter, namely sigmoid. Parameter C is obtained by finding the best parameter by conducting experiments, namely C = 2.25.



```
y_pred = sig.predict(Test_X_Tfidf)
print(confusion_matrix(Test_Y,y_pred))
print("SVM Accuracy Score -> ",accuracy_score(y_pred,
Test_Y)*100)
print("SVM Recall Score -> ",recall_score(y_pred,
Test_Y)*100)
print("SVM Precision Score -> ",precision_score(y_pred,
Test_Y)*100)
print("SVM f1 Score -> ",f1_score(y_pred, Test_Y)*100)
```

Figure 10. Source Code of Sigmoid Kernel Prediction

In figure 10, the first line of code is useful for creating a variable for y_pred using data testing. The second line will display the confusion matrix. The third line to the end of the line will display a report of the results of these predictions.

TABLE 14. Sigmoid Result

SVM Accuracy Score	87
SVM Recall Score	71.6
SVM Precision Score	49.1
SVM F1 Score	58.3

3.5 Model Evaluation

At this stage, the results of the tests carried out on each kernel are presented. Here is a comparison of the accuracy values of each kernel:

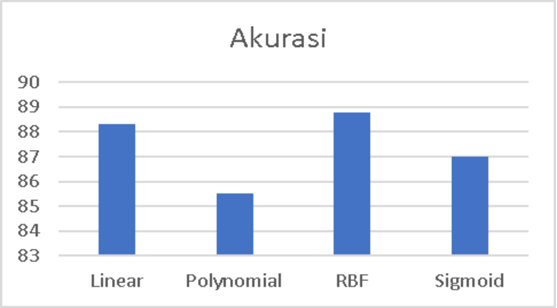


Figure 11. Comparison of accuracy of each kernel

The comparison of accuracy values shows that the RBF has the most superior accuracy level, with a score of 88.8. While the Polynomial got the lowest score of 85.5, the Linear kernel got a score of 88.3, and the sigmoid kernel got a score of 87.

4. CONCLUSION

This study succeeded in applying the Support Vector Machine method to classify data related to the Covid-19 vaccine to produce an analysis of public sentiment. The data used amounted to 5000 obtained from Twitter. The dataset is divided by 80% of the training data and 20% of the test data. The model results show that the RBF kernel has the best accuracy, followed by the Linear, Sigmoid, and Polynomial kernels with 88.8%, 88.3%, 87%, and 85.5% accuracy, respectively.

REFERENCES

[1] H. Vanam and J. Retna Raj R, “Analysis of twitter data through big data based sentiment analysis approaches,” *Mater. Today Proc.*, no. xxxx, 2021, doi: 10.1016/j.matpr.2020.11.486.

[2] S. Styawati and K. Mustofa, “A Support Vector Machine-Firefly Algorithm for Movie Opinion Data Classification,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 3, p. 219, 2019, doi: 10.22146/ijccs.41302.

[3] S. Styawati, A. Nurkholis, A. A. Aldino, S. Samsugi, E. Suryati, and R. P. Cahyono, “Sentiment Analysis on Online Transportation Reviews Using Word2Vec Text Embedding Model Feature Extraction and Support Vector Machine (SVM) Algorithm,” *2021 Int. Semin. Mach. Learn. Optim. Data Sci. ISMODE 2021*, pp. 163–167, 2022, doi: 10.1109/ISMODE53584.2022.9742906.

[4] Styawati., N. Hendrastuty, A. R. Isnain, and A. Y. Rahmadhani, “Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine,” *J. Inform. J. Pengemb. IT*, vol. 6, no. 3, pp. 150–155, 2021, [Online]. Available: http://situs.com.

[5] Styawati, Andi Nurkholis, Zaenal Abidin, and Heni Sulistiani, “Optimasi Parameter Support Vector Machine Berbasis Algoritma Firefly Pada Data Opini Film,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 5, pp. 904–910, 2021, doi: 10.29207/resti.v5i5.3380.

[6] R. Tineges, A. Triayudi, and I. D. Sholihati, “Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM),” *J. Media Inform. Budidarma*, vol. 4, no. 3, p. 650, 2020, doi: 10.30865/mib.v4i3.2181.

[7] A. R. Isnain *et al.*, “Comparison of Support Vector Machine and Naïve Bayes on Twitter Data Sentiment Analysis,” *J. Inform. J. Pengemb. IT*, vol. 6, no. 1, pp. 56–60, 2021.

[8] R. Joshi and R. Tekchandani, “Comparative analysis of twitter data using supervised classifiers,” *Proc. Int. Conf. Inven. Comput. Technol. ICICT 2016*, vol. 2016, 2016, doi: 10.1109/INVENTIVE.2016.7830089.

[9] P. M. Kellstedt and G. D. Whitten, *Data Mining: Concepts and Techniques : Concepts and Techniques*. 2018.

[10] A. Rahmansyah, O. Dewi, P. Andini, T. Hastuti, P. Ningrum, and M. E. Suryana, “Membandingkan Pengaruh Feature Selection Terhadap Algoritma Naïve Bayes dan Support Vector Machine,” *Semin. Nas. Apl. Teknol. Inf.*, pp. 1907–5022, 2018.

[11] D. H. Wahid and A. SN, “Peringkasan Sentimen Esktraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 10, no. 2, p. 207, 2016, doi: 10.22146/ijccs.16625.

[12] A. Kowalczyk, “Support Vector Machines Succintctly, Syncfusion,” *E-Book*, vol. 2, no. 2, p. 114, 2017, [Online]. Available: www.syncfusion.com.

[13] M. Ahmad, S. Aftab, and I. Ali, “Sentiment Analysis of Tweets using SVM,” *Int. J. Comput. Appl.*, vol. 177, no. 5, pp. 25–29, 2017, doi: 10.5120/ijca2017915758.