

# Analisis Sentimen Masyarakat Indonesia Terkait Vaksin Covid-19 Pada Media Sosial Twitter Menggunakan Metode Support Vector Machine (SVM)

Muhammad Fadilah Arfat<sup>1</sup>, Styawati<sup>2\*</sup>, Andi Nurkholis<sup>3</sup>, Indra Kurniawan<sup>4</sup>

<sup>1,3</sup>Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia

<sup>2</sup>Sistem Informasi, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia

<sup>4</sup>Teknik Komputer, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia

<sup>1,2,3,4</sup>Jl. ZA. Pagar Alam No.9 -1 I, Labuhan Ratu, Kec. Kedaton, Kota Bandar Lampung, Lampung

email: <sup>1</sup>fadilitas14@gmail.com, <sup>2</sup>styawati@teknokrat.ac.id, <sup>3</sup>andinh@teknokrat.ac.id, <sup>4</sup>indrakurniawan@teknokrat.ac.id

**Abstract** – COVID-19 is a new disease outbreak that has been officially designated as a global pandemic by the World Health Organization (WHO) on March 11, 2020. Seeing the rapid development of COVID-19, the Government of Indonesia has carried out vaccinations that have been carried out since January 13, 2021, this vaccination is prioritized for medical personnel and red zone areas. Since its emergence, there have been many pros and cons regarding the vaccination process and it has also become a trending topic on social media Twitter on January 13, 2021. One of the most widely used social media by Indonesian people is twitter social media. According to We are Social sources in 2020, twitter social media is ranked fifth in the category of social media that is often used with a user percentage of 56% after Youtube, Whatsapp, Facebook as well as Instagram. This shows that there is a huge opportunity for data sources that can be used to find out the positive and negative sentiments of the related community, which is useful for interested parties to carry out evaluations. So that it can see how many people agree and disagree. If the percentage of people who disagree is more, the government must do better socialization so that people can better understand and not feel afraid of the vaccine. This study aims to find out how public sentiment is about the government's policies regarding the COVID-19 vaccine using the Support Vector Machine method. by extracting the tf-idf feature and comparing the kernels contained in the SVM, including Linear, RBF, Polynomial, and Sigmoid. With tests that will later see how the values of accuracy, precision, recall and F1-Score are.

**Keywords** – Sentiment Analysis, Twitter Social Media, SVM, Vaccines, Covid-19

**Abstrak** – COVID-19 merupakan wabah penyakit baru yang telah ditetapkan secara resmi sebagai pandemi global oleh World Health Organization (WHO) pada tanggal 11 Maret 2020 lalu. Melihat pesatnya perkembangan COVID-19, Pemerintah Indonesia telah melakukan vaksinasi yang telah dilaksanakan sejak 13 Januari 2021 lalu, vaksinasi ini diprioritaskan untuk tenaga medis dan wilayah zona. Sejak kemunculannya banyak

tanggal 13 Januari 2021. Salah satu media sosial yang paling banyak digunakan oleh masyarakat Indonesia adalah media sosial twitter, Menurut sumber We are Social tahun 2020, media sosial twitter berada di peringkat kelima dalam kategori media sosial yang sering digunakan dengan jumlah presentase pengguna sebesar 56% setelah Youtube, Whatsapp, Facebook dan juga Instagram. Hal ini menunjukkan adanya peluang sumber data yang sangat besar yang dapat dimanfaatkan untuk mengetahui sentimen positif dan negatif masyarakat terkait, yang bermanfaat bagi pihak berkepentingan agar dapat melakukan evaluasi. Sehingga dapat melihat seberapa masyarakat yang setuju dan tidak setuju, Jika persentase masyarakat yang tidak setuju lebih banyak maka pemerintah harus melakukan sosialisasi yang lebih baik agar masyarakat dapat lebih mengerti dan tidak merasa takut terhadap vaksin tersebut. Penelitian ini bertujuan untuk mengetahui bagaimana sentiment publik terhadap kebijakan yang akan dilakukan pemerintah mengenai vaksin COVID-19 menggunakan metode Support Vector Machine dengan ekstraksi fitur tf-idf serta membandingkan kernel yang terdapat pada SVM, antara lain Linear, RBF, Polynomial, dan Sigmoid. Dengan pengujian yang nantinya akan dilihat bagaimana nilai accuracy, precision, Recall dan F1-Score.

**Kata Kunci** – Analisis Sentimen, SVM, Twitter, Vaksin, Covid-19

## I. PENDAHULUAN

COVID-19 merupakan wabah penyakit baru yang telah ditetapkan secara resmi sebagai pandemi global oleh World Health Organization (WHO) pada tanggal 11 Maret 2020 lalu [1]. Di Indonesia COVID-19 masuk pertama kali yaitu pada tanggal 2 Maret 2020 lalu, yang menjangkit 2 orang Warga Negara Indonesia asal Depok, Jawa Barat [2]. Berawal dari kasus tersebut, jumlah kasus masyarakat Indonesia yang terjangkit virus korona semakin bertambah setiap harinya, hingga per tanggal 13 Februari 2021, Berdasarkan data yang diambil dari World Health Organization (WHO) tercatat sebanyak lebih dari 1,2 juta kasus dengan kasus kematian sebanyak 32,656 ribu jiwa.

Salah satu media sosial yang paling banyak digunakan oleh masyarakat Indonesia adalah Twitter, Media sosial twitter berada di peringkat kelima dalam kategori media sosial yang sering digunakan dengan jumlah presentase pengguna sebesar 56% setelah Youtube, Whatsapp, Facebook serta juga Instagram [3]. Twitter digunakan oleh semua kalangan untuk memberikan tanggapan netral, positif bahkan tanggapan negatif terhadap suatu trending topic, Aktivitas sosial

\*) penulis korespondensi: Styawati

Email: styawati@teknokrat.ac.id

pro dan kontra mengenai proses vaksinasi tersebut dan juga sempat menjadi trending topik di media social twitter pada

tersebut dipercaya sangat memudahkan seorang dalam berdiskusi, berbisnis, berkomentar secara bebas [4]. Hal ini menunjukkan adanya peluang sumber data yang sangat besar yang dapat dimanfaatkan untuk mengetahui sentimen positif dan negatif masyarakat terkait vaksinasi COVID-19. Untuk mengetahui sentimen positif dan negatif dapat menggunakan teknik *Text Mining*. Salah satu teknik dari *Text Mining* yaitu SVM [5], *Support Vector Machine (SVM)* merupakan metode klasifikasi *supervised learning* yang memprediksi kelas berdasarkan model atau pola dari hasil proses *training*. Klasifikasi dilakukan dengan mencari *hyperplane* atau garis pembatas yang memisahkan antara suatu kelas dengan kelas lain. Dalam kasus ini garis tersebut berperan memisahkan tweet bersentimen positif dengan tweet bersentimen negatif. SVM melakukan pencarian nilai *hyperlane* yang paling maksimal dengan menggunakan *support vector* dan nilai *margin*.

Analisis sentimen semakin maju dan banyak dibahas dalam berbagai jurnal, penelitian salah satunya yang dilakukan oleh Windasari, dkk. 2017 dengan judul "*Sentiment Analysis on Twitter Posts: An analysis of Positive or Negative Opinion on GoJek*". Penelitian ini ditujukan untuk mengusulkan system yang bisa mendeteksi sentimen publik dengan menggunakan tweet opini pengguna Twitter tentang layanan transportasi online terutama GoJek menggunakan metode SVM. Hasil dari pengujian yang dilakukan membuktikan tingkat akurasi dari metode SVM sebesar 86%, *prediction error rate* 14%, tingkat prediksi yang benar untuk sentimen positif 100%, dan tingkat prediksi yang benar untuk sentimen negatif 67,44% [6]. Penelitian yang dilakukan oleh Evita Fitri (2020) dengan judul "Analisis Sentimen Aplikasi Ruang Guru Menggunakan Metode Support Vector Machine dan Naive Bayes Classifier" juga membahas bagaimana sentimen pengguna terhadap ulasan aplikasi ruang guru yang di dapat dari Google Play Store dengan batas waktu tertentu. Analisis ini menggunakan metode SVM dan Naive Bayes sekaligus bertujuan untuk membandingkan metode mana yang memiliki akurasi terbaik. Hasil perbandingan metode dalam proses klasifikasi juga membuktikan bahwa ternyata metode SVM memiliki tingkat akurasi yang lebih baik dari [7].

Berdasarkan penelitian tersebut maka penelitian ini ingin melakukan klasifikasi data Twitter terkait vaksin COVID-19 selain itu melihat performa dari metode Support Vector Machine untuk data yang dipakai serta membandingkan kernel yang terdapat pada SVM untuk melihat manakah kernel terbaik.

## II. PENELITIAN YANG TERKAIT

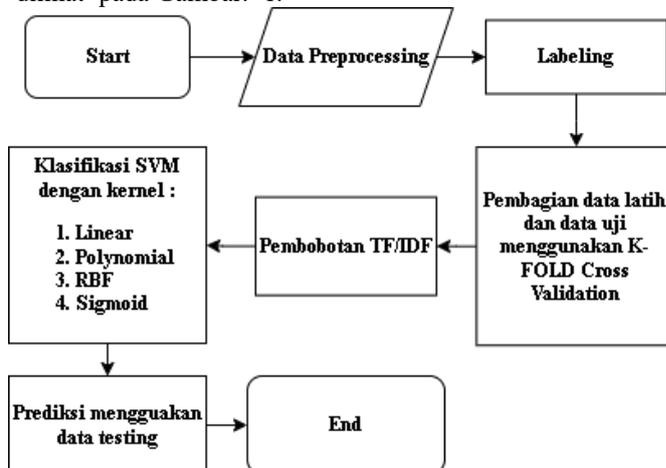
Penelitian yang dilakukan oleh Pangestu tahun 2019 menggunakan objek opini mengenai partai politik Indonesia. Penelitian ini bertujuan untuk memakai metode support vector machine (SVM) untuk mengklasifikasikan sikap politik partai politik Indonesia. yang mempunyai output accuracy sebesar 86% dan pengujian dengan 10-Fold Cross Validation rata-rata accuracy sebesar 71% serta error rate sebanyak 29% [8]. Sedangkan pada penelitian yang dilakukan Rian (2020)

dalam penelitiannya tentang indihome berdasarkan twitter menggunakan metode SVM, diperoleh nilai positif sebanyak 18,4% dengan hasil negatif sebanyak 81,6% dan keakuratan sebesar 87% dengan ketepatan antara output perkiraan menggunakan data sebenarnya presisi sebesar 86%, tingkat keberhasilan sistem dalam memprediksi suatu data (recall) sebesar 95%, tingkat error rate semua data yang diprediksi (error rate) sebesar 13%, selain itu untuk nilai perbandingan rata-rata precision serta recall (f1-score) adalah sebesar 90% [9]. Pada penelitian yang dilakukan oleh Imelda (2015) dalam penelitian menerapkan metode Support Vector Machine pada klasifikasi Tweet dengan kernel Radial Basis Function agar tweet yang ada tidak tercampur antara iklan dan non-iklan. Parameter yang digunakan berupa nilai C dan  $\gamma$  yang kemudian dihasilkan nilai akurasi tertinggi yaitu 99,12% sehingga dapat diimplementasikan untuk membantu pengguna mengelola tweet, terutama tweet iklan [10]. Pada penelitian yang dilakukan oleh Dinar Ajeng Kristianti pada tahun 2019 yang berjudul "*Prediction of Indonesia Presidential Election Results for the 2019-2024 Period Using Twitter Sentiment Analysis*", Berhasil menerapkan SVM untuk melakukan prediksi calon presiden dengan tingkat akurasi sebesar 80,05% untuk pasangan nomor urut 1 dan 82,35% untuk pasangan nomor urut 2 [11]. Sedangkan pada penelitian Rohit Joshi pada tahun 2021 yang berjudul "Comparative Analysis Of Twitter Data Using Supervised Classifiers", Membandingkan dua algoritma *supervised learning Naive Bayes* dan SVM dengan hasil akurasi dimana SVM lebih unggul sebesar 83% dari *Naive Bayes* yang hanya mendapat 73% tingkat akurasi [12].

## III. METODE PENELITIAN

### A. Alur Penelitian

Pada Penelitian ini terdapat tahapan yang dilakukan untuk menyelesaikan ipenelitian, alur penelitian dapat dilihat pada Gambar. 1.



Gambar. 1 Alur penelitian

### B. Preprocessing

Dalam Preprocessing data, kita perlu melakukan pembersihan data dengan tujuan supaya data digunakan dapat digunakan pada tahap selanjutnya. Adapun tahapan yang dilakukan pada penelitian ini adalah dengan melakukan *cleansing*, *case folding*, *tokenizing*, *filtering*, *stemming* [13].

C. Labeling

Pada tahap ini pelabelan akan dilakukan manual yang mana akan dibantu oleh Bapak Yusniar seorang ahli bahasa Indonesia. Dataset tersebut terbagi kedalam 2 kategori yaitu opini positif serta opini negative.

D. Pembagian Data

Pada tahapan ini data sebanyak 5000 akan dibagi menjadi dua,yaitu data latih serta data uji dengan rasio data latih sebesar 80% serta data uji sebesar 20%. Data latih digunakan untuk melatih algoritma dan pengembangan model, sedangkan data uji digunakan sebagai acuan dan menentukan kinerja dari algoritma atau model yang dilatih sebelumnya. [14].

E. Pembobotan Data

Tahap pembobotan merupakan tahap dimana pemberian bobot dalam setiap kata dengan menggunakan perhitungan Term Frequency dan Inverse Document [15]. Rumus IDF seperti pada rumus (1):

$$idf = \log D/df_i \tag{1}$$

Dengan:

D = jumlah dokumen

df<sub>i</sub> = kemunculan term dari D

Setelah mendapatkan IDF, maka dapat dilakukan perhitungan TF-IDF(w) untuk setiap term dengan persamaan (2)

$$W_{ij} = tf_{ij} + (\log D/df_i) \tag{2}$$

Dengan:

W<sub>ij</sub> = bobot kata/term; terhadap dokumen d<sub>i</sub>

tf<sub>ij</sub> = jumlah kemunculan kata/term; dalam dokumen d<sub>i</sub>

Contoh Dapat dilihat pada Tabel 1 data training.

TABEL I  
DATA TRAINING

KODE	TEKS
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

Berikut merupakan hasil dari perhitungan TF-IDF, Hasil perhitungan dapat dilihat pada tabel 2.

TABEL 2  
TF-IDF

W <sub>dt</sub> = 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	0,39794	0	0,39794
0	0	0	0	0,69897
0	0	0	0	0,69897

F. Klasifikasi

Data yang telah dilakukan preprocessing dan juga di ubah nilai dari teks menjadi bobot dengan TF-IDF (Term Frequency – Invert Document Frequency) akan dilakukan perhitungan dengan menggunakan algoritma untuk membentuk suatu model yang bisa mengklasifikasi data tahap berikut disebut dengan Modelling. Model dibangun dengan algoritma Support Vector Machine yang diimplementasikan menggunakan bahasa pemrograman Python yang ada pada Jupyter Notebook Kemudian akan dilakukan perhitungan dengan algoritma Support Vector Machine untuk membangun model pada kernel Linear, Polynomial, Radial Basis Function (RBF), serta Sigmoid.

Metode SVM memungkinkan perhitungan untuk masalah linier menerapkan transformasi matematis ke ruang belajar menggunakan fungsi kernel [16]. Metode SVM memiliki konsep sentral dalam klasifikasi data, yaitu menentukan hyperplane terbaik untuk memberikan jarak atau pemisahan antara dua kelas yang telah ditentukan [17]. Cara kerja SVM adalah dengan mencari hyperplane optimal yang memberikan spasi atau pemisah antara dua kelas. Hyperplane dengan margin maksimum. Jarak antara titik data terdekat dari ke hyperplane diklaim sebagai margin. Support vector adalah titik yang paling dekat dengan hyperplane [18].

Menurut Santosa, Hyperplane klasifikasi linier dapat dinotasikan:

$$f(x) = w T x + b \tag{3}$$

sehingga menurut Vapnik dan Cortes diperoleh persamaan:

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

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

Dengan  $x_i$  = himpunan data training ,  $i = 1,2,\dots,n$  dan  $y_i$  = label kelas dari  $x_i$  Untuk mendapatkan hyperplanei terbaik adalah dengan mencari hyperplanei yang terletak di tengah-tengah antara dua bidang pembatas kelas dan untuk mendapatkan hyperplane terbaik itu, sama dengan memaksimalkan margin atau jarak antara dua seti objek dari kelas yang berbeda [19].

G. Evaluasi

Evaluasi dilakukan menggunakan *K-Fold Cross validation*, *K-Fold Cross validation* adalah metode paling umum yang dilakukan untuk memprediksi tingkat kegagalan. Data latih dibagi menjadi beberapa bagian dengan perbandingan yang sama, kemudian tingkat kesalahan dihitung perbagian, dan kemudian semua tingkat kesalahan dirata-ratakan untuk mendapatkan tingkat kesalahan total. Pada penelitian ini penulis menggunakan metode pengujian *10 fold cross validation* yang akan mengulang pengujian se banyak 10 kali serta hasil pengukuran adalah nilai rata-rata dari 10 kali pengujian, karena output dari berbagai percobaan yang ekstensif serta pembuktian teoritis, menunjukkan bahwa *10 fold cross validation* adalah pilihan terbaik untuk mendapatkan hasil validasi yang akurat.

H. Pengujian

Tahap pengujian adalah tahapan upaya untuk mengukur keberhasilan suatu sistem dengan membandingkan output perolehan implementasi dengan kriteria standar yang telah ditetapkan [20] Umumnya untuk mengevaluasi hasil implementasi pada sentimen analisis memakai confusion matrix. Pengukuran evaluasi dilakukan berdasarkan confusion matrix yang diperlihatkan pada tabel 3.

TABEL 3  
CONFUSION MATRIX

No	Jenis	Model
1	Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$
2	Precision	$\frac{TP}{TP + FP}$
3	Recall	$\frac{TP}{TP + FN}$
4	F1-Score	$2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$

IV. HASIL DAN PEMBAHASAN

A. Crawling Data

Pengumpulan data yang dipakai dalam penelitian ini merupakan kumpulan data tweet yang diperoleh menggunakan stream twitter API yang disediakan oleh twitter dengan menggunakan bahasa pemograman python. Dalam source ini juga dituliskan kata kunci “Vaksinasi Covid 19”, “Vaksin Covid”, serta “Vaksin

Covid Sinovac”. Dengan melakukan dua tahap crawling yang pertama sebanyak 3000 dan yang kedua sebanyak 2000 total seluruh data yang didapatkan sebanyak 5000 data. Seluruh tweet yang telah diunduh disimpan kedalam dokumen .csv untuk dilakun proses selanjutnya.

```
for tweet in tweepy.Cursor(api.search,
    q = "vaksincovid sinovac",
    since = "2021-05-29",
    until = "2021-06-04",
    lang = "id").items(3000):
    csvWriter.writerow([tweet.created_at, tweet.text.encode('utf-8')])
print(tweet.created_at, tweet.text)
csvFile.close()
```

Gambar 2. Source Code Crawling

B. Preprocessing

*Preprocessing* merupakan tahapan menyiapkan data yang berupa teks yang tadinya tidak terstruktur akan menjadi data terstruktur yang akan siap pakai untuk proses berikutnya. *Preprocessing* terdiri dari lima tahap yaitu *cleansing*, *tokenisasi*, *case folding*, penghilangan *stopword*, serta *stemming*. Contoh untuk tahapan *cleansing* dapat dilihat pada gambar 3.

```
def remove_punct(text):
    text = re.sub("([A-Za-z0-9]+)|([\^0-9A-Za-z \t])|(\w+:\w\S+)", "", text)
    text = re.sub('RT[\s]+', '', text)
    text = re.sub('/n', '', text)
    text = re.sub('((www\.[^\s]+)|(https?://[^\s]+)|(http?://[^\s]+))!',
    ',text)
    text = re.sub(' +', '', text)
    return text
df['ISI'] = df['isi'].apply(lambda x: remove_punct(x))
```

Gambar 3. Source Code Cleansing

Pada gambar 3 baris ke 1 akan membuat fungsi “remove\_punch” yang berisi fungsi untuk menghilangkan tanda baca, angka, spasi berlebih, URL, serta hastag. Serta pada baris ke 8 berisi perintah untuk mengeksekusi fungsi tersebut, Selanjutnya tahapan *Case Folding* bisa dilihat pada gambar 4.

```
def lowercase(text):
    return text.lower()
df['ISI'] = df['isi'].apply(lambda x: lowercase(x))
```

Gambar 4. Source Code Case Folding

Pada gambar 4 baris pertama code tersebut berguna untuk membuat fungsi yang akan mengubah seluruh huruf besar ke huruf kecil, Serta pada baris ke 3 akan dieksekusi ke dataframe, Selanjutnya pada tahapan *Tokenization* dapat dilihat pada gambar 5.

```
def tokenization(text):
    text = re.split('\W+', text)
    return text
df['TOKENIZATION'] = df['ISI'].apply(lambda x:
tokenization(x))
```

Gambar 5. Source Code Tokenization

Pada gambar 5 baris pertama membuat fungsi yang akan membagi kalimat menjadi perkata dengan perintah seperti pada baris ke 2, Serta pada baris ke 4 akan di eksekusi ke dataframe, Pada tahap Stop Word Removal bisa dilihat pada gambar 6.

```
stopword = nltk.corpus.stopwords.words('indonesian')
def remove_stopwords(text):
    text = [word for word in text if word not in stopwords] return
    text
df['STOP_REMOVAL'] =
df['TOKENIZATION'].apply(lambda x:
remove_stopwords(x))
```

Gambar 6. Source Code Stop Word Removal

Pada gambar 6 baris pertama kode tersebut berfungsi untuk memanggil library stopwords berbahasa indonesia dari nltk, Pada baris ke 2 kode tersebut akan membuat fungsi untuk menghilangkan stopwords yang ada pada data, Terakhir pada baris ke 5 akan mengeksekusi ke dataframe, Selanjutnya tahap Stemming bisa dilihat pada gambar 7.

```
factory = StemmerFactory()
stemmer = factory.create_stemmer()
def stemming(text):
    return stemmer.stem(text)
df['STEMMING'] = df['STOP_REMOVAL'].apply(lambda x:
stemming(x))
```

Gambar 7. Source Code Stemming

Pada gambar 7 baris pertama kode tersebut akan membuat variabel factory serta pada baris kedua membuat variabel stemmer. Kemudian pada baris ke 3 kode tersebut membuat fungsi stemming yg akan mengubah kata menjadi kata dasar, Serta yang terakhir pada baris ke 5 kode tersebut berguna untuk mengeksekusi fungsi tersebut ke dataframe.

### C. Implementasi TF-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)
```

Gambar 8. Source Code TF-IDF

Pada gambar 8 baris pertama kode tersebut berfungsi membuat variabel Tfidf\_vect berisi TfidfVectorizer(), Kemudian pada baris kedua variabel tersebut akan di terapkan pada dataframe dengan kolom 'data' yang berisi teks tweet, Pada baris tiga serta empat kode tersebut berguna untuk membuat variabel baru yaitu Train\_X\_Tfidf serta Test\_X\_Tfidf yang berisi data yang sudah di

transformasi menggunakan tfidf [19].

### D. Pemodelan

Pada tahap ini melakukan pengklasifikasian dengan algoritma *Support Vector Machine (SVM)* menggunakan kernel berbeda yaitu: *Linear, Polynomial, RBF*, serta *Sigmoid*. Pada metode *Support Vector Machine (SVM)* membutuhkan parameter C serta untuk kernel RBF membutuhkan parameter gamma, Parameter C serta Gamma yang digunakan merujuk pada penelitian yang dilakukan oleh Styawati [21] Parameter C serta gamma dapat dilihat pada tabel 4.

Tabel 4

Confussion 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)
```

Gambarr 9. Source Code Linear

Pada gambar 9 baris pertama kode tersebut berfungsi membuat variabel clf yang berisi *SVC(Support Vector Classifier)* dengan parameter kernel yaitu linear serta parameter C yang didapat dengan mencari paramter terbaik dengan cara melakukan *trail and error* yang hasilnya 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)
```

Gambar 10. Source Code Predict Linear

Pada gambar 10 baris pertama kode tersebut berfungsi membuat variabel untuk y\_pred dengan menggunakan data testing kemudian pada baris kedua akan menampilkan confusion matrix serta pada baris ketiga sampai akhir akan menampilkan laporan hasil dari prediksi tersebut.

Tabel 5  
Hasil Linear

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)
```

Gambar 11. Source Code Polynomial

Pada gambar 11 baris pertama kode tersebut berfungsi membuat variabel poly yang berisi SVC(Support Vector Classifier) dengan parameter kernel yaitu polynomial serta parameter C yang didapat dengan mencari paramter terbaik dengan cara melakukan trail and erroryang hasilnya C=2.33.

```
y_pred= poly.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)
```

Gambar 12. Source Code Predict Polynomial

Pada gambar 12 baris pertama kode tersbut berfungsi membuat variabel untuk y\_pred dengan menggunakan data testing kemudian pada baris kedua akan menampilkan confusion matrix serta pada baris ketiga sampai akhir akan menampilkan laporan hasil dari prediksi tersebut.

Tabel 6  
Hasil Polynomial

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)
```

Gambar 13. Source Code RBF

Pada gambar 13 baris pertama kode tersebut berfungsi membuat variabel rbf yang berisi SVC(Support Vector Classifier) dengan parameter kernel yaitu rbf serta parameter C serta parameter gamma yang didapat dengan mencari paramter terbaik dengan cara melakukan trail and error yaitu C=2.13 & gamma=0.50.

```
y_pred= rbf.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)
```

Gambar 14. Source Code Predict RBF

Pada gambar 14 baris pertama kode tersebut berfungsi membuat variabel untuk y\_pred dengan menggunakan data testing kemudian pada baris kedua akan

menampilkan confusion matrix serta pada baris ketiga sampai akhir akan menampilkan laporan hasil dari prediksi tersebut.

Tabel 7  
Hasil RBF

SVM Accuracy Score	88.8
SVM Recall Score	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)
```

Gambar 15. Source Code Sigmoid

Pada gambar 15 baris pertama kode tersebut berguna membuat variabel sig yang berisi SVC(Support Vector Classifier) dengan parameter kernel yaitu sigmoid serta parameter C yang didapat dengan mencari paramter terbaik dengan cara melakukan trail and error yaitu 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)
```

Gambar 16. Source Code Predict Sigmoid

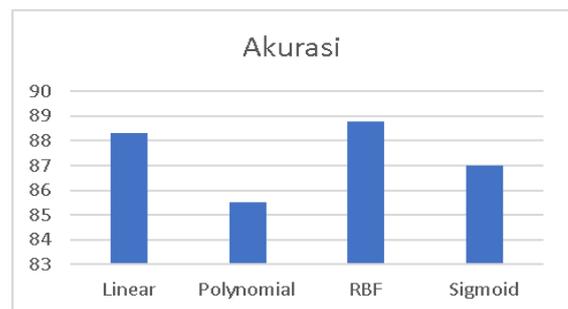
Pada gambar 16 baris pertama kode tersebut berguna untuk membuat variabel untuk y\_pred dengan menggunakan data testing kemudian pada baris kedua akan menampilkan confusion matrix serta pada baris ketiga sampai akhir akan menampilkan laporan hasil dari prediksi tersebut.

Tabel 7  
Hasil Sigmoid

SVM Accuracy Score	87
SVM Recall Score	71.6
SVM Precision Score	49.1
SVM f1 Score	58.3

E. Evaluasi Data

Pada tahap ini akan melihat hasil data yang telah dilakukan pengujian pada setiap kernel. Berikut perbandingan nilai akurasi setiap kernel :



Gambar 17. Perbandingan akurasi setiap kernel

Hasil perbandingan nilai akurasi menunjukkan bahwa Rbf paling unggul tingkat akurasinya dengan *score* 88,8, Sedangkan *Polynomial* mendapatkan *score* terendah yaitu 85,5 serta pada kernel *Linear* mendapatkan *score* 88,3 seserta kernel *sigmoid* mendapat *score* 87.

## V KESIMPULAN

Penelitian ini berhasil mengimplementasikan metode Support Vector Machine untuk mengklasifikasikan analisis sentimen data tweet terkait vaksin Covid-19, Menggunakan 5000 data yang diambil dari twitter. Diantaranya, 80% data merupakan data latih, serta 20% data merupakan data uji, Hasil menunjukkan Kernel rbf memiliki akurasi terbaik, sebesar 88,8%, diikuti oleh kernel Linear sebesar 88,3% serta kernel Sigmoid 87% Untuk kernel dengan tingkat akurasi terendah yaitu Polynomial, Nilai akurasi 85,5%.

## DAFTAR PUSTAKA

- [1] H. A. Rothan and S. N. Byrareddy, "The epidemiology and pathogenesis of coronavirus disease (COVID-19) outbreak," *Journal of Autoimmunity*, vol. 109, 2020. doi: 10.1016/j.jaut.2020.102433.
- [2] S. Syafrida and R. Hartati, "Bersama Melawan Virus Covid 19 di Indonesia," *SALAM: Jurnal Sosial dan Budaya Syar-i*, vol. 7, no. 6, 2020, doi: 10.15408/sjsbs.v7i6.15325.
- [3] D. A. Broniatowski, M. J. Paul, and M. Dredze, "Twitter: Big data opportunities," *Science*, vol. 345, no. 6193, 2014. doi: 10.1126/science.345.6193.148-a.
- [4] H. Vanam and J. Retna Raj R, "Analysis of twitter data through big data based sentiment analysis approaches," *Materials Today: Proceedings*, 2021, doi: 10.1016/j.matpr.2020.11.486.
- [5] D. Alita, S. Priyanta, and N. Rokhman, "Analysis of Emoticon and Sarcasm Effect on Sentiment Analysis of Indonesian Language on Twitter," *Journal of Information Systems Engineering and Business Intelligence*, vol. 5, no. 2, 2019, doi: 10.20473/jisebi.5.2.100-109.
- [6] I. P. Windasari, F. N. Uzzi, and K. I. Satoto, "Sentiment analysis on Twitter posts: An analysis of positive or negative opinion on GoJek," in *Proceedings - 2017 4th International Conference on Information Technology, Computer, and Electrical Engineering, ICITACEE 2017*, 2017, vol. 2018-January. doi: 10.1109/ICITACEE.2017.8257715.
- [7] F. F. Irfani, "ANALISIS SENTIMEN REVIEW APLIKASI RUANGGURU MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE," *JBMI (Jurnal Bisnis, Manajemen, dan Informatika)*, vol. 16, no. 3, pp. 258–266, Feb. 2020, doi: 10.26487/jbmi.v16i3.8607.
- [8] S. Y. Pangestu, Y. Astuti, and L. D. Farida, "Algoritma Support Vector Machine untuk Klasifikasi Sikap Politik terhadap Partai Politik Indonesia," *Jurnal Mantik Penusa*, vol. 3, no. 1, 2019.
- [9] R. Tineges, A. Triayudi, and I. D. Sholihati, "Analisis Sentimen Terhadap Layanan Indihome Berdasarkan Twitter Dengan Metode Klasifikasi Support Vector Machine (SVM)," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 4, no. 3, 2020, doi: 10.30865/mib.v4i3.2181.
- [10] M. T. Imelda A.Muis & Muhammad Affandes, "Penerapan Metode Support Vector Machine ( SVM ) Menggunakan Kernel Radial Basis Function ( RBF ) Pada Klasifikasi Tweet," *Sains, Teknologi dan Industri.UIN Sultan Syarif Kasim Riau*, vol. 12, no. 2, 2015.
- [11] D. A. Kristiyanti, Normah, and A. H. Umam, "Prediction of Indonesia presidential election results for the 2019-2024 period using twitter sentiment analysis," in *Proceedings of 2019 5th International Conference on New Media Studies, CONMEDIA 2019*, Oct. 2019, pp. 36–42. doi: 10.1109/CONMEDIA46929.2019.8981823.
- [12] R. Joshi and R. Tekchandani, "Comparative analysis of twitter data using supervised classifiers," in *Proceedings of the International Conference on Inventive Computation Technologies, ICICT 2016*, 2016, vol. 2016. doi: 10.1109/INVENTIVE.2016.7830089.
- [13] Jiawei Han and M. Kamber, *Data Mining: Concepts and Techniques Second Edition*, vol. 53, no. 9, 2013.
- [14] 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," 2018.
- [15] D. Haryalesmana Wahid, "Peringkasan Sentimen Esktraktif di Twitter Menggunakan Hybrid TF-IDF dan Cosine Similarity," *IJCCS*, vol. 10, no. 2, pp. 207–218, 2016.
- [16] A. Kowalczyk, "Support Vector Machines Succinctly," *Journal of Chemical Information and Modeling*, vol. 53, no. 9, 2017.
- [17] S. Rani and J. Singh, "SENTIMENT ANALYSIS OF TWEETS USING SUPPORT VECTOR MACHINE," *International Journal of Computer Science and Mobile Applications*, vol. 5, 2017.
- [18] N. Hendrastuty, A. Rahman Isnain, and A. Yanti Rahmadhani, "Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine," vol. 6, no. 3, 2021, [Online]. Available: <http://situs.com>
- [19] Styawati, Andi Nurkholis, Zaenal Abidin, and Heni Sulistiani, "Optimasi Parameter Support Vector Machine Berbasis Algoritma Firefly Pada Data Opini Film," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 5, pp. 904–910, Oct. 2021, doi: 10.29207/resti.v5i5.3380.
- [20] "A New Approach for Evaluation of Data Mining Techniques," *International Journal of Computer Science Issues*, vol. 7, no. 5, 2010.
- [21] 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.

