

Perbandingan Metode Naïve Bayes Classifier Dan Support Vector Machine Pada Kasus Analisis Sentimen Terhadap Data Vaksin Covid-19 Di Twitter

Rizki Anom Raharjo¹, I Made Gede Sunarya², Dewa Gede Hendra Divayana³

¹ Ilmu Komputer, rizki.anomraharjo@gmail.com, Universitas Pendidikan Ganesha

² Ilmu Komputer, sunarya@undiksha.ac.id, Universitas Pendidikan Ganesha

³ Ilmu Komputer, divayana@undiksha.ac.id, Universitas Pendidikan Ganesha

Jl. Majapahit 605, (024) 6723456 , Semarang, Jawa Tengah

ARTICLE INFO

Article history:

Received 12 Agustus 2022

Received in revised form 28 Agustus 2022

Accepted 3 September 2022

Available online 1 Desember 2022

ABSTRACT

In the context of the Covid-19 pandemic, the Indonesian government has also asked and encouraged the public to participate in supporting vaccination, but this effort actually has advantages and disadvantages, so that many people express their opinions on social networks, one of which is Twitter. This study aims to determine the results of applying sentiment analysis and measuring the performance of the Naïve Bayes Classifier (NBC) and Support Vector Machine (SVM) algorithms on Covid-19 vaccine data by classifying the data into positive and negative classes. The tweet data obtained is then text preprocessed to optimize data processing. There are 4 stages of text preprocessing including Case Folding, Tokenizing, Filtering, and Stemming. This study examines the performance of the Naïve Bayes Classifier (NBC) and Support Vector Machine (SVM) by adding the TF-IDF (Term Frequency-Inverse Document Frequency) technique which aims to give weight to the relationship of words (terms) in a document. Then do data splitting, namely dividing 80% of training data and 20% of testing data in the hope of getting the model with the best performance and finally visualizing tweet data using Word Cloud so that you can draw a conclusion. The results of the classification of the Covid-19 vaccine tweet data using the Naïve Bayes Classifier algorithm get an accuracy value of 81%, a precision of 80%, a recall of 99%, and an f1-score of 89%, while the Support Vector Machine algorithm obtains an accuracy value of 87% . , precision of 88%, recall of 96%, and f1-score of 92%.

Keywords: Soil Moisture, Water Temperature, Internet of Things, Rice

1. Pendahuluan

Kasus virus covid-19 pertama kali terdeteksi pada bulan Desember 2019 di kota Wuhan, China. Dari hasil data yang bersumber dari Satuan Tugas Penanganan dan Penanggulangan covid-19 dan KPCPEN (<https://covid19.go.id/>), Kasus penduduk Indonesia yang positif virus corona terus meningkat setiap hari hingga Juli 2022, Ada lebih dari 6,1 juta kasus positif dan lebih dari 156.000 kematian. Dampak serius dari covid-19 adalah sistem kesehatan global menghadapi

tantangan dan masalah besar dan dapat memiliki implikasi yang luas bagi ekonomi global. (Liu C, et al, 2020). Dalam menyikapi hal tersebut, Pemerintah Indonesia mengambil langkah untuk mencegah penyebaran penyakit ini dengan cara mengembangkan vaksin. pelaksanaan vaksinasi ini menjadi pembahasan yang menimbulkan pro dan kontra dimasyarakat, sehingga masyarakat banyak yang mengutarakan opininya terkait vaksin covid-19 di media sosial. Contohnya pada media sosial Twitter, vaksin covid-19 sempat menjadi trending topic karena ramai dibahas oleh masyarakat Indonesia. Sehingga hal ini menjadi menarik dan penting diketahui baik buruknya sebuah sentiment atau opini masyarakat Indonesia. Opini masyarakat yang ada di twitter ini akan menjadi dataset untuk dilakukan analisis sentimen. Data yang digunakan merupakan hasil dari Crawling Tweet dengan keyword Vaksin Covid-19. Data yang terkumpul dibagi untuk digunakan sebagai data training dan data testing untuk proses klasifikasi. Pada penelitian ini metode untuk klasifikasi menggunakan Naïve Bayes Classifier dan Support Vector Machine (SVM) untuk mengetahui tingkat akurasi tertinggi.

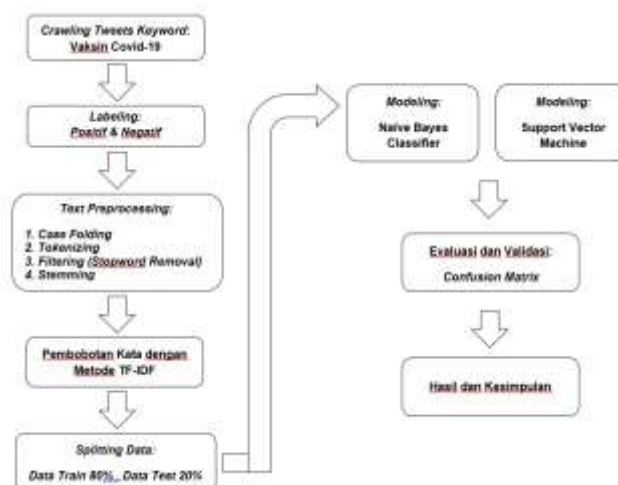
2. Metode Penelitian

Peneliti menggunakan metode pengumpulan data dengan melakukan teknik Crawling Tweet pada media sosial Twitter dengan keyword Vaksin Covid-19. Data yang digunakan dalam penelitian ini adalah hasil crawling pada tanggal 1 April 2021 sampai 31 Agustus 2021 sebanyak 1106 data tweet. Data yang berjumlah 1106 akan dibagi menjadi data training dan data testing dengan perbandingan 80% untuk data training dan 20% untuk data testing (80:20). Setelah dilakukan preprocessing, maka pada penelitian ini terbentuk struktur data teks tweet yaitu berupa kata dasar setiap tweet dan variabel respon yang mengklasifikasi sentimen tweet positif dan negatif.

Tabel 1. Contoh Struktur Data

No	Tweet	Sentimen/Label
1	Alhamdulillah saya sudah di vaksin covid-19	Positif
2	Saya tepat seminggu setelah vaksin AZ positive covid-19 :(Negatif
3	Warga Antusias Ikuti Vaksin COVID-19 Gratis di TSM Cibubur	Positif
4	Ada Dokter Meninggal Meski Telah Divaksinasi Covid-19, Pemerintah Diminta Upayakan Vaksin <i>Booster</i>	Negatif
5	Banyaknya isu yang menyebar di masyarakat membuat sebagian orang merasa ragu untuk melakukan vaksin Covid-19	Negatif

Adapun tahapan penelitian data yang dilakukan dalam penelitian ini bisa dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian Data

Keterangan:

1. Mengambil data tweet menggunakan teknik *Crawling Tweet* dengan bantuan library *snsrape* pada Python. *Keyword* yang digunakan yaitu “Vaksin Covid-19”.
2. Menentukan label atau kelas dari masing-masing sentimen kedalam kelas positif dan negatif oleh anota-tor.
3. Tahap *Text Preprocessing*
Text Preprocessing adalah tahap untuk mengubah data teks yang tidak terstruktur menjadi data terstruktur agar selanjutnya dapat digunakan untuk analisa. Adapun beberapa tahapan *text preprocessing* yaitu :
 - a) *Case Folding* merupakan proses untuk mengubah kalimat menjadi huruf kecil. Pada tahap ini menggunakan bantuan library *Regular Expression* untuk menghilangkan komponen tertentu seperti URL, *username*, RT (Retweet), karakter HTML, dan hashtag.
 - b) *Tokenizing* merupakan proses untuk memisahkan kalimat menjadi kata per kata.
 - c) *Filtering* (Stopword Removal) merupakan proses menghapus kata-kata yang tidak penting dan tidak memiliki arti. Pada tahap ini menggunakan bantuan library NLTK untuk menghapus kata-kata yang terdapat dalam daftar *Stopword NLTK*.
 - d) *Stemming* merupakan proses untuk merubah kata menjadi kata dasar. Pada tahap ini menggunakan bantuan library Sastrawi.
4. Memberikan pembobotan pada setiap kata dari suatu dokumen dengan metode TF-IDF menggunakan bantuan library Scikit-Learn pada Python.
5. Proses Splitting Data atau membagi data *training* dan data *testing* menjadi 80% untuk data *training* dan 20% untuk data *testing*. Penentuan proporsi jumlah data *training* dan data *testing* tersebut berdasarkan penelitian sebelumnya dan perbandingan proporsi terbaik.
6. Membangun model *machine learning* dengan menggunakan metode *Naïve Bayes Classifier* dan *Support Vector Machine* sebagai metode klasifikasi.
7. Menggunakan metode *Confusion Matrix* sebagai metode evaluasi untuk mengetahui nilai *Accuracy*, *Precision*, *Recall*, dan *F1-score* dari masing-masing metode klasifikasi.
8. Interpretasi dan menarik kesimpulan..

3. HASIL DAN PEMBAHASAN**3.1 Pengambilan Data**

Proses pengambilan data dari media sosial Twitter dilakukan menggunakan teknik crawling dengan bantuan library *snsrape*, yaitu proses pengambilan data tweet dengan memanfaatkan sebuah library sehingga memudahkan untuk mengambil data tweet dengan jumlah yang besar dan rentang waktu kapanpun. *Snsrape* adalah sebuah library yang digunakan untuk mengambil data dari media sosial Twitter. Proses ini dibangun dengan bahasa pemrograman Python pada *notebook Google Colaboratory*. Langkah awal yang dilakukan adalah install dan import modul dari library *snsrape* yang berfungsi untuk mengambil data tweet dan library *pandas* untuk menampilkan serta menyimpan data dalam format *.csv*. Kemudian tentukan jumlah data tweet, query, dan rentang waktu yang diinginkan, Pada contoh ini mengambil data tweet dengan query Vaksin Covid-19 pada tanggal 1 Agustus 2021 sampai 31 Agustus 2021 yang berjumlah 500 tweet.

3.2 Text Preprocessing

Data dalam penelitian ini merupakan data tweet terkait Vaksin Covid-19 di Indonesia yang memiliki berbagai macam gaya penulisan sehingga data yang diperoleh merupakan data yang tidak terstruktur. Oleh karena itu, data tersebut perlu diubah menjadi data yang lebih terstruktur sebelum dilakukan klasifikasi. Tahapan untuk mengubah data yang tidak terstruktur menjadi data yang lebih terstruktur disebut tahap *text preprocessing*. Adapun langkah-langkah *text preprocessing* yang akan dilakukan adalah:

1) Case Folding

Proses ini dilakukan dengan mengubah huruf besar menjadi huruf kecil dengan tujuan agar huruf besar dan huruf kecil tidak terdeteksi memiliki arti yang berbeda. Perbedaan arti yang dimaksud adalah seperti kata “Vaksin” dan “vaksin”. Kedua kata tersebut memiliki arti yang sama, namun oleh sistem dideteksi sebagai kata yang berbeda karena memiliki susunan huruf kecil dan besar yang berbeda. Pada tahap ini menggunakan bantuan library *Regular Expression* untuk menghilangkan komponen tertentu seperti URL, *username*, RT (Retweet), karakter HTML, dan hashtag.

Tabel 2. Proses Case Folding

Sebelum <i>Case Folding</i>	Sesudah <i>Case Folding</i>
Pemerintah Gencar Lakukan Pemberian Vaksin untuk Atasi Pandemi Covid-19 https://t.co/DXerjS26FD https://t.co/LkiTMHG1zQ,1	pemerintah gencar lakukan pemberian vaksin untuk atasi pandemi covid
Terciptanya Vaksin Masih Belum Bisa Menghentikan Penyebaran Covid-19 https://t.co/RGBw8MGZ6f,0	terciptanya vaksin masih belum bisa menghentikan penyebaran covid
Beberapa Teman Tertular Covid-19 Setelah di Vaksin!!	beberapa teman tertular covid setelah di vaksin

2) *Tokenizing*

Merupakan proses pemisahan teks ulasan menjadi kata per kata yang tidak saling berpengaruh. Potongan kata tersebut dinamakan token, yaitu sebuah entitas yang mempunyai nilai dalam penyusunan matriks dokumen. Tokenizing berguna untuk mempermudah perhitungan frekuensi kemunculan kata dalam dokumen.

Tabel 3. Proses Tokenizing

Sebelum <i>Tokenizing</i>	Sesudah <i>Tokenizing</i>
pemerintah gencar lakukan pemberian vaksin untuk atasi pandemi covid	“pemerintah” “gencar” “lakukan” “pemberian” “vaksin” “untuk” “atasi” “pandemic” “covid”
terciptanya vaksin masih belum bisa menghentikan penyebaran covid	“terciptanya” “vaksin” “masih” “belum” “bisa” “menghentikan” “penyebaran” “covid”
beberapa teman tertular covid setelah di vaksin	“beberapa” “teman” “tertular” “covid” “setelah” “di” “vaksin”

3) *Filtering (Stopword Removal)*

Merupakan proses menghapus kata-kata yang tidak penting dan tidak memiliki arti. Pada tahap ini menggunakan bantuan library NLTK untuk menghapus kata-kata yang terdapat dalam daftar Stopword NLTK.

Tabel 4. Proses Filtering (Stopword Removal)

Sebelum <i>Filtering (Stopword Removal)</i>	Sesudah <i>Filtering (Stopword Removal)</i>
pemerintah gencar lakukan pemberian vaksin untuk atasi pandemi covid	pemerintah gencar lakukan pemberian vaksin atasi pandemi covid
terciptanya vaksin masih belum bisa menghentikan penyebaran covid	terciptanya vaksin menghentikan penyebaran covid
beberapa teman tertular covid setelah di vaksin	teman tertular covid vaksin

4) *Stemming*

Merupakan proses untuk merubah kata menjadi kata dasar. Pada tahap ini menggunakan bantuan library Sastrawi.

Tabel 5. Proses Stemming

Sebelum <i>Stemming</i>	Sesudah <i>Stemming</i>
-------------------------	-------------------------

pemerintah gencar lakukan pemberian vaksin atasi pandemi covid	perintah gencar laku beri vaksin atas pandemi covid
terciptanya vaksin menghentikan penyebaran covid	cipta vaksin henti sebar covid
teman tertular covid vaksin	teman tular covid vaksin

3.3 Ekstraksi Fitur

Setelah melalui tahap *text preprocessing*, data tweet selanjutnya melalui tahap ekstraksi fitur. Tahap ini berguna untuk mempersiapkan dataset, agar bisa diproses pada tahap klasifikasi menggunakan algoritma *Naïve Bayes Classifier* dan *Support Vector Machine*. Langkah pertama yang dilakukan pada ekstraksi fitur adalah merepresentasikan dataset ke dalam bentuk vektor dengan menggunakan library Python yang bernama *CountVectorizer*. Pada tahap pembuatan fitur dilakukan dua proses yaitu pembuatan word vector dimana dilakukan pengubahan fitur teks menjadi sebuah representasi vector dan pembobotan kata dengan *Tf-Idf*. Word vector adalah membuat kalimat yang sudah menjadi kumpulan array menjadi suatu matriks, yang mana setiap baris matriks tersebut mewakili baris dokumen, sedangkan kolom pada matriks mewakili seluruh kata yang ada di seluruh teks yang ada. Kemudian data yang sudah menjadi word vector kemudian dihitung menggunakan rumus *TF-IDF* sehingga menghasilkan word vector dengan nilai yang sudah terbobot. Adapun *TF* (*Term Frequency*) adalah frekuensi dari kemunculan sebuah term dalam dokumen yang bersangkutan, sedangkan *IDF* (*Inverse Document Frequency*) merupakan sebuah perhitungan dari bagaimana term didistribusikan secara luas pada koleksi dokumen yang bersangkutan. Setelah itu dataset siap diuji sebagai data training dan testing menggunakan metode *Naïve Bayes Classifier* dan metode *Support Vector Machine*.

3.4 Pelatihan dan Pengujian Data

Machine learning adalah sebuah mesin yang dirancang untuk belajar. Maka dalam model machine learning tersebut harus ada tujuan yang harus dicapai untuk melihat apakah performa yang diberikan sudah sesuai dengan tingkat akurasi yang diinginkan. Untuk mencapai tujuan tersebut, *machine learning* diberikan dataset untuk pelatihan (*data training*) dan pengujian (*data testing*). Semakin banyak dan variatif dataset yang diberikan maka model machine learning akan semakin baik. *Data training* akan digunakan untuk melatih model dan *data testing* akan digunakan untuk pengujian model. Pada penelitian ini akan menggunakan proporsi 80% untuk data latih dan 20% untuk data uji.

3.5 Klasifikasi *Naïve Bayes Classifier*

Data yang sudah melalui tahap *preprocessing* kemudian akan melalui tahap klasifikasi menggunakan *Naïve Bayes Classifier* untuk mengetahui probabilitas dari data tersebut apakah berkelas positif atau negatif. Metode ini menggunakan perhitungan probabilitas dan statistik dalam pengoperasiannya. Metode klasifikasi *Naïve Bayes Classifier* digunakan untuk mengambil keputusan dengan melakukan prediksi suatu kasus berdasarkan hasil dari klasifikasi yang telah diperoleh. Notasi klasifikasi *Naïve Bayes Classifier* ditampilkan pada Persamaan 1.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \quad (1)$$

Nilai prior adalah salah satu langkah mencari nilai probabilitas pada masing-masing kelas yang akan menghasilkan klasifikasi. Probabilitas prior kelas positif sebesar 0,71 yang artinya terdapat peluang kejadian sebesar 71%. Sedangkan untuk probabilitas prior kelas negatif sebesar 0,28 yang artinya terdapat peluang kejadian sebesar 28%. Dengan perhitungan sebagai berikut:

$$\text{Probabilitas Prior Positif} = \frac{\text{Jumlah Kelas Positif}}{\text{Jumlah Keseluruhan Data}}$$

$$= \frac{790}{1106} = 0,71$$

$$\text{Probabilitas Prior Negatif} = \frac{\text{Jumlah Kelas Negatif}}{\text{Jumlah Keseluruhan Data}}$$

$$= \frac{316}{1106} = 0,28$$

Proses ekstraksi fitur dan klasifikasi dengan metode *Naïve Bayes Classifier* menggunakan bantuan library Python yang bernama Scikit-Learn untuk proses klasifikasi. Algoritma Multinomial Naïve Bayes dipilih karena memiliki performa yang bagus dengan akurasi cukup tinggi dan waktu pemrosesan data yang cukup singkat.

3.6 Klasifikasi Support Vector Machine

Metode SVM bekerja dengan cara mencari hyperplane atau garis pemisah terbaik yang memiliki margin atau jarak antar kelas terbesar menggunakan beberapa kernel dalam klasifikasi SVM antara lain Linier, Radial Basic Function (RBF), dan Polynomial untuk memperoleh klasifikasi. Usaha untuk mencari lokasi hyperplane ini merupakan inti dari proses pembelajaran pada SVM. Pada penelitian ini akan difokuskan menggunakan fungsi Linier dan membagi menjadi 2 kelas yaitu positif dan negatif. Fungsi mencari hyperplane dapat didefinisikan pada persamaan 2 dan garis tepi pada hyperplane untuk membagi kelas positif dan negatif didefinisikan pada persamaan 3.

$$(w \cdot x) + b = 0 \quad (2)$$

$$(w \cdot x) + b = +1 \text{ (untuk kelas positif)} \quad (3)$$

$$(w \cdot x) + b = -1 \text{ (untuk kelas negatif)}$$

Sama seperti proses klasifikasi *Naïve Bayes Classifier*, Untuk proses klasifikasi dengan metode *Support Vector Machine* juga menggunakan bantuan library Scikit-Learn. Adapun library Scikit-Learn yang dipakai di antaranya adalah CountVectorizer, TfidfVectorizer, svm, accuracy_score, precision_score, recall_score, f1_score, classification_report, dan confusion_matrix. Langkah pertama untuk melakukan klasifikasi adalah menginstall library yang diperlukan. Selanjutnya setelah semua library yang diperlukan dipastikan sudah terinstal maka dilanjutkan dengan proses prediksi menggunakan data test untuk mendapatkan nilai accuracy, precision, recall, dan f1_score.

3.7 Evaluasi Model

Evaluasi model dilakukan setelah proses uji dan klasifikasi model selesai. Evaluasi model dilakukan untuk menghitung performa metode yang digunakan yaitu metode *Naïve Bayes Classifier* dan *Support Vector Machine*. Pada proses uji model yang dilakukan menghasilkan confusion matrix dengan ukuran 2x2.

Tabel 6. Hasil Confusion Matrix SVM

SVM		Predict Class	
		Positive	Negative
Actual Class	Positive	32	22
	Negative	8	160

Tabel 7. Hasil Confusion Matrix NBC

NBC		Predict Class	
		Positive	Negative

<i>Actual Class</i>	<i>Positive</i>	13	41
	<i>Negative</i>	1	167

Hasil confusion matrix pada metode SVM yaitu True Positive (TP) 32 data, False Positive (FP) 22 data, True Negative (TN) 160 data, dan False Negative (FN) 8 data. Kemudian untuk Hasil confusion matrix pada metode NBC yaitu True Positive (TP) 13 data, False Positive (FP) 41 data, True Negative (TN) 167 data, dan False Negative (FN) 1 data.

Dimana:

TP = Data positif yang terdeteksi atau terklasifikasi sebagai data positif.

FP = Data negatif namun terdeteksi atau terklasifikasi sebagai data positif.

TN = Data negatif yang terdeteksi atau terklasifikasi sebagai data negatif.

FN = Data positif namun terdeteksi atau terklasifikasi sebagai data negatif.

Berdasarkan hasil confusion matrix maka untuk metode SVM pada kelas positif mendapatkan nilai *accuracy* sebesar 0.87, nilai *precision* sebesar 0.88, nilai *recall* sebesar 0.96, dan nilai f-1 score sebesar 0.92. Sedangkan untuk kelas negatif mendapatkan nilai *accuracy* sebesar 0.81, nilai *precision* sebesar 0.84, nilai *recall* sebesar 0.57, dan nilai f-1 score sebesar 0.68.

Berikut tabel hasil *accuracy*, *precision*, *recall*, dan f-1 score di setiap kelasnya pada metode SVM.

Tabel 8. Nilai Accuracy, Precision, Recall, dan f1-score Metode SVM

Jenis Klasifikasi	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
Positif	0.87	0.88	0.96	0.92
Negatif	0.87	0.84	0.57	0.68

Selanjutnya berdasarkan hasil confusion matrix dan perhitungan menggunakan persamaan (7), (8), (9), dan (10) untuk metode NBC pada kelas positif mendapatkan nilai *accuracy* sebesar 0.81, nilai *precision* sebesar 0.80, nilai *recall* sebesar 0.99, dan nilai f-1 score sebesar 0.89. Sedangkan untuk kelas negatif mendapatkan nilai *accuracy* sebesar 0.81, nilai *precision* sebesar 0.93, nilai *recall* sebesar 0.24, dan nilai f-1 score sebesar 0.38.

Berikut tabel hasil *accuracy*, *precision*, *recall*, dan f-1 score di setiap kelasnya pada metode NBC.

Tabel 9. Nilai Accuracy, Precision, Recall, dan f1-score Metode NBC

Jenis Klasifikasi	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
Positif	0.81	0.80	0.99	0.89
Negatif	0.81	0.93	0.24	0.38

Sehingga didapatkan hasil perbandingan performansi dari keseluruhan proses evaluasi model SVM dan NBC, bahwa SVM cenderung memiliki performansi yang lebih baik dibandingkan model NBC.

Tabel 10. Hasil Perbandingan Performansi NBC dan SVM

Model	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>
NBC	0.81	0.80	0.99	0.89
SVM	0.87	0.88	0.96	0.92

4. KESIMPULAN DAN SARAN

Berdasarkan hasil pengujian menggunakan *Naïve Bayes Classifier* dan *Support Vector Machine* maka ada beberapa hal yang dihasilkan, antara lain:

- 1) Hasil ketepatan klasifikasi menggunakan metode *Naïve Bayes Classifier* diperoleh *accuracy* sebesar 81%, *precision* sebesar 80%, *recall* sebesar 99%, dan *f1_score* sebesar 89%.
- 2) Hasil ketepatan klasifikasi menggunakan metode *Support Vector Machine* diperoleh *accuracy* sebesar 87%, *precision* sebesar 88%, *recall* sebesar 96%, dan *f1_score* sebesar 92%.
- 3) Secara keseluruhan perbandingan performansi metode *Naïve Bayes Classifier* dan *Support Vector Machine* menunjukkan hasil bahwa *Support Vector Machine* lebih baik dalam mengklasifikasi data..

Berdasarkan hasil analisis dan kesimpulan, peneliti dapat memberi saran yaitu untuk penelitian selanjutnya, dalam memperoleh data dapat mengambil data pada media sosial populer lainnya seperti Facebook dan Instagram agar data yang didapat lebih bervariasi dan dapat menggunakan metode klasifikasi yang lain sehingga diperoleh hasil klasifikasi yang lebih spesifik dan baik.

5. DAFTAR PUSTAKA

- Alvina Felicia Watratan, Arwini Puspita, B., & Dikwan Moeis. (2020). Implementasi Algoritma *Naive Bayes* Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia. *Journal of Applied Computer Science and Technology*, 1(1), 7–14. <https://doi.org/10.52158/jacost.v1i1.9>
- Anees, A. F., Shaikh, A., Shaikh, A., & Shaikh, S. (2020). Survey Paper on Sentiment Analysis : Techniques and Challenges. *EasyChair*, 2516–2314.
- Astari, N. M. A. J., Dewa Gede Hendra Divayana, & Gede Indrawan. (2020). Analisis Sentimen Dokumen Twitter Mengenai Dampak Virus Corona Menggunakan Metode *Naive Bayes Classifier*. *Jurnal Sistem Dan Informatika (JSI)*, 15(1), 27–29. <https://doi.org/10.30864/jsi.v15i1.332>
- D'Andrea, A., Ferri, F., Grifoni, P., & Guzzo, T. (2015). Approaches, Tools and Applications for Sentiment Analysis Implementation. *International Journal of Computer Applications*, 125(3), 26–33. <https://doi.org/10.5120/ijca2015905866>
- Dey, S., Wasif, S., Tonmoy, D. S., Sultana, S., Sarkar, J., & Dey, M. (2020). A Comparative Study of *Support Vector Machine* and *Naive Bayes Classifier* for Sentiment Analysis on Amazon Product Reviews. *2020 International Conference on Contemporary Computing and Applications, IC3A 2020, February*, 217–220. <https://doi.org/10.1109/IC3A48958.2020.233300>
- Fridom Mailo, F., Lazuardi, L. (2019). Analisis Sentimen Data Twitter Menggunakan Metode Text Mining Tentang Masalah Obesitas di Indonesia. *Jurnal Sistem Informasi Kesehatan Masyarakat Journal of Information Systems for Public Health*, 4(1), 28–36. <https://jurnal.ugm.ac.id/jisph/article/view/44455>
- Giovani, A. P., Ardiansyah, A., Haryanti, T., Kurniawati, L., & Gata, W. (2020). Analisis Sentimen Aplikasi Ruang Guru Di Twitter Menggunakan Algoritma Klasifikasi. *Jurnal Teknoinfo*, 14(2), 115. <https://doi.org/10.33365/jti.v14i2.679>
- Gokgoz, E., & Subasi, A. (2015). Biomedical Signal Processing and Control Comparison of decision tree algorithms for EMG signal classification using DWT. *Biomedical Signal Processing and Control*, 18, 138–144.
- Ilmiah, J. P. (2017). Text Mining Dan Sentimen Analisis Twitter Pada Gerakan Lgbt. *Intuisi : Jurnal Psikologi Ilmiah*, 9(1), 18–25. <https://doi.org/10.15294/intuisi.v9i1.9561>
- Irsyad, H., Farisi, A., & Pribadi, M. R. (2019). Klasifikasi Opini Masyarakat Terhadap Jasa ISP MyRepublic dengan *Naive Bayes*. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi (JNTETI)*, 8(1), 30. <https://doi.org/10.22146/jnteti.v8i1.487>
- KOMINFO, KEMENKES, K. P. (2021). *Data Vaksinasi COVID-19 (Update per 9 Oktober 2021)*. <https://covid19.go.id/berita/data-vaksinasi-covid-19-update-9-oktober-2021>
- Lestandy, M., Abdurrahim, A., & Syafa'ah, L. (2021). Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(10), 802–808.
- Lin, Y., Wang, X., & Zhou, A. (2016). Opinion spam detection. *Opinion Analysis for Online Reviews*, May, 79–94. https://doi.org/10.1142/9789813100459_0007
- Liu, C., Zhou, Q., Li, Y., Garner, L. V., Watkins, S. P., Carter, L. J., Smoot, J., Gregg, A. C., Daniels, A. D., Jervey, S., & Albaiu, D. (2020). Research and Development on Therapeutic Agents and Vaccines for COVID-19 and Related Human Coronavirus Diseases. *ACS Central Science*, 6(3), 315–331. <https://doi.org/10.1021/acscentsci.0c00272>
- Mean, R., Error, S., Mean, R., Error, S., Matrix, C., Curve, R. O. C., Curve, A. U., Evaluation, I., & Evaluation, E. (n.d.). *Evaluasi dan Validasi Evaluasi*.

- Mika Parwita, I. M., & Siahaan, D. (2019). Classification of Mobile Application Reviews using Word Embedding and Convolutional Neural Network. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, 10(1), 1. <https://doi.org/10.24843/lkjiti.2019.v10.i01.p01>
- Nakov, P., Ritter, A., Rosenthal, S., Sebastiani, F., & Stoyanov, V. (2016). SemEval-2016 task 4: Sentiment analysis in twitter. *SemEval 2016 - 10th International Workshop on Semantic Evaluation, Proceedings*, 1–18. <https://doi.org/10.18653/v1/s16-1001>
- Prastyo, P. H., Sumi, A. S., Dian, A. W., & Permanasari, A. E. (2020). Tweets Responding to the Indonesian Government's Handling of COVID-19: Sentiment Analysis Using SVM with Normalized Poly Kernel. *Journal of Information Systems Engineering and Business Intelligence*, 6(2), 112. <https://doi.org/10.20473/jisebi.6.2.112-122>
- President of the Republic of Indonesia. (2020). Presidential Regulation 99/2020 on COVID-19 Vaccine Procurement [Peraturan Presiden No. 99 Tahun 2020 tentang Pengadaan Vaksin dan Pelaksanaan Vaksinasi dalam Rangka Penanggulangan Pandemi Corona Virus Disease 2019 (Covid-19)]. *Presidential Regulation*, 2019(039471), 1–13. <https://peraturan.bpk.go.id/Home/Details/147944/perpres-no-99-tahun-2020>
- Rachman, F. F., & Pramana, S. (2020). Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media Sosial Twitter. *Health Information Management Journal*, 8(2), 100–109. <https://inohim.esaunggul.ac.id/index.php/INO/article/view/223/175>
- Rachman, F., Health, S. P.-I. of, & 2020, undefined. (2020). Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media Sosial Twitter. *Inohim.Esaunggul.Ac.Id*, 8(2), 2655–9129. <https://inohim.esaunggul.ac.id/index.php/INO/article/download/223/175>
- Rahmadya Trias Handayanto, H. (2020). *Data Mining dan Machine Learning Menggunakan Matlab dan Python* (Oktober 20). Informatika Bandung. <https://openlibrary.telkomuniversity.ac.id/pustaka/163039/data-mining-dan-machine-learning-menggunakan-matlab-dan-python.html>
- Rasool, A., Tao, R., Marjan, K., & Naveed, T. (2019). Twitter Sentiment Analysis: A Case Study for Apparel Brands. *Journal of Physics: Conference Series*, 1176(2). <https://doi.org/10.1088/1742-6596/1176/2/022015>
- Rochmawati, N., & Wibawa, S. C. (2018). Opinion Analysis on Rohingya using Twitter Data. *IOP Conference Series: Materials Science and Engineering*, 336(1). <https://doi.org/10.1088/1757-899X/336/1/012013>
- Sistem, R., Lestandy, M., Abdurrahim, A., & Syafa, L. (2021). Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent. 5(10), 802–808.
- Song, J., Kim, K. T., Lee, B., Kim, S., & Youn, H. Y. (2017). A novel classification approach based on Naïve Bayes for Twitter sentiment analysis. *KSII Transactions on Internet and Information Systems*, 11(6), 2996–3011. <https://doi.org/10.3837/tiis.2017.06.011>
- Susilo, A., Rumende, C. M., Pitoyo, C. W., Santoso, W. D., Yulianti, M., Herikurniawan, H., Sinto, R., Singh, G., Nainggolan, L., Nelwan, E. J., Chen, L. K., Widhani, A., Wijaya, E., Wicaksana, B., Maksum, M., Annisa, F., Jasirwan, C. O. M., & Yuniastuti, E. (2020). Coronavirus Disease 2019: Tinjauan Literatur Terkini. *Jurnal Penyakit Dalam Indonesia*, 7(1), 45. <https://doi.org/10.7454/jpdi.v7i1.415>
- Syarifuddin, M. (2020). Analisis Sentimen Opini Publik Mengenai Covid-19 Pada Twitter Menggunakan Metode Naïve Bayes Dan Knn. *Inti Nusa Mandiri*, 15(1), 23–28.
- Tri, F., & Saputra, G. (2020). *Analisis Sentimen Bahasa Indonesia Berbasis Leksikon pada Twitter Pembahasan Pustaka*.
- Watrianthos, R., Suryadi, S., Irmayani, D., Nasution, M., & Simanjorang, E. F. S. (2019). Sentiment analysis of traveloka app using Naïve Bayes Classifier method. *International Journal of Scientific and Technology Research*, 8(7), 786–788. <https://doi.org/10.31227/osf.io/2dbe4>