

Komparasi Algoritma Support Vector Machine Dengan Naive Bayes Untuk Analisis Sentimen Pada Aplikasi BRImo

Anggi Puji Astuti¹, Syariful Alam², Irsan Jaelani³

Teknik Informatika, Sekolah Tinggi Teknologi Wastukencana

Jl. Cikopak No. 53 Mulyamekar, Kec. Babakancikao, Sadang, Kabupaten Purwakarta, Jawa Barat

¹anggi puji20@wastukencana.ac.id, ²syarifulalam@wastukencana.ac.id, ³irsan@wastukencana.ac.id

Intisari — Perbankan merupakan industri yang saat ini sudah berkembang dalam pemanfaatan teknologi informasi dengan meningkatkan standar kualitas layanan agar dapat bersaing dipasar pada era digital yang semakin ketat. Pada saat ini bank BRI sedang menarik perhatian masyarakat akan kualitas pembaharuan dengan meluncurkan aplikasi *mobile banking*, maka dari itu dilakukanlah analisis mengenai ulasan pengguna *mobile banking* BRImo untuk dijadikan sebagai objek penelitian dengan melakukan komparasi metode klasifikasi text mining. Penelitian ini bertujuan untuk mengetahui hasil komparasi algoritma Support vector machine dan Naive bayes. Algoritma Support vector machine dan Naive bayes adalah suatu metode klasifikasi untuk mengolah data berupa teks dengan tingkat akurasi yang baik. Algoritma ini biasanya digunakan untuk analisis text mining dengan 4 tahapan yaitu Scrapping data, Preprocessing, Klasifikasi dan evaluasi. Pada tahap preprocessing memiliki beberapa proses diantaranya filtering, labeling, case folding, tokenization, stopword removal & Stemming dan normalisasi agar mendapatkan suatu kata yang bisa diklasifikasikan. Hasil dari penelitian ini yaitu algoritma Support vector machine merupakan algoritma yang lebih baik dalam klasifikasi data ulasan aplikasi mobile banking BRImo dengan nilai akurasi sebesar 97,69% dibandingkan algoritma Naive Bayes dengan nilai akurasi sebesar 96,53%.

Kata kunci — Support vector machine, Naive bayes, perbankan, Aplikasi dan Mobile banking BRImo

Abstract — Banking is an industry that is currently developing in the use of information technology by increasing service quality standards in order to compete in the market in an increasingly tight digital era. At this time, BRI bank is attracting public attention to the quality of renewal by launching a mobile banking application, therefore an analysis of reviews of BRImo mobile banking users is carried out to serve as an object of research by comparing text mining classification methods. This study aims to determine the results of the comparison of the Support vector machine and Naive Bayes algorithms. Support vector machine and Naive Bayes algorithm are classification methods used to process data in the form of text with a good level of accuracy. This algorithm is usually used for text mining analysis with 4 stages, namely data scrapping, preprocessing, classification and evaluation. At the preprocessing stage, there are several processes including filtering, labeling, case folding, tokenization, stopword removal & stemming and normalization to get relevant words to be classified. The results of this study are the Support vector machine algorithm has a better performance in classifying the BRImo mobile banking application review data with an accuracy value of 97,69% compared to the Naive Baye algorithm with an accuracy value of 96,53%.

Keywords — Support vector machine, Naive bayes, Bank, Application and Mobile banking BRImo

I. PENDAHULUAN

Perbankan merupakan industri yang saat ini sudah berkembang dalam pemanfaatan teknologi informasi dengan meningkatkan standar kualitas layanan agar dapat bersaing dipasar pada era digital yang semakin ketat. Pada saat ini perusahaan bank memiliki aplikasi *mobile banking* (m-banking) yang digunakan oleh nasabahnya. Akan tetapi perusahaan bank harus mengumpulkan informasi terkait pelayanan yang dibutuhkan oleh nasabah. [1]. Penelitian ini dilakukan untuk mengetahui ulasan Aplikasi BRImo pada *Google Play Store* yang dijadikan sebagai objek penelitian dikarenakan Bank BRI adalah bank tertua di Indonesia saat ini sedang menarik perhatian masyarakat akan kualitas

pembaharuan dengan meluncurkan aplikasi *mobile banking*. Selain itu analisis sentimen juga dilakukan dengan menggunakan algoritma data mining untuk melakukan proses, analisa dan mengubah data yang berupa teks kedalam suatu objek atau entitas.

Berdasarkan penelitian serupa yang telah dilakukan memiliki perbedaan mengenai metode atau algoritma serta penambahan fitur pembobotan kata yang digunakan. Pada penelitian [2] algoritma *Support Vector Machine* dan *Naive Bayes* kemudian ditambah fitur *Genetic Algorithm* agar nilai akurasi pengklasifikasian tersebut menentukan pola positif atau negatif mengenai Calon Gubernur Jabar 2018-2023 berdasarkan data pengguna twitter dengan hasil yang diperoleh membuktikan bahwa model algoritma SVM berbasis GA adalah model algoritma terbaik dalam pengujian

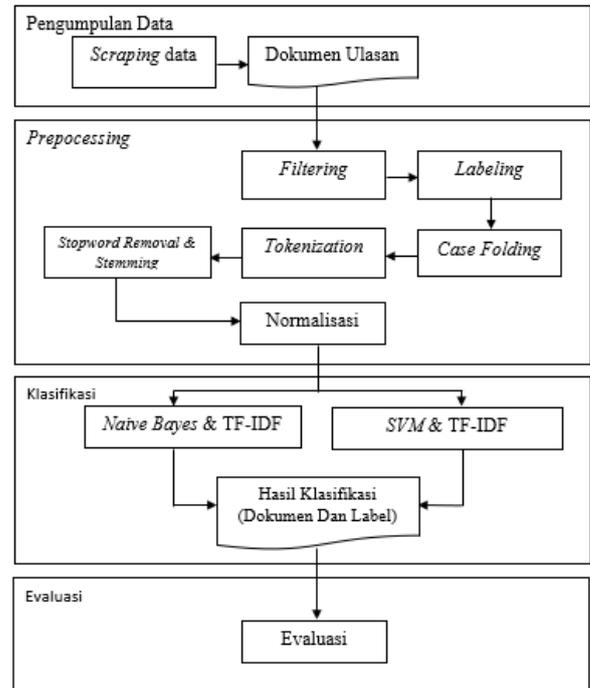
dan pengklasifikasi dibandingkan hasil yang diperoleh oleh algoritma NB berbasis GA. Penelitian selanjutnya [3] menggunakan algoritma *Neural Network*, *K-Neighbor*, *Naive Bayes*, *support Vector Machine* dan *Decision Tree* untuk melakukan ekstraksi atribut dalam ulasan agar diklasifikasikan menjadi ulasan positif dan negatif mengenai usaha waralaba berdasarkan data twitter dengan hasil algoritma *Neural Network*, *Support Vector Machine* dan *Naive Bayes* sebagai klasifikasi algoritma terbaik. Penelitian [4] algoritma *K-Nearest Neighbor* dan *Support Vector Machine* dengan *K-Fold Cross Validation* atau tanpa *K-Fold Cross Validation* dengan objek teks sentimen bertujuan untuk melakukan perbandingan agar diketahui algoritma yang lebih baik dari segi akurasi dan waktu yang menghasilkan bahwa metode *K-Nearest Neighbor* tanpa *K-Fold Cross Validation* memiliki performa yang lebih baik. Penelitian [5] algoritma *Naive Bayes*, *Random Forest* dan *Support Vector Machine* dengan tujuan untuk mengetahui eksperimen mengenai ulasan yang ada pada aplikasi belajar RuangGuru dengan melakukan pengujian menggunakan *Cross Validation* serta *K-Fold* sebanyak K 10 dengan hasil akhir analisa dari beberapa pengujian model klasifikasi tersebut, didapatkan jumlah komentar positif lebih banyak dari komentar negatif dengan performa menunjukkan algoritma *Random Forest* memperoleh akurasi yang tinggi. Penelitian [6] melakukan klasifikasi terhadap aplikasi jenius yang memiliki komentar di *Google Play Store* dengan algoritma *Naive Bayes* dengan hasil performa yang didapat tersebut dapat disimpulkan bahwa ulasan pada aplikasi jenius memiliki nilai yang dihasilkan oleh akurasi relatif kecil yaitu 0,57. Selanjutnya penelitian[7] menggunakan algoritma *Support Vector Machine* dan menambahkan fitur *Word N-Gram* untuk melakukan identifikasi kata melalui pendekatan dengan tujuan penelitian agar mengetahui pengaruh penerapan *Word N-Gram* terhadap nilai akurasi yang diperoleh dengan hasil kenaikan tertinggi yaitu pada penerapan tri-gram sebesar 0,92% sehingga penerapan *Word N-Gram* dapat meningkatkan suatu akurasi berjumlah banyak dengan data yang digunakan.

Berdasarkan penelitian sebelumnya, penelitian menggunakan algoritma *Support Vector Machine* dan *Naive Bayes* dengan tujuan untuk melakukan komparasi agar mengetahui nilai akurasi dengan dikombinasikan menggunakan pembobotan kata TF-IDF.

II. METODOLOGI PENELITIAN

A. Diagram Alur Penelitian

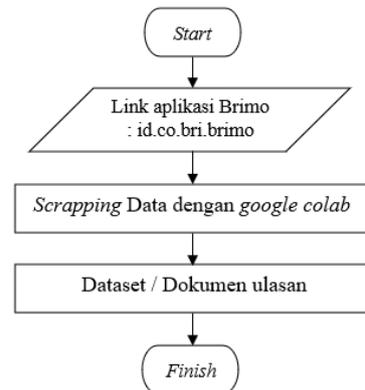
Pada Diagram Alur memiliki tahapan untuk proses penelitian. Gambar 1 berikut menjelaskan mengenai gambaran umum alur penelitian.



Gambar 1. Alur Penelitian

B. Pengumpulan Data

Pada tahapan ini, peneliti mengumpulkan data menggunakan teknik *scrapping* dengan bantuan *Google Colab*. *Scrapping* data digunakan untuk mengambil data komentar atau ulasan dari sebuah halaman *website* atau aplikasi android dari *Google Play Store* yang kemudian data tersebut diolah kedalam suatu format yang dapat disimpan menggunakan *python* [8] di *google colab* dengan link tujuan aplikasi BRImo yaitu "*id.co.bri.brimo*" menggunakan *library* *pandas* dan *numpy*. Hasil *scrapping* sebanyak 5000 data ulasan pengguna. Berikut adalah alur dalam melakukan *scrapping* data pada gambar 2. :



Gambar 2. Pengumpulan Data

C. Preprocessing

Setelah didapatkan data hasil *scrapping* maka selanjutnya dokumen ulasan pengguna ini akan dilakukan

tahapan *preprocessing*. *Preprocessing* digunakan dalam dokumen mentah hingga siap untuk dilakukan analisis teks yang telah disiapkan [9]. Tahapan *preprocessing* penelitian ini yaitu *filtering*, *labeling*, *case folding*, *tokenization*, *stopword removal* & *stemming* dan normalisasi. Berikut adalah tabel penjelasan mengenai tahapan *Preprocessing* pada tabel 1. :

TABEL I
PENJELASAN TAHAPAN *PREPROCESSING*

No	Tahapan	Penjelasan
1	<i>Filtering</i>	Digunakan untuk menghilangkan atau menghapus data duplikat dan memilih atribut apa saja yang akan digunakan.
2	<i>Labeling</i>	Untuk menentukan kelas daripada suatu ulasan dalam dokumen untuk menentukan label pada setiap ulasan antara positif atau negatif
3	<i>Case folding</i>	Untuk mengubah semua tulisan menjadi huruf kecil
4	<i>Tokenization</i>	Untuk pemecahan kalimat dan menghilangkan tanda baca
5	<i>Stopword removal & stemming</i>	Teknik untuk menghapus kata berimbuhan
6	Normalisasi	Proses untuk mengubah bahasa baku

D. Klasifikasi

Tahapan klasifikasi merupakan proses perhitungan yang menerapkan algoritma data mining untuk mengolah dataset ulasan dari aplikasi BRIMO pada *Google Play Store* yang sudah melewati tahap *preprocessing* terlebih dahulu. Algoritma yang digunakan yaitu *Naive Bayes* dan *Support Vector Machine* dibantu dengan pembobotan TF-IDF. Pada tahap algoritma *Naive Bayes* dilakukan perhitungan kumpulan dari probabilitas dengan cara menjumlahkan frekuensi dan kombinasi nilai. Sedangkan tahap yang dilakukan algoritma *Support Vector Machine* yaitu mengklasifikasi proses data untuk mengetahui *hyperplanes*, dengan tepi terbesar dari *hyperplanes* menjadi pembagi data antar kelas.

E. Evaluasi

Tahapan evaluasi yang digunakan yaitu metode *Confusion Matrix* yang biasa digunakan untuk melakukan perhitungan akurasi dan evaluasi model klasifikasi. Setelah metode *Confusion Matrix* ini selesai maka proses terakhir yang dilakukan pada evaluasi yaitu menampilkan *wordcloud* pada masing-masing label positif atau negatif. Berikut rumus *Confusion Matrix* bisa dilihat di tabel 2. Dibawah ini :

TABEL II
RUMUS *CONFUSION MATRIX*

Data	Kelas Prediksi	
	1	0
Kelas	1	TN
Sebenarnya	0	FP

Keterangan :

TP (*True Positive*) = jumlah dokumen dari kelas 1 yang benar di klasifikasikan sebagai kelas 1

TN (*True Negative*) = jumlah dokumen dari kelas 0 yang benar diklasifikasikan sebagai kelas 0

FP (*False Positive*) = jumlah dokumen dari kelas 0 yang salah diklasifikasikan sebagai kelas 1

FN (*False Negative*) = jumlah dokumen dari kelas 1 yang salah diklasifikasikan sebagai kelas 0

III. HASIL DAN PEMBAHASAN

A. Data

Data yang diambil pada situs *Google Play Store* merupakan ulasan pengguna aplikasi BRIMO. Pengambilan data menggunakan teknik *scrapping* dengan memanfaatkan *Google Colab*. Pada penelitian ini *Google Colab* digunakan untuk mengambil data ulasan dan mengolah data menggunakan bahasa pemrograman *python* dengan jumlah data yang didapatkan sebanyak 5000 data ulasan terbaru yang didapatkan. Berikut hasil *Scrapping* Data pada gambar 3. :

A	B	C	D	E	F	G	H	I	J	K
userName	score	at	content							
tresno listya	5	29/03/2022 03.14	Terima kasih aplikasi ini sangat bermanfaat...							
zacky zidan	5	29/03/2022 03.14	mantap BR!							
Helmi Pim	5	29/03/2022 03.14	BRIMO SANGAT BERARTI BAGIKU, DISAAT KEMALAMAN, SAYA BISA TRANSPER SEGAI							
Syahrul Lujap	5	29/03/2022 03.14	Lebih aman mudah sukses bri memberdayakan nasabah							
Refayan Carolis	5	29/03/2022 03.14	Sangat bermanfaat terimakasih ...							
AGUS RIYANTO	5	29/03/2022 03.14	ðŸ™‚ðŸ™‚ðŸ™‚ðŸ™‚ðŸ™‚ðŸ™‚							
Sondang Borujait	5	29/03/2022 03.14	Sangat mudah dn rmbantu.							
Abdul Rohman	5	29/03/2022 03.13	Amat sangat membantu makasih BRIMO							
Rubani	5	29/03/2022 03.13	Terima kasih BRIMO.							
Asad Mustyafa	5	29/03/2022 03.13	Sangat membantu							
Hironimus Telaumbanua	5	29/03/2022 03.13	Sangat membantu							
Yusri Mubarak	5	29/03/2022 03.13	Sejauh ini lancarr...							
game bryand	5	29/03/2022 03.13	Mantab dan sangat membantu							
Abd Hakim	5	29/03/2022 03.13	Memudahkan semua urusan dengan BRIMO cukup dirumah aja semua serba cepat..							
Akramuiddin Hambali	5	29/03/2022 03.12	Bagus							
Leksi FamFair	5	29/03/2022 03.11	Allhamdulillah sangat membantu ðŸ™‚							
Fabry Sulistiana	5	29/03/2022 03.11	Mantap banget.. Komplit							
tommy nuany	5	29/03/2022 03.11	Jangan ada potongan saldo dong							
Orsia Seja	5	29/03/2022 03.10	Transaksi Lancar							
Pidin Pidin	5	29/03/2022 03.10	Apk yang sangat bagus							

Gambar 3. Dataset Hasil *Scrapping*

B. *Preprocessing*

Data yang sudah didapatkan pada tahap *Scrapping* akan diolah menjadi data terstruktur atau biasa disebut *Preprocessing*. Berikut merupakan tahapan *Preprocessing* yang dilakukan:

1. *Filtering*

Dari hasil *Scrapping* atribut yang diperoleh yaitu *UserName*, *Score*, *at dan content*. Pada tahap *filtering*, atribut yang digunakan hanya *score* dan *content* yang nantinya akan di analisis. Berikut adalah data yang sudah dilakukan *Filtering* pada tabel 3. :

TABEL III
HASIL *FILTERING*

<i>Score</i>	<i>Content</i>
5	Terima kasih aplikasi ini sangat bermanfaat...
5	mantap BRI

2. *Labeling*

Proses yang dilakukan yaitu menentukan kelas dari setiap ulasan yang di dapat apakah masuk kedalam kelas positif atau kelas negatif. Proses pelabelan ini menggunakan bahasa pemrograman *python* pada *Google Colab*. Pada tabel 4. merupakan proses *Labeling* :

TABEL IV
PROSES *LABELING*

<i>Score</i>	<i>Content</i>	<i>Sentiment</i>
5	Terima kasih aplikasi ini sangat bermanfaat...	positif
1	Gak bisa daftar brimo padahal foto ktpnya sudah pas dan gak blurr anehh buang" data aja download	negatif

3. *Case Folding*

Proses konversi seluruh ulasan untuk mengubah tulisan dari huruf kapital menjadi *lowercase*. Proses *case folding* pada tabel 5 dibawah ini :

TABEL V
PROSES *CASE FOLDING*

<i>Input</i>	Gak bisa daftar brimo padahal foto ktpnya sudah pas dan gak blurr anehh buang" data aja download
<i>Output</i>	gak bisa daftar brimo padahal foto ktpnya sudah pas dan gak blurr anehh buang" data aja download

4. *Tokenization*

Merupakan proses pemecahan setiap kata atau teks ulasan sekaligus proses untuk menghilangkan tanda baca. Pada tabel 6. merupakan proses dari *tokenization*.

TABEL VI
PROSES *TOKENIZATION*

<i>Input</i>	terbaik!!! tingkatkan terus pelayanannya, segala urusan menjadi mudah dengan apk brimo ðŸ•• ðŸ••
<i>Output</i>	[terbaik, tingkatkan, terus, pelayanannya, segala, urusan, menjadi, mudah, dengan, apk, brimo]

5. *Stopword removal & stemming*

Merupakan proses untuk menghapus konjungsi atau imbuhan seperti "ke", "dan" atau "yang" sehingga hanya kata yang memiliki makna penting yang digunakan. Pada tabel 7. merupakan proses dari *stopword removal* dan *stemming*.

TABEL VII
PROSES *STOPWORD REMOVAL & STEMMING*

<i>Input</i>	transaksi lebih mudah tidak perlu ke atm
<i>Output</i>	[transaksi, lebih, mudah, tidak, perlu, atm]

6. *Normalisasi*

Merupakan proses untuk mengubah kata yang tidak sesuai dengan ejaan (tidak baku) berdasarkan database kamus kata yang dibuat sendiri dari data yang digunakan. Pada table 8. berikut merupakan proses dari normalisasi.

TABEL VIII
PROSES *NORMALISASI*

<i>Input</i>	[bgus, sekl, mudah, transfer, atm]
<i>Output</i>	[bagus, sekali, mudah, transfer, atm]

C. *Klasifikasi*

Pada tahapan klasifikasi diawali menggunakan pembobotan TF-IDF dengan pengelompokan label. Setelah proses pembobotan TF-IDF selesai maka data dibagi dua yaitu data *training* dan data *testing*.

1. *Pembobotan TF-IDF*

Setelah tahap *Preprocessing* selesai, maka dilakukan pemberian bobot pada setiap kata. Nilai bobot yang diberikan akan memberi pengaruh kemiripan saat mengukur kata dari suatu dokumen. Pembobotan TF-IDF ini melihat dari jumlah label positif dan negatif yang sudah ditentukan, label positif memiliki jumlah 2140 sedangkan label negatif berjumlah 194.

2. *Data Training dan Data Testing*

Proses *Data Training* dan *Data Testing* dibagi menjadi 3 (tiga) nilai perbandingan untuk memperkuat nilai akurasi yang diperoleh. Nilai perbandingan pertama dengan nilai *Data Training* 70% dan data *testing* 30% pada tabel 9 dibawah ini:

TABEL IX
PERBANIDINGAN 70 : 30

<i>Jenis Data</i>	<i>Presentase</i>	<i>Jumlah</i>
-------------------	-------------------	---------------

Data Training	70%	1.633,4 = 1.633
Data Testing	30%	700,6 = 701
Jumlah	100%	2.334

Kemudian nilai perbandingan kedua dengan nilai data *training* 80% dan data *testing* 20% pada tabel 10 dibawah ini :

TABEL X
PERBANDINGAN 80:20

Jenis Data	Presentase	Jumlah
Data Training	80%	1.867,2 = 1867
Data Testing	20%	466,8 = 467
Jumlah	100%	2.334

selanjutnya nilai perbandingan ketiga atau terkahir dengan nilai data *training* 90% dan data *testing* 10% pada tabel 11 dibawah ini :

TABEL XI
PERBANDINGAN 90:10

Jenis Data	Presentase	Jumlah
Data Training	90%	2.100,4 = 2.100
Data Testing	10%	233,6 = 234
Jumlah	100%	2.334

3. Klasifikasi *Support Vector Machine* dan *Naive bayes*.
Proses pengujian dilakukan dengan bahasa pemrograman *python* pada *google collab* dengan dikombinasikan menggunakan pembobotan kata TF-IDF. Pada tahap klasifikasi ini di uji 5 kali percobaan dengan *randomstate* yang berbeda-beda. Hasil dari klasifikasi antara algoritma *Support Vector Machine* dan *Naive bayes* dapat dilihat pada tabel 12. dibawah ini :

TABEL XII
HASIL PERBANDINGAN AKURASI

Percobaan	Akurasi Model	
	SVM	NB
Perbandingan 70:30		
Percobaan 1	97,28%	96,43%
Percobaan 2	97,00%	95,72%
Percobaan 3	97,86%	96,14%
Percobaan 4	96,86%	96,57%
Percobaan 5	97,57%	95,72%
Rata-Rata	97,31%	96,12%

Percobaan	Akurasi Model	
	SVM	NB
Perbandingan 80:20		
Percobaan 1	97,85%	97,00%
Percobaan 2	97,21%	96,35%
Percobaan 3	97,64%	97,21%
Percobaan 4	97,43%	96,14%
Percobaan 5	97,43%	97,64%
Rata-Rata	97,51%	96,87%
Perbandingan 90:10		
Percobaan 1	97,86%	95,29%
Percobaan 2	96,58%	95,29%
Percobaan 3	97,43%	97,43%
Percobaan 4	99,14%	97,00%
Percobaan 5	98,29%	97,86%
Rata-Rata	97,86%	96,57%
Rata-Rata Total	97,56%	96,52%

Berdasarkan hasil perbandingan akurasi algoritma *Support Vector Machine* memiliki nilai lebih besar untuk semua percobaan kecuali pada percobaan kelima dengan perbandingan 80:20 nilai akurasi *Naive bayes* lebih besar . Sehingga dapat disimpulkan berdasarkan hasil rata-rata total akurasi yang diperoleh metode *Support vector machine* memperoleh akurasi yang besar yaitu 97,56% dibandingkan dengan metode *Naive bayes* memiliki rata-rata total akurasi sebesar 96,52%.

D. Evaluasi

Pada tahap evaluasi akan dilakukan perhitungan manual *confusion matrix* untuk menguji hasil klasifikasi dengan hasil sesungguhnya [10]. Selain itu akan ditampilkan hasil *wordcloud* berdasarkan dataset yang sudah melalui proses pengolahan data.

1. Confusion Matrix

Pada tahapan ini dilakukan pengukuran performa permasalahan klasifikasi. Dalam *Confusion matrix* memiliki 4 tabel campuran berbeda pada setiap nilai. [10]. Berikut adalah hasil *confusion matrix* pada tabel 13 dibawah ini :

TABEL XIII
HASIL CONFUSION MATRIX

Prediksi	SVM		NB	
	1	0	1	0
1	427	1	427	13

- Machine (Studi Kasus: Aplikasi Sambara),” *Sistemasi*, vol. 9, no. 3, p. 610, 2020.
- [8] H. P. Doloksaribu and Y. T. Samuel, “Komparasi Algoritma Data Mining Untuk Analisis Sentimen Aplikasi Pedulilindungi,” *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 16, no. 1, pp. 1–11, 2022.
- [9] N. M. A. J. Astari, Dewa Gede Hendra Divayana, and Gede Indrawan, “Analisis Sentimen Dokumen Twitter Mengenai Dampak Virus Corona Menggunakan Metode Naive Bayes Classifier,” *J. Sist. dan Inform.*, vol. 15, no. 1, pp. 27–29, 2020.
- [10] H. Hozairi, A. Anwari, and S. Alim, “Implementasi Orange Data Mining Untuk Klasifikasi Kelulusan Mahasiswa Dengan Model K-Nearest Neighbor, Decision Tree Serta Naive Bayes,” *Netw. Eng. Res. Oper.*, vol. 6, no. 2, p. 133, 2021.