

KOMPARASI ALGORITMA NAÏVE BAYES, SUPPORT VECTOR MACHINE, DAN LOGISTIC REGRESSION PADA ANALISIS SENTIMEN PENGGUNA APLIKASI TRANSPORTASI ONLINE

Krisna Perdana Jaya Sitompul¹, Adi Rizky Pratama², Kiki Ahmad Baihaqi³

Universitas Buana Perjuangan Karawang

Jl. HS.Ronggo Waluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361

¹f19.krisnasitompul@mhs.ubpkarawang.ac.id, ²Adi.Rizky@ubpkarawang.ac.id,

³Kikiahmad@ubpkarawang.ac.id

Abstract

Online transportation is one of the transportation that is increasingly in demand by the public at this time. Grab is an online transportation application that has many users in Indonesia. However, this system certainly has many shortcomings that are felt by users. One way to find out user satisfaction and disappointment with the application is to do sentiment analysis. By analyzing the deficiencies of the application, the company can find out the shortcomings of the application and how to fix it. The purpose of this study is to compare the accuracy between the Support Vector Machine, Naive Bayes, and Logistic Regression algorithms by conducting sentiment analysis on Grab application review data. The results of the comparative test found that the Naive Bayes algorithm has the best performance compared to other classification algorithms with an accuracy obtained by the Naive Bayes algorithm of 88.5%, while the Support Vector Machine algorithm has the lowest accuracy with an accuracy of 85.5%. So it can be concluded that the Naive Bayes algorithm has a better value than the Logistic Regression and Support Vector Machine algorithms.

Keywords: *Grab, Support Vector Machine, Naive Bayes, Logistic Regression*

Abstrak

Transportasi online adalah salah satu transportasi yang semakin diminati masyarakat pada saat ini. Grab adalah salah satu aplikasi transportasi online yang memiliki pengguna bisa dikatakan banyak di Indonesia. Namun dalam system ini pasti memiliki banyak kekurangan yang dirasakan penggunanya. Salah satu cara untuk mengetahui kepuasan dan kekecewaan pengguna terhadap aplikasi tersebut yaitu melakukan analisis sentimen. Dengan menganalisis kekurangan dari aplikasi perusahaan dapat mengetahui kekurangan dari aplikasi dan bagaimana cara memperbaikinya. Tujuan penelitian ini untuk mengetahui perbandingan keakurasian antara algoritma Support Vector Machine, Naive Bayes, dan Logistic Regression dengan melakukan analisis sentimen pada data ulasan aplikasi Grab. Hasil pengujian komparasi ditemukan bahwa algoritma Naive bayes memiliki kinerja terbaik dibandingkan algoritma klasifikasi lainnya dengan akurasi yang di dapat algoritma Naive bayes sebesar 88.5%, sedangkan algoritma Support Vector Machine memiliki akurasi terendah dengan akurasi sebesar 85.5%. Sehingga dapat disimpulkan bahwa algoritma Naive bayes memiliki nilai yang lebih baik dibandingkan algoritma Logistic Regression dan Support Vector Machine.

Kata kunci: *Grab, Support Vector Machine, Naive Bayes, Logistic Regression*

1. PENDAHULUAN

Perkembangan teknologi yang sangat pesat membawa pengaruh yang besar bagi manusia. Salah satu pengaruh dari perkembangan teknologi adalah semakin maraknya penggunaan transportasi *online* melalui media internet. Salah satu transportasi *online* yang ramai dibicarakan oleh masyarakat adalah ojek *online*. Ojek *online* atau sering disebut Ojol merupakan transformasi dari ojek konvensional yang biasanya bertempat di pangkalan untuk menunggu pelanggan. Seiring berjalannya waktu, Ojol ini semakin diminati masyarakat apalagi dengan bantuan *smartphone*. Pelayanan yang disediakan tidak hanya transportasi angkutan penumpang saja, namun juga melayani jasa kurir untuk pemesanan makanan, jasa pembersih untuk bersih-bersih rumah dan kantor, pengiriman barang, dokumen, berbelanja, dan lain sebagainya[1]. Perusahaan transportasi *online* juga kerap dipuji oleh masyarakat, masyarakat menyampaikan tanggapan terhadap transportasi *online* bisa melalui media sosial, *playstore* tempat mengunduh aplikasi transaksi *online* tersebut .

Grab Indonesia sebagai salah satu perusahaan transportasi *online* di Indonesia yang memiliki jumlah pelanggan mencakup hampir seluruh wilayah di Indonesia. Setiap pelanggan memiliki tingkat kepuasan berbeda terhadap layanan yang diberikan oleh Grab Indonesia, sehingga selalu ada saran dan keluhan [2]. Pengguna jasa ojek *online* memberi pendapat dan opini melalui ulasan *google playstore*. Pengalaman pengguna menggunakan aplikasi tergambar menjadi dua bagian, yaitu nilai rating dan komentar. Suatu aplikasi selalu memiliki kekurangan dan kelebihan masing-masing, dimana hal tersebut dapat menimbulkan berbagai respon dari pengguna aplikasi seperti kepuasan dan kekecewaan terhadap aplikasi tersebut. Kolom ulasan komentar menjadi salah satu tempat untuk melontarkan kepuasan dan kekecewaan pengguna atau opini terhadap aplikasi tersebut. Selama ini sebuah predikat untuk aplikasi terbaik diberikan kepada aplikasi dengan jumlah unduhan dan pemberian bintang paling tinggi di *play store*[3].

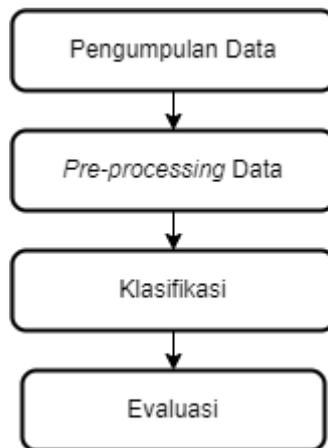
Penelitian terdahulu terkait sentimen dilakukan oleh [4] tentang Komparasi Algoritma Klasifikasi Untuk Menentukan Evaluasi Kinerja Terbaik Pada Status Akreditasi Sekolah/Madrasah Kalimantan Timur Berdasarkan IASP 2020 dengan Hasil komparasi algoritma klasifikasi didapatkan bahwa algoritma SVM memiliki kinerja terbaik dibandingkan dengan algoritma klasifikasi lainnya dengan nilai *accuracy* 88.8%, *precision* 91%, *recall* 81%, dan *f1-score* 84%. Penelitian selanjutnya dilakukan oleh [5] tentang Analisis Sentimen pada *review* Aplikasi Grab di *Google Play Store* Menggunakan *Support Vector Machine* dengan Hasil dari analisis menggunakan *Support Vector Machine* menghasilkan akurasi 85,54% dan Hasil *Review* positif yang paling sering diulas adalah "ovo", sedangkan *review* negatif yang paling sering diulas adalah "driver". Penelitian selanjutnya yaitu Komparasi Algoritma *Naïve Bayes*, *Logistic Regression* Dan *Support Vector Machine* pada Klasifikasi *File Application Package Kit Android Malware* oleh [6] dengan Hasil uji akurasi menunjukkan algoritma *Naive Bayes* mampu mengklasifikasi keluarga malware dengan tingkat akurasi 97.75%, sedangkan algoritma *Logistic Regression* akurasinya 88.75% dan akurasi *Support Vector Machine* mencapai 96,75%. Penelitian terakhir dilakukan oleh [7] tentang Analisis Perbandingan Algoritma *Support Vector Machine*, *Naive Bayes*, dan Regresi Logistik untuk memprediksi

Donor Darah mendapatkan Hasil penelitian dengan algoritma *Naïve Bayes* 89.90%, Regresi Logistik 82.59% dan SVM 94.79%.

Berdasarkan uraian diatas dan beberapa penelitian terdahulu dapat disimpulkan bahwa metode yang menghasilkan akurasi terbaik terdapat pada *Support Vector Machine*. Pada penelitian ini, penulis menggunakan tiga algoritma yaitu *Naïve bayes*, *Logistic Regression* dan *Support Vector Machine* untuk mendapatkan hasil akurasi terbaik dengan mengelompokkan opini positif dan negatif pada ulasan aplikasi Grab. Selain itu penelitian ini juga dilakukan untuk meningkatkan kualitas dan layanan yang ada pada aplikasi grab.

2. METODOLOGI PENELITIAN

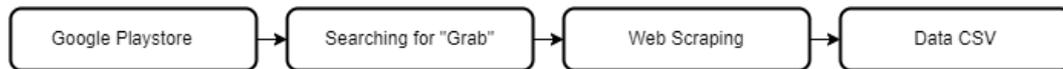
Metode penelitian yang dilakukan terdiri dari beberapa tahap yaitu dimulai dari pengumpulan data, *Pre-processing*, *klasifikasi* menggunakan algoritma *Support Vector Machine*, *Logistic Regression*, dan *Naive Bayes* dan tahap yang terakhir adalah evaluasi. Untuk lebih jelasnya bisa dilihat pada Gambar 1.



Gambar 1. Alur Penelitian

2.1. Pengumpulan Data

Proses pengumpulan data pada penelitian ini yaitu mengambil data dari ulasan aplikasi grab pada *Google Playstore*. Data yang diambil sejumlah 2000 ulasan pengguna dengan kategori *review newest* atau paling terbaru, karena dengan mengambil data terbaru menunjukkan masih adanya pengguna yang aktif sampai saat ini pada aplikasi Grab sehingga komentar yang ditunjukkan masih bersifat *fresh* atau baru. Metode pengambilan data yang digunakan untuk mengambil data pada aplikasi Grab di *Google Playstore* yaitu *Web Scraping*. Tools yang digunakan untuk melakukan *web scraping* pada penelitian ini adalah menggunakan *Google Colab*. Kemudian data *review* yang diperoleh disimpan dalam file bertipe *.csv* agar data yang digunakan dapat terbaca dengan baik pada *read csv*. Terdapat empat informasi yang diperoleh dari hasil *Web Scraping* dari *Google Playstore* yaitu *username*, tanggal dan waktu, *content*, dan skor yang diberikan dari *review* pengguna.



Gambar 2. Pengumpulan Data

2.2. Pre-Processing Data

Pada tahap *Pre-processing* data, data mentah terlebih dahulu dilakukan dengan proses *case folding, tokenizing, stemming, stopwords removal, serta TF-IDF*. *Pre-processing* menghindari data yang tidak teratur, data tidak sempurna dan data yang tidak konsisten[8].

2.3. Klasifikasi

Klasifikasi merupakan salah satu tugas utama dalam *machine learning* dan data *mining* yang termasuk kedalam tipe *supervised learning*. Istilah klasifikasi didapat dari tujuan utama teknik ini yaitu untuk memprediksi sebuah kategori dari masukkan data[9]. Tujuan klasifikasi ini untuk proses perhitungan yang menerapkan algoritma data *mining* untuk mengolah dataset ulasan dari aplikasi Grab pada *Google Playstore*. Terdapat beberapa algoritma yang umum digunakan untuk klasifikasi data *mining* antara lain *Naives Bayes, Support Vector Machine (SVM)*, dan *Logistic Regression*.

2.3.1. Naïve Bayes

Algoritma klasifikasi *Naïve Bayes* merupakan salah satu algoritma teknik klasifikasi, algoritma ini memanfaatkan metode probabilitas dan statistik. Algoritma ini merupakan salah satu metode klasifikasi yang populer dan termasuk dalam sepuluh algoritma teratas dalam data *mining* menurut *IEEE International Conference on Data Mining* di Hongkong [10]. Persamaannya dapat dilihat dibawah ini.

$$P(H|X) = \frac{P(H)P(X|H)}{P(X)}$$

Keterangan :

X = Data kelas tidak diketahui

H = Data hipotesis X adalah kelas yang terpisah

P(H|X) = Probabilitas hipotesis H berdasarkan kondisi (*posteriori probability*)

P(H) = Probabilitas hipotesis H

P(X|H) = Probabilitas X berdasarkan kondisi pada hipotesis

P(X) = Probabilitas X

2.3.2. Support Vector Machine (SVM)

Algoritma *Support Vector Machine* merupakan salah satu teknik *machine learning* yang populer untuk klasifikasi teks serta memiliki performa yang baik pada banyak domain. Kemampuan SVM dalam mengidentifikasi *hyperplane* secara terpisah diantara dua kelas berbeda sehingga termaksimalkan dan SVM menjamin untuk memaksimalkan jarak antara data yang paling dekat dengan *hyperplane*[11].

Dalam penelitian ini penulis menggunakan rumus kernel *linear* untuk menentukan nilai akurasi pada metode ini. Berikut persamaan kernel *linear*.

Kernel	Persamaan
Linear	$K(x_i, x_j) = x_i^T x_j$

Tabel 1. Rumus Kernal

2.3.3. Logistic Regression

Logistic Regression adalah jenis regresi analisis yang digunakan untuk menjelaskan hubungan antara variabel dependen dan variabel independen menghubungkan satu atau lebih variabel bebas dengan variabel terikat jenis kategori bisa 0 dan 1, benar atau salah, besar atau kecil. Tipe variabel bebasnya berupa kategori. Inilah yang membedakan regresi logistik dari regresi berganda atau regresi *linear* lainnya [12]. Persamaan *Logistic Regression* dinyatakan dalam Persamaan.

$$\ln \left(\frac{p}{1-p} \right) = B_0 + B_1 X$$

Keterangan :

B_0 = Konstanta

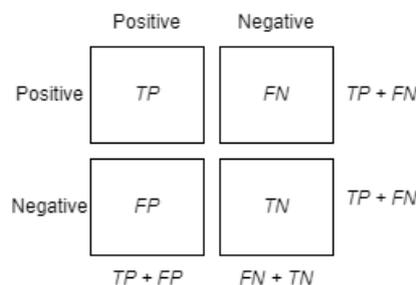
B_1 = Koefisien dari masing-masing variabel

Nilai p atau peluang ($Y = 1$) dapat dicari dengan persamaan

$$p = \frac{e^{(B_0 + B_1 X)}}{(1 + e^{(B_0 + B_1 X)})}$$

2.4. Evaluasi

Tahap ini dilakukan untuk memastikan bahwa tes itu benar. Evaluasi ini adalah tentang menemukan hasil terbaik dari hasil uji[13]. Pengukuran akurasi terhadap model menggunakan *confusion matrix*. *Confusion matrix* adalah alat yang berfungsi untuk menganalisis seberapa model klasifikasi mengenali tuple dari data yang berbeda [14].



Gambar 3. Confusion Matrix

Keterangan :

TP (*True Positive*) = Data positif diklasifikasikan secara benar

TN (*True Negative*) = Data negatif diklasifikasikan secara benar

FP (*False Positive*) = Data negatif diklasifikasikan secara positif
 FN (*False Negative*) = Data positif diklasifikasikan secara negatif

3. HASIL DAN PEMBAHASAN

3.1. Pengumpulan Data

Proses pengambilan data ulasan aplikasi Grab yang ada pada *Google Play* menggunakan Proses *Web Scraping* dengan memanfaatkan *Google Colab* didapatkan sebanyak 2000 data. Dataset dapat dilihat pada Gambar 4.

no	userName	score	at	content
1	Soleh Abdullah	5	2022-11-17 09:36:05	Bismillah
2	Anas Plus	4	2022-11-17 09:33:26	So so lah
3	sandik rahman	5	2022-11-17 09:25:36	Tepat cepat sekali
4	Imam Syamsudin Arifin	1	2022-11-17 09:24:33	"Grab skrg mau cari untung banyak yaaa :) Saya beli makanan pembayaran pake kartu BCA (non tunai)
5	Badar alam Sarohman	5	2022-11-17 09:21:50	aplikasi bagus banget
6	Jafar Sodik	1	2022-11-17 09:17:48	"Mo order grabfood, liat ongkirnya aj males banget.. masa ongkir nya gk masuk akal tarifnya.."
7	rachmad nindo	5	2022-11-17 09:15:27	sangat membantu
8	Fajar Mukti	1	2022-11-17 09:07:13	Tolong dong di kasih tau driver kalo njemput jangan berjam ² . Kalo ngga di kasih pengaturan biar bisa ngripot
9	Echo Firmans	5	2022-11-17 09:02:51	ok
10	maryuni maryuni	5	2022-11-17 09:01:11	recomended
...	Yulia Sumanti	5	2022-11-17 08:58:21	bagus
2000	Pratiwi Ritonga	5	2022-11-17 08:50:34	good

Gambar 4. Isi Dataset

Setelah dataset di dapatkan kemudian data tersebut dilabel kan menjadi dua kelas yaitu Positif dan Negatif, dimana nilai Negatif memiliki score 1-2 dan nilai Positif 3-5.

	content	score	Label
0	Bismillah	5	POSITIF
1	So so lah	4	POSITIF
2	Tepat cepat sekali	5	POSITIF
3	Grab skrg mau cari untung banyak yaaa :) Saya ...	1	NEGATIF
4	aplikasi bagus banget	5	POSITIF

Gambar 5. Pelabelan Data

3.2. Pre-Processing

Setelah melakukan pengumpulan data, proses selanjutnya yaitu melakukan pembersihan data dengan cara *pre-processing*. Berikut tahapan *pre-processing* dalam penelitian ini.

a. Case Folding

Tahapan ini mengubah data dari huruf besar menjadi huruf kecil, menghapus semua *emoticon* yang ada pada data, menghapus angka, tanda baca maupun karakter yang ada pada data. Bisa dilihat pada Tabel 2.

Sebelum	Sesudah
---------	---------

Saya suka banget apk ini jadi lebih gampang berbelanja.	saya suka banget apk ini jadi lebih gampang berbelanja.
6s8aâµyâ8ang â¶7.	Sayang.

Tabel 2. Hasil *Case Folding*

b. *Tokenizing*

Tahap berikutnya setelah melakukan *Case Folding* yaitu tahap *Tokenizing*. Pada tahapan ini data dipecah dari yang tadinya berupa kalimat, paragraf atau dokumen menjadi token-token atau bagian-bagian tertentu berdasarkan tiap kata. Bisa dilihat pada Tabel 3.

Sebelum	Sesudah
aplikasi bagus banget	['aplikasi', 'bagus', 'banget']
Wah sangat lemot	['wah', 'sangat', 'lemot']

Tabel 3. Hasil *Tokenizing*

c. *Stopwords Removal*

Tahap selanjutnya yaitu melakukan *Stopwords Removal*. Tahapan ini akan melakukan penyaringan kata atau melakukan pengurangan kata yang ada pada data. Kata yang disaring berupa kata yang dirasa tidak cukup penting dan tidak berpengaruh. Bisa dilihat pada Tabel 4.

Sebelum	Sesudah
Tolong ditingkatkan lagi, tolong banget udah 2 jam makanan saya ga nyampe, nunggu Asam lambung naik, TOLOL	['tolong', 'ditingkatkan', 'lagi', 'banget', 'udah', 'jam', 'makanan', 'saya', 'ga', 'nyampe', 'nunggu', 'asam', 'lambung', 'naik']

Tabel 4. Hasil *Stopwords Removal*

d. *Stemming*

Berikut adalah tahap *Stemming*, Dimana pada tahapan ini untuk menghilangkan semua imbuhan yang ada pada kata tersebut menjadi kata dasar. Sebagai contoh imbuhan yaitu : meng-, me-, kan-, di-, pe-, peng-, dan lain-lain. *Stemming* dilakukan untuk mengurangi variasi kata yang memiliki kata dasar sama. Bisa dilihat pada Tabel 5.

Sebelum	Sesudah
ovo ny ga bsa di gabung dengan pembayaran tunai .	['ovo', 'ny', 'ga', 'bsa', 'di', 'gabung', 'dengan', 'bayar', 'tunai'].
Kebanyakan update teruss.	['banyak', 'update', 'terus'].

Tabel 5. Hasil *Stemming*

e. TF-IDF

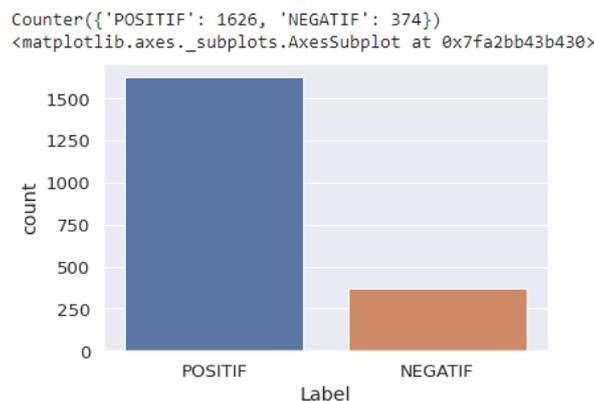
Pembobotan kata dilakukan untuk mendapatkan nilai kata atau istilah yang berhasil diekstraksi. Proses pembobotan kata yang bertujuan untuk menghitung bobot pada masing-masing kata yang akan digunakan sebagai fitur, semakin banyak dokumen yang akan diproses maka semakin banyak fitur.

term	TF	TF-IDF
bisa	0.16666666666666666	0.4689017861266727
cepat	0.16666666666666666	0.5762946221917582
kejar	0.16666666666666666	1.1512925464970227
waktu	0.16666666666666666	0.6323733282952937
pakai	0.16666666666666666	0.6944858758428226
grab	0.16666666666666666	0.3914381719159087

Gambar 6. Hasil TF – IDF

3.3. Klasifikasi

Setelah proses *pre-processing* selesai selanjutnya yaitu melakukan klasifikasi. Proses ini dilakukan menggunakan *Google Colab* dengan Bahasa Pemrograman *Python* serta menggunakan algoritma *Support Vector Machine*, *Logistic Regression* dan *Naïve Bayes*. Pada proses klasifikasi ini data yang diambil sebanyak 2000 data kemudian data dilabelkan menjadi 2 kelas yaitu Positif dan Negatif. Hasil perbandingan kelas positif dan negatif pada data ini dapat dilihat pada Gambar 7.



Gambar 7. Hasil Klasifikasi Sentimen Positif dan Negatif

Gambar 7 diatas menjelaskan bahwa kelas Positif mendapatkan jumlah sebanyak 1.626 data sedangkan kelas Negatif sebanyak 347 data. Setelah itu data dibagi menjadi data *training* dan data *testing* dengan perbandingan ratio 0.3. Artinya data pada data *training* berjumlah 70% yaitu sebanyak 1.400 data dan data *testing* berjumlah 30% yaitu sebanyak 600 data. Data yang digunakan untuk klasifikasi mencari nilai akurasi terbaik menggunakan algoritma adalah data *testing*. Hasil

akurasi klasifikasi data *testing* yang berisi analisis sentimen pengguna aplikasi Grab menggunakan algoritma *Support Vector Machine*, *Logistic Regression* dan *Naïve Bayes* disajikan pada tabel 6.

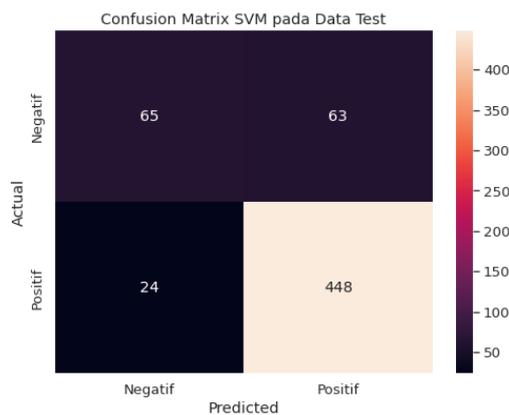
NO	Algoritma	Akurasi(%)	Precision(%)	Recall(%)
1	Naïve Bayes	88.5	0.93	0.93
2	Logistic Regression	85.6	0.87	0.97
3	Support Vector Machine	85.5	0.88	0.95

Tabel 6. Hasil Perbandingan tingkat performa

Berdasarkan tabel 6 diatas Algoritma pertama *Naïve Bayes* mendapatkan akurasi 88.5%, *Precision* 0.93%, *Recall* 0.93%. Algoritma kedua *Logistic Regression* mendapatkan akurasi 85.6%, *Precision* 0.87%, *Recall* 0.97%. Algoritma ketiga *Support Vector Machine* medapatkan akurasi 85.5%, *Precision* 0.88%, *Recall* 0.95%. Dapat disimpulkan bahwa nilai performa akurasi model algoritma *Naive bayes* mendapatkan nilai akurasi yang lebih baik yaitu 88.5% dengan selisih 3% dibandingkan dengan algoritma *Support Vector Machine* sedangkan dengan algoritma *Logistic Regression* mendapatkan selisih 2.9% dan performa akurasi model yang mendapatkan nilai sangat rendah adalah algoritma *Support Vector Machine* mendapatkan model akurasi yaitu 85.5% dengan selisih 3% dibandingkan algoritma *Naive bayes* dan mendapatkan selisih 0.1 pada algoritma *Logistic Regression*.

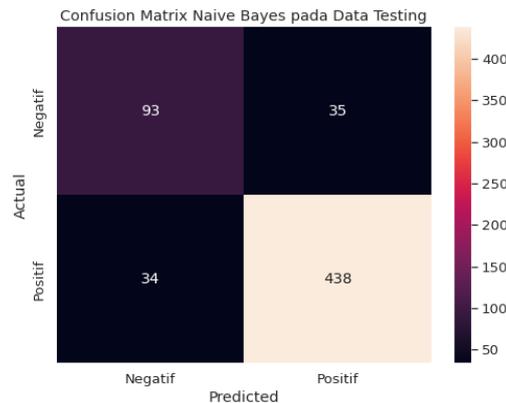
3.4. Evaluasi

Tahap terakhir ini dimana menghitung ulang hasil dari nilai akurasi yang didapat oleh masing-masing algoritma menggunakan *Confusion Matrix*.



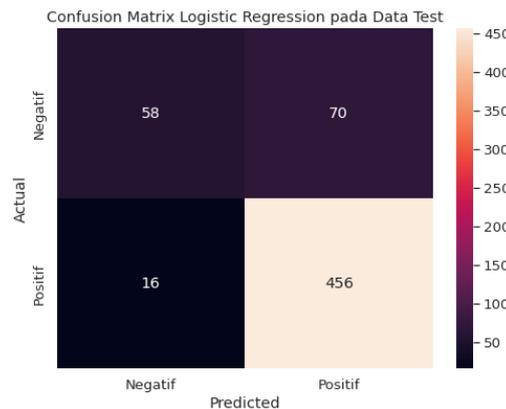
Gambar 8. *Confusion Matrix Support Vector Machine*

Berdasarkan Gambar 9 diatas terlihat bahwa jumlah TP sebanyak 65, FP sebanyak 24, FN Sebanyak 63, dan TN sebanyak 448. Cara menghitung manual nya adalah dengan $TP+FP+FN+TN = 600$, Berikutnya $(TP+TN)/600 = 0.855$. Hasil dari perhitungan *Confusion Matrix Support Vector Machine* adalah 85.5%.



Gambar 9. *Confusion Matrix Naive Bayes*

Gambar 10 diatas terlihat bahwa jumlah TP sebanyak 93, FP sebanyak 34, FN sebanyak 35, dan TN sebanyak 438. Cara menghitung manual nya adalah dengan $TP+FP+FN+TN = 600$, Berikutnya $(TP+TN)/600 = 0.885$. Hasil dari perhitungan *Confusion Matrix Naive Bayes* adalah 88.5%.



Gambar 10. *Confusion Matrix Logistic Regression*

Berikutnya pada Gambar 11 terlihat bahwa jumlah TP sebanyak 58, FP sebanyak 16, FN sebanyak 70, dan TN sebanyak 456. Cara menghitung manual nya yaitu dengan $TP+FP+FN+TN = 600$, Berikutnya $(TP+TN)/600 = 0.856$. Hasil dari perhitungan *Confusion Matrix Logistic Regression* adalah 85.6%.

4. SIMPULAN

Hasil dari pengujian telah ditunjukkan bahwa penggunaan Algoritma *Support Vector Machine, Naive bayes, dan Logistic Regression* dapat digunakan dalam melakukan klasifikasi data analisis sentimen menggunakan aplikasi Grab. Dari hasil pengujian komparasi ditemukan bahwa algoritma *Naive bayes* memiliki kinerja terbaik dibandingkan algoritma klasifikasi lainnya pada penelitian ini, hal ini dibuktikan dengan akurasi yang didapatkan algoritma *Naive bayes* sebesar 88.5%,

sedangkan algoritma *Support Vector Machine* memiliki akurasi terendah yang dibuktikan dengan akurasi sebesar 85.5%. Jumlah data yang digunakan sebanyak 2000 data dari hasil *Web scrapping* pada *Google Playstore*, kemudian data dibagi menjadi 2 yaitu data *training* sebanyak 1400 data dan data *testing* sebanyak 600 data. Sehingga dapat disimpulkan bahwa algoritma *Naive bayes* memiliki kinerja yang lebih baik untuk klasifikasi data analisis sentimen pada Aplikasi Grab dibandingkan algoritma *Logistic Regression* dan *Support Vector Machine*.

DAFTAR PUSTAKA

- [1] S. Mandasari, B. H. Hayadi, and R. Gunawan, "Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD Analisis Sentimen Pengguna Transportasi Online Terhadap Layanan Grab Indonesia Menggunakan Multinomial Naive Bayes Classifier Jurnal Teknologi Sistem Informasi dan Sistem Komputer TGD," vol. 5, pp. 118–126, 2022.
- [2] D. R. Alghifari, M. Edi, and L. Firmansyah, "Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia Bidirectional LSTM Implementation for Sentiment Analysis Against Grab Indonesia Services," vol. 12, pp. 89–99, 2022.
- [3] A. Erfina *et al.*, "Analisis Sentimen Aplikasi Pembelajaran Online Di Play Store Pada Masa Pandemi Covid-19 Menggunakan Algoritma Support Vector Machine," vol. 2020, no. Semasif, pp. 145–152, 2020.
- [4] N. A. V. Taghfirul Azhima Yoga Siswa, "Komparasi Algoritma Klasifikasi Untuk Menentukan Evaluasi Kinerja Terbaik Pada Status Akreditasi Sekolah/Madrasah Kalimantan Timur Berdasarkan IASP 2020," vol. 4, no. 3, pp. 185–192, 2022.
- [5] R. Wahyudi and G. Kusumawardhana, "Analisis Sentimen pada review Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine," vol. 8, no. 2, pp. 200–207, 2021.
- [6] R. E. Indrajit and E. Dazki, "Komparasi Algoritma Naïve Bayes , Logistic Regression Dan Support Vector Machine pada Klasifikasi File Application Package Kit Android Malware".
- [7] S. D. Hendriyana, Ichwanul Muslim Karo Karo, "Analisis Perbandingan Algoritma Support Vector Machine, Naive Bayes, Dan Regresi Logistik Untuk Memprediksi Donor Darah," vol. 8, no. 2, pp. 121–126, 2022.
- [8] I. P. Rahayu, A. Fauzi, and J. Indra, "Analisis Sentimen Terhadap Program Kampus Merdeka Menggunakan Naive Bayes Dan Support Vector Machine," vol. 4, pp. 296–301, 2022, doi: 10.30865/json.v4i2.5381.
- [9] A. Supriyatna and W. P. Mustika, "Komparasi Algoritma Naive bayes dan SVM Untuk Memprediksi Keberhasilan Imunoterapi Pada Penyakit Kutil," no. 2, pp. 152–161, 2018.
- [10] A. P. Giovani, T. Haryanti, and L. Kurniawati, "Analisis Sentimen Aplikasi Ruang Guru Di Twitter Menggunakan Algoritma Klasifikasi," vol. 14, no. 2, pp. 116–124, 2020.
- [11] D. Gunawan, D. Riana, D. Ardiansyah, F. Akbar, and S. Alfarizi, "Komparasi

- Algoritma Support Vector Machine Dan Naïve Bayes Dengan Algoritma Genetika Pada Analisis Sentimen Calon Gubernur Jabar 2018-2023,” vol. VI, no. 1, pp. 121–129, 2023, doi: 10.31294/jtk.v4i2.
- [12] F. D. Pramakrisna, F. D. Adhinata, N. Annisa, and F. Tanjung, “Aplikasi Klasifikasi SMS Berbasis Web Menggunakan Algoritma Logistic Regression Web-based Classifying SMS Application Using Logistic Regression Algorithm,” vol. 11, no. 2, pp. 90–97, 2022, doi: 10.34148/teknika.v11i2.466.
- [13] A. S. Rahayu and A. Fauzi, “Komparasi Algoritma Naïve Bayes Dan Support Vector Machine (SVM) Pada Analisis Sentimen Spotify,” vol. 4, pp. 349–354, 2022, doi: 10.30865/json.v4i2.5398.
- [14] M. Rangga, A. Nasution, and M. Hayaty, “Perbandingan Akurasi dan Waktu Proses Algoritma K-NN dan SVM dalam Analisis Sentimen Twitter,” vol. 6, no. 2, pp. 226–235, 2019.