

Perbandingan Algoritma Klasifikasi *Support Vector Machine*, *Random Forest* dan *Logistic Regression* Pada Ulasan Shopee

Asep Saepudin^{1*}, Ahmad Faqih², Gifthera Dwilestari³

^{1,2}Teknik Informatika, STMIK IKMI CIREBON, Cirebon, Indonesia

³Sistem Informasi, STMIK IKMI CIREBON, Cirebon, Indonesia

Email: ^{1*} saepudinasep2001@gmail.com, ²ahmadfaqih367@gmail.com, ³ggdwilestari@gmail.com

Abstrak– Dengan kemajuan teknologi, penggunaan internet semakin meluas di Indonesia dan turut mendorong perkembangan *e-commerce*. Namun, masih terdapat beberapa permasalahan yang perlu diselesaikan, salah satunya adalah kurangnya kepercayaan konsumen terhadap platform *e-commerce*. Oleh karena itu, penelitian tentang platform *e-commerce* seperti Shopee perlu dilakukan untuk memahami arah sentimen pengguna. Analisis sentimen digunakan untuk mengklasifikasikan sentimen ulasan pengguna Shopee. Penelitian ini bertujuan untuk membandingkan performa algoritma klasifikasi berdasarkan proses data mining menggunakan *CRISP-DM*. Algoritma klasifikasi yang digunakan adalah *Support Vector Machine (SVM)*, *Random Forest*, dan *Logistic Regression*. Pengujian menggunakan 3.000 data, dengan 90% *data training* dan 10% *data testing*. Hasil pelabelan data menunjukkan bahwa sentimen positif berjumlah 307, sentimen netral 2.537, dan sentimen negatif 156. Hasil pengujian akurasi menggunakan *confusion matrix* menunjukkan bahwa algoritma *Random Forest* memiliki akurasi tertinggi sebesar 94%, diikuti oleh *Support Vector Machine (SVM)* sebesar 91%, dan *Logistic Regression* sebesar 86%. Penelitian ini berhasil mendapatkan algoritma yang efektif dan terbaik dalam mengklasifikasikan sentimen ulasan positif, sentimen ulasan netral, dan sentimen ulasan negatif terkait dengan aplikasi Shopee. Hasil dari algoritma terbaik tersebut akan diintegrasikan dalam antarmuka web menggunakan bahasa pemrograman PHP dan Python, untuk mengetahui arah sentimen yang akan dimasukkan di masa mendatang.

Kata Kunci: *e-commerce*, sentimen pengguna, analisis sentimen, algoritma klasifikasi, shopee.

Abstract– With advances in technology, internet use is increasingly widespread in Indonesia and has contributed to the development of *e-commerce*. However, there are still several problems that need to be resolved, one of which is the lack of consumer trust in *e-commerce* platforms. Therefore, research on *e-commerce* platforms such as Shopee needs to be carried out to understand the direction of user sentiment. Sentiment analysis is used to classify the sentiment of Shopee user reviews. This research aims to compare the performance of classification algorithms based on data mining processes using *CRISP-DM*. The classification algorithms used are *Support Vector Machine (SVM)*, *Random Forest*, and *Logistic Regression*. Testing uses 3,000 data, with 90% training data and 10% testing data. The data labeling results show that there are 307 positive sentiments, 2,537 neutral sentiments, and 156 negative sentiments. The results of accuracy testing using the confusion matrix show that the *Random Forest* algorithm has the highest accuracy at 94%, followed by *Support Vector Machine (SVM)* at 91%, and *Logistic Regression* is 86%. This research succeeded in obtaining an effective and best algorithm in classifying positive review sentiment, neutral review sentiment, and negative review sentiment related to the Shopee application. The results of the best algorithm will be integrated into a web interface using the PHP and Python programming languages, to determine the direction of sentiment that will be entered in the future.

Keywords: *e-commerce*, user sentiment, sentiment analysis, classification algorithms, shopee.

1. PENDAHULUAN

Seiring dengan kemajuan teknologi, pengguna internet semakin meluas di Indonesia, yang mempengaruhi perkembangan *e-commerce*. Hal ini disebabkan karena sebagian besar orang memiliki mobilitas yang tinggi dalam kegiatan sehari-harinya [1]. Menurut data dari Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), jumlah pengguna internet di Indonesia mencapai 213 juta orang per Januari 2023 dan diperkirakan akan terus meningkat pada tahun-tahun mendatang. Hal ini memberikan potensi besar bagi pengembangan *e-commerce* di Indonesia. Semakin banyak pelaku ekonomi, baik perusahaan besar maupun kecil, yang beralih ke bisnis digital dalam bentuk *e-commerce*. [2]. Namun, untuk memanfaatkan potensi ini, perlu adanya upaya untuk meningkatkan kepercayaan konsumen terhadap platform *e-commerce* serta mengetahui arah sentimen pengguna terhadap produk atau layanan yang ditawarkan [2].

Shopee saat ini menjadi platform belanja dan jual beli paling banyak dikunjungi di Indonesia. Rata-rata pengunjung bulanannya mencapai 216 juta pada kuartal ketiga tahun 2023. Shopee dapat diunduh dari Google Play Store, yang menyediakan berbagai fitur penilaian dan ulasan [3]. Salah satu fitur yang tersedia adalah fitur rating dan ulasan, yang memungkinkan pengguna untuk berbagi pendapat mereka tentang produk yang telah mereka beli [4]. Review dari pengguna sebelumnya menjadi sumber informasi yang semakin penting untuk menilai kualitas produk dan layanan. Banyak pengguna internet yang mengandalkan rekomendasi dan pendapat pengguna sebelumnya dalam membuat keputusan pembelian [3].

Bagian dari *data mining*, disebut juga analisis sentimen atau *opinion mining*, yang sering digunakan untuk menganalisis teks untuk mendapatkan opini yang terpolarisasi dan kemudian mengubah data tersebut

menjadi informasi yang bernilai positif, negatif, atau netral [5]. Tugas dasar analisis sentimen adalah dengan mengumpulkan data atau teks yang ada menjadi sebuah kalimat atau dokumen, yang kemudian mengkategorikan kalimat atau dokumen tadi menjadi bentuk positif, negatif atau netral [5].

Sebelumnya, telah dilakukan penelitian mengenai perbandingan kinerja algoritma *Naive Bayes*, *K-Nearest Neighbors*, dan *Random Forest* untuk klasifikasi sentimen terhadap BPJS kesehatan di Twitter. Hasil penelitian menunjukkan bahwa algoritma *Random Forest* memiliki akurasi tertinggi, yaitu 86%, diikuti oleh *Naive Bayes* dengan akurasi 80%, dan *K-Nearest Neighbors* dengan akurasi 67% [6].

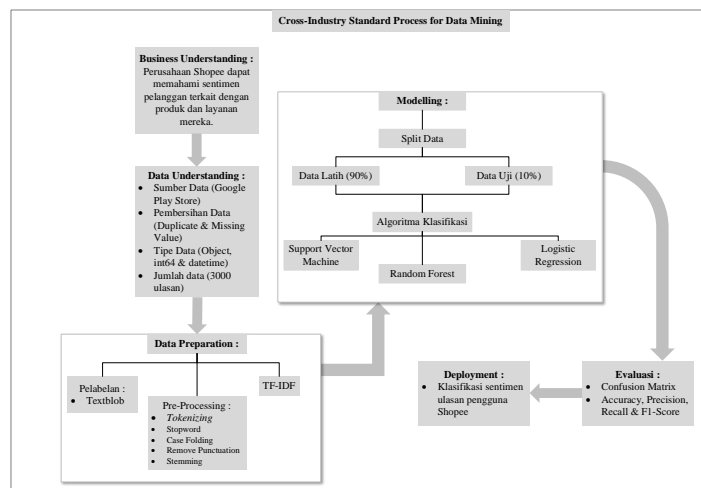
Topik ini sangat menarik untuk dibahas, sehingga banyak penelitian yang dilakukan di bidang analisis sentimen. Salah satu penelitian tersebut mempelajari analisis sentimen ulasan meetings cloud Zoom di Play Store menggunakan dua algoritma, yaitu *Naive Bayes* dan *Support Vector Machine*. Pada penelitiannya didapatkan bahwa tingkat akurasi yang didapatkan algoritma *Support Vector Machine* lebih unggul 6,85% dibandingkan algoritma *Naive Bayes* dengan 1007 record dataset [7]. Penelitian lainnya tentang analisis sentimen opini masyarakat terkait penyelenggaraan sistem elektronik menggunakan metode *Logistic Regression*. Hasil penelitian tersebut menunjukkan bahwa analisis sentimen menggunakan algoritma *Logistic Regression* mencapai akurasi sebesar 79,07% pada dataset opini masyarakat terhadap kebijakan Operasi Sistem Elektronik (PSE). Dengan sebaran sentimen pada dataset adalah 126 sentimen negatif, 657 sentimen netral, dan 291 sentimen positif [8].

Berdasarkan penelitian sebelumnya, penelitian ini menggunakan tiga algoritma, yaitu *Support Vector Machine*, *Random Forest*, dan *Logistic Regression*, untuk melakukan analisis sentimen. Algoritma ini umum dan dapat digunakan untuk mengklasifikasi data opini Google Playstore. Proses ini akan memberikan label kelas untuk opini positif, netral, atau negatif [3]. Algoritma tersebut akan dibandingkan dan dievaluasi menggunakan *confusion matrix*. Temuan penelitian ini diharapkan dapat memberikan hasil klasifikasi yang akurat dan informasi yang bermanfaat bagi pihak yang memerlukan.

2. METODE PENELITIAN

2.1 CRISP-DM

Penelitian klasifikasi ulasan pengguna shopee disesuaikan dengan tahapan metode *CRISP-DM* dengan diagram alir yang telah dibuat agar penelitian yang dilakukan dapat lebih terstruktur dan hasil yang didapatkan dapat lebih akurat dan maksimal. Diagram alir yang digunakan oleh peneliti dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian sesuai CRISP-DM

a. Business Understanding

Dalam tahap ini, tujuan dan kebutuhan penelitian secara rinci diidentifikasi. Selanjutnya, tujuan dan batasan tersebut diterjemahkan menjadi rumusan permasalahan *data mining*. Proses ini melibatkan pemahaman mendalam terhadap objek penelitian guna menghasilkan hasil yang sesuai dengan harapan. Pemahaman terhadap objek penelitian dapat diperoleh dengan mencari dan menggali informasi dari berbagai sumber yang terkait dengan objek tersebut.

b. Data Understanding

Proses yang dilakukan pada tahap ini adalah pengambilan data mentah dari Google Play Store yang dikumpulkan dengan teknik web scraping menggunakan pemrograman python pada tanggal 24 Oktober 2023. Setelah data diperoleh, dilakukan proses pembersihan data untuk mengatasi *missing value* dan *duplicate data*. Setelah melakukan pembersihan data, dilakukan eksplorasi data untuk memahami struktur data. Eksplorasi ini melibatkan analisis statistik deskriptif dan visualisasi data guna mendapatkan wawasan awal mengenai karakteristik dataset yang akan digunakan dalam analisis lebih lanjut.

c. Data Preparation

Dalam tahap *data preparation* ada 3 tahapan yaitu pelabelan data, preprocessing dan pembobotan kata.

1. Pelabelan Data

Teks diklasifikasikan secara otomatis menggunakan *textblob*. *Textblob* menghitung nilai *polarity* dan *subjectivity* untuk menentukan kelas teks. *Polarity* menunjukkan emosi teks, sedangkan *subjectivity* menunjukkan jenis teks[9]. Teks dengan *subjectivity* tinggi adalah opini, sedangkan teks dengan *polarity* tinggi adalah positif[9]. Dari nilai *polarity*, teks diklasifikasikan menjadi positif, negatif, atau netral.

1. Preprocessing

Berikut ini merupakan bentuk *preprocessing* yang dilakukan yaitu :

- *Tokenizing*: Tahap ini adalah proses untuk membagi kalimat menjadi unit-unit terkecil yang disebut token. Token dapat berupa kata, frasa, atau elemen makna lainnya[10]. Pada penelitian ini, proses tokenisasi dapat dilakukan dengan menggunakan *library nltk.tokenize* untuk membagi kata menjadi kumpulan huruf.
- *Stopword*: Proses ini bertujuan untuk menghilangkan kata-kata yang sering digunakan dan tidak memiliki arti penting dalam sebuah kalimat. Hal ini dilakukan agar data dapat diproses lebih efisien pada tahap selanjutnya[10]. Dalam penelitian ini, *stopword* dihilangkan dengan menggunakan daftar *stopword* yang tersedia dalam *library nltk.corpus*.
- *Case Folding*: Pada tahap ini, kata-kata diubah menjadi bentuk yang sama dengan mengubah semua kata menjadi huruf kecil[10].
- *Remove Punctuation*: Tahap ini adalah proses untuk menghilangkan karakter yang tidak diperlukan, seperti simbol, angka, dan tanda baca. Hal ini dilakukan karena ulasan pengguna Google Play Store biasanya berisi simbol, angka, dan tanda baca yang tidak diperlukan untuk proses klasifikasi[10]. Proses ini dilakukan dengan menggunakan sintaks ekspresi reguler pada Python.
- *Stemming*: Tahap ini adalah proses mengubah kata berimbuhan menjadi kata dasar dengan cara membuang imbuhan[10]. Penelitian ini menggunakan pustaka Python *Sastrawi* untuk melakukan proses tersebut.

2. Pembobotan Kata

- *Term Frequency*

Proses ini menghitung berapa kali sebuah kata muncul dalam sebuah kumpulan data.[4]. Frekuensi kemunculan kata dapat dihitung dengan menggunakan persamaan berikut.

$$TF(t, d) = \frac{\text{jumlah kemunculan kata dalam dokumen}}{\text{total kata dalam dokumen}} \quad (1)$$

- *Inverse Document Frequency*

Proses ini menghitung jumlah dokumen yang mengandung kata kunci yang dicari [4]. Perhitungan jumlah dokumen yang mengandung kata kunci dapat dilakukan dengan menggunakan persamaan berikut.

$$IDF(t, D) = \log\left(\frac{N}{df} + 1\right) \quad (2)$$

Keterangan :

N = jumlah semua dokumen

df = jumlah kata pada dokumen

- a. *Modelling*

Sebelum melakukan pemodelan data, terlebih dahulu dilakukan pembagian data dengan *data training* sebesar 90% dan *data testing* sebesar 10%. Setelah itu, *data training* dilakukan *modeling* dengan algoritma klasifikasi *Support Vector Machine*, *Random Forest* dan *Logistic Regression*. Dan data uji dilakukan proses klasifikasi menggunakan algoritma yang telah dilakukan modeling tersebut.

- b. *Evaluation*

Tahap ini adalah tahap uji model menggunakan *confusion matrix*. *Confusion matrix* adalah matriks berukuran 3x3 yang digunakan untuk mengevaluasi kinerja model dengan *data training* dan *data testing*[4]. *Confusion matrix* dapat menghasilkan nilai accuracy, recall, precision, dan f1-score. Nilai-nilai ini dapat dihitung dan ditampilkan dalam bentuk persentase[4]. Tabel *confusion matrix* dapat dilihat sebagai berikut :

Tabel 1. Confusion Matrix

Data Aktual	Data Prediksi		
	Positif	Netral	Negatif
Positif	True Positif	False Netral	False Negatif
Netral	False Positif	True Netral	False Negatif
Negatif	False Positif	False Netral	True Negatif

$$Accuracy = \frac{TP+TN+TNe}{TP+FN+FNe+FP+TN+TNe} \quad (3)$$

Keterangan :

TP = True Positif FP = False Positif
 FN = False Negatif TN = True Negatif
 FNe = False Netral TNe = True Netral

c. Deployment

Tahap terakhir dari CRISP-DM adalah tahap diseminasi hasil. Hasil yang diperoleh dari proses *data mining* perlu diorganisir dan disajikan dengan cara yang mudah dipahami oleh pengguna.

2.2 Support Vector Machine

Support vector machine (SVM) adalah metode pembelajaran terbimbing yang digunakan untuk klasifikasi dan analisis data [3]. *SVM* adalah algoritma klasifikasi yang populer dan dapat digunakan untuk mengklasifikasikan data ke dalam dua atau lebih kelas. *SVM* menggunakan teknik kernel untuk memetakan data ke dalam ruang berdimensi tinggi, sehingga dapat menangani masalah klasifikasi *non-linear* [3]. Rumus untuk perhitungan SVM menggunakan persamaan (4) :

$$(w \cdot xi) + b = 0 \quad (4)$$

Data xi yang termasuk pada kelas -1 dapat direpresentasikan dengan persamaan (5).

$$(w \cdot xi+b) \leq yi = -1 \quad (5)$$

Untuk data xi yang termasuk pada kelas +1, rumusnya dapat dilihat pada persamaan (6).

$$(w \cdot xi+b) \geq yi=1 \quad (6)$$

2.3 Random Forest

Random Forest adalah algoritma yang dapat digunakan untuk klasifikasi dan regresi. *Random Forest* terdiri dari banyak pohon keputusan yang dilatih secara independen. Hasil klasifikasi ditentukan oleh suara terbanyak dari pohon-pohon tersebut. *Random Forest* menghasilkan hasil yang kompetitif dengan algoritma lain, tetapi tidak mengubah data pelatihan secara progresif. Akurasi yang ditunjukkan oleh *Random Forest* adalah untuk mengurangi bias[11].

$$Gini = N_L \sum_k PkL (1 - PkL) + N_R \sum_k PkR (1 - PkR) \quad (7)$$

Random Forest menggunakan indeks Gini untuk memilih atribut yang paling memisahkan data berdasarkan kelasnya[11]. Indeks Gini mengukur kemungkinan kesalahan klasifikasi, dan nilainya akan semakin kecil jika data lebih tersegmentasi. Indeks Gini dapat dihitung menggunakan persamaan (7), di mana k mewakili kategori kejadian, PkL mewakili proporsi kelas k di simpul kiri, dan PkR mewakili proporsi kelas k di simpul kanan[11].

2.4 Logistic Regression

Logistic regression adalah suatu metode statistik yang efektif dalam menganalisis *data multivariat* dan dalam memodelkan *variabel biner*. Dalam analisis *multivariat*, *Logistic Regression* memungkinkan peneliti untuk memperkirakan koefisien pada setiap prediktor yang dimasukkan dalam model akhir. Lebih lanjut, metode ini

memungkinkan penyesuaian prediktor satu sama lain, sehingga memungkinkan penentuan pengaruh relatif masing – masing prediktor terhadap *variabel* respon[5].

Dalam konteks analisis sentimen, *Logistic Regression* digunakan untuk memahami bagaimana berbagai fitur atau variabel independen (seperti kata – kata kunci atau fitur-fitur dalam ulasan) dapat memengaruhi kecenderungan ulasan untuk menjadi positif, negatif, atau netral. Melalui estimasi koefisien, *Logistic Regression* membantu mengukur sejauh mana setiap prediktor memengaruhi probabilitas ulasan termasuk dalam kategori sentimen tertentu.

Logistic regression memiliki fungsi dalam menentukan beberapa fungsi linier yang dapat dinyatakan sebagai berikut[5]:

$$\text{Logit } (P) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 \dots \dots \beta_k X_k \tag{8}$$

P = probabilitas fitur
 X1, X2, ... Xk = nilai prediktor
 β1, β2, ... βk = intersepsi model

3. HASIL DAN PEMBAHASAN

3.1 Hasil

3.1.1 Business Understanding

Pemahaman bisnis pada kasus perbandingan algoritma klasifikasi pada sentimen ulasan pengguna Shopee adalah sebagai berikut:

- a. Perusahaan Shopee dapat memahami sentimen pelanggan terkait dengan produk dan layanan mereka.
- b. Menggunakan algoritma klasifikasi untuk mengkategorikan ulasan pengguna sebagai positif, negatif, atau netral.
- c. Meningkatkan pemahaman Shopee terhadap sentimen pelanggan, memungkinkan perbaikan layanan dan produk, serta pengambilan keputusan berdasarkan ulasan pelanggan.

3.1.2 Data Understanding

Data yang dikumpulkan dari Google Playstore menggunakan teknik web *scrapping* didapatkan dengan atribut sebagai berikut.

Tabel 2. Atribut Dataset

Atribut	Deskripsi
reviewId	Id Ulasan
userName	Nama Pengguna
userImage	Photo Pengguna
Content	Ulasan Pengguna
Score	Nilai Ulasan
thumbsUpCount	Parameter Jumlah Suka
reviewCreatedVersion	Ulasan Versi Aplikasi
At	Tanggal Dibuatnya Ulasan
replyContent	Umpan Balik/Balasan Ulasan
repliedAt	Tanggal Umpan Balik/Balasan Ulasan
appVersion	Versi Aplikasi

Selanjutnya, data tersebut dilakukan pengecekan *missing value* dan *duplicate* dengan hasil dari pengecekan tersebut tidak didapatkan *duplicate* data dan Hasilnya terdapat 977 *missing value* pada atribut *replyContent* dan *repliedAt*. Maka untuk mengatasi hal tersebut peneliti melakukan pengisian data pada nilai yang kosong dengan "0" dan dengan kode pada Gambar 2.

```

Data Cleaning

# Mengecek missing value
df.isnull().sum()
# Mengisi missing value dengan 0
df.fillna(0, inplace=True)
# mengecek kembali missing value
df.isnull().sum()
    
```

Gambar 2. Missing Value

Dataset ulasan pengguna Shopee, terdiri dari total 3000 data, menunjukkan bahwa top 10 versi aplikasi berdasarkan ulasan pengguna mencakup versi 3.08.12, 3.06.14, 3.11.08, 3.09.20, 3.08.08, 3.07.09, 3.11.30, 3.11.12, 3.09.10, dan 3.06.12. Versi aplikasi 3.08.12 mendapatkan ulasan terbanyak dengan total 597 ulasan pengguna Shopee, sedangkan versi aplikasi 3.06.12 memperoleh ulasan paling sedikit dengan hanya 91 ulasan. Informasi ini memberikan gambaran tentang distribusi versi aplikasi dan seberapa umum atau jarang suatu versi menerima ulasan dari pengguna Shopee.



Gambar 3. Top 10 Versi Aplikasi Berdasarkan Ulasan

3.1.3 Data Preparation

a. Pelabelan Data

Sebelum melakukan preprocessing dataset, dilakukan pelabelan otomatis menggunakan library TextBlob. Pelabelan data dihitung dengan memanfaatkan TextBlob dengan rincian data dengan label positif berjumlah 307, data dengan label negatif berjumlah 156, dan data dengan label netral berjumlah 2537. Hasil dari pelabelan ini dapat dilihat pada Gambar 4 :

	content	sentiment	polarity	subjectivity
0	Aplikasi lemot bahkan ngreset sendiri, udah n...	Positif	0.150000	0.65
1	Shopee dari dulu ngga ada perubahan sama sekal...	Netral	0.000000	0.00
2	Yang paling menjengkelkan dari Shopee itu sela...	Netral	0.000000	0.00
3	Dear shopee Pada saat checkout mohon bijak dal...	Netral	0.000000	0.00
4	Jaringan tak stabil, padahal jaringan aslinya ...	Netral	0.000000	0.00
5	Udah males belanja di shopee sekarang. Gak bis...	Netral	0.000000	0.00
6	Aplikasi pengganggu, nelpon berkali-kali menaw...	Netral	0.000000	0.00
7	Tolong perbaiki lg aplikasinya.. Masa shopeepa...	Positif	0.136364	0.50
8	Aplikasi lemot seharusnya tidak perlu diberi f...	Positif	0.136364	0.50
9	Setelah di update lagi, malah jadi berat, erro...	Netral	0.000000	0.00

Gambar 4. Pelabelan Data

a. Preprocessing

Preprocessing merupakan tahap pengolahan data yang telah dikumpulkan menjadi satu dokumen, yang kemudian akan menjalani proses analisis. *Preprocessing* ini terbagi menjadi beberapa bagian, yaitu:

1. Tokenizing

Proses untuk memecah kata menjadi beberapa bagian, dan hasilnya disebut token. Berikut merupakan hasil dari proses *Tokenizing*:

Tabel 3. Hasil Tokenizing

Sebelum	Sesudah
Shopee dari dulu nnga ada perubahan sama sekali katanya suruh update sampe yg terbaru pun nggak ada bedanya tetep berat... padahal wifi stabil, paket data stabil, log out sudah, hapus data juga sudah, tapi hasilnya tetap nihil. Lag.. bikin hp jadi cepat panas. Solusinya katanya harus chat admin namun tetep ga ada solusi yg work. Mohon di oprek lagi apk nya agar lebih enteng dan hemat data baru aku kasih bintang 5...	['Shopee', 'dari', 'dulu', 'ngga', 'ada', 'perubahan', 'sama', 'sekali', 'katanya', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'pun', 'nggak', 'ada', 'bedanya', 'tetep', 'berat', '...', 'padahal', 'wifi', 'stabil', ',', 'paket', 'data', 'stabil', ',', 'log', 'out', 'sudah', ',', 'hapus', 'data', 'juga', 'sudah', ',', 'tapi', 'hasilnya', 'tetap', 'nihil', ',', 'Lag', '..', 'bikin', 'hp', 'jadi', 'cepat', 'panas', ',', 'Solusinya', 'katanya', 'harus', 'chat', 'admin', 'namun', 'tetep', 'ga', 'ada', 'solusi', 'yg', 'work', ',', 'Mohon', 'di', 'oprek', 'lagi', 'apk', 'nya', 'agar', 'lebih', 'enteng', 'dan', 'hemat', 'data', 'baru', 'aku', 'kasih', 'bintang', '5', '...']

2. Case Folding

Proses di mana semua huruf diubah menjadi huruf kecil. Berikut merupakan hasil dari proses Case Folding:

Tabel 4. Hasil Case Folding

Sebelum	Sesudah
['Shopee', 'dari', 'dulu', 'ngga', 'ada', 'perubahan', 'sama', 'sekali', 'katanya', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'pun', 'nggak', 'ada', 'bedanya', 'tetep', 'berat', '...', 'padahal', 'wifi', 'stabil', ',', 'paket', 'data', 'stabil', ',', 'log', 'out', 'sudah', ',', 'hapus', 'data', 'juga', 'sudah', ',', 'tapi', 'hasilnya', 'tetap', 'nihil', ',', 'Lag', '..', 'bikin', 'hp', 'jadi', 'cepat', 'panas', ',', 'Solusinya', 'katanya', 'harus', 'chat', 'admin', 'namun', 'tetep', 'ga', 'ada', 'solusi', 'yg', 'work', ',', 'Mohon', 'di', 'oprek', 'lagi', 'apk', 'nya', 'agar', 'lebih', 'enteng', 'dan', 'hemat', 'data', 'baru', 'aku', 'kasih', 'bintang', '5', '...']	['shopee', 'dari', 'dulu', 'ngga', 'ada', 'perubahan', 'sama', 'sekali', 'katanya', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'pun', 'nggak', 'ada', 'bedanya', 'tetep', 'berat', '...', 'padahal', 'wifi', 'stabil', ',', 'paket', 'data', 'stabil', ',', 'log', 'out', 'sudah', ',', 'hapus', 'data', 'juga', 'sudah', ',', 'tapi', 'hasilnya', 'tetap', 'nihil', ',', 'lag', '..', 'bikin', 'hp', 'jadi', 'cepat', 'panas', ',', 'solusinya', 'katanya', 'harus', 'chat', 'admin', 'namun', 'tetep', 'ga', 'ada', 'solusi', 'yg', 'work', ',', 'mohon', 'di', 'oprek', 'lagi', 'apk', 'nya', 'agar', 'lebih', 'enteng', 'dan', 'hemat', 'data', 'baru', 'aku', 'kasih', 'bintang', '5', '...']

3. Remove Punctuation

Proses untuk menghapus tanda baca dan simbol dari dataset. Berikut adalah hasil dari proses tersebut:

Tabel 5. Hasil Remove Punctuation

Sebelum	Sesudah
['shopee', 'dari', 'dulu', 'ngga', 'ada', 'perubahan', 'sama', 'sekali', 'katanya', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'pun', 'nggak', 'ada', 'bedanya', 'tetep', 'berat', '...', 'padahal', 'wifi', 'stabil', ',', 'paket', 'data', 'stabil', ',', 'log', 'out', 'sudah', ',', 'hapus', 'data', 'juga', 'sudah', ',', 'tapi', 'hasilnya', 'tetap', 'nihil', ',', 'lag', '..', 'bikin', 'hp', 'jadi', 'cepat', 'panas', ',', 'solusinya', 'katanya', 'harus', 'chat', 'admin', 'namun', 'tetep', 'ga', 'ada', 'solusi', 'yg', 'work', ',', 'mohon', 'di', 'oprek', 'lagi', 'apk', 'nya', 'agar', 'lebih', 'enteng', 'dan', 'hemat', 'data', 'baru', 'aku', 'kasih', 'bintang', '5', '...']	['shopee', 'dari', 'dulu', 'ngga', 'ada', 'perubahan', 'sama', 'sekali', 'katanya', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'pun', 'nggak', 'ada', 'bedanya', 'tetep', 'berat', 'padahal', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'sudah', 'hapus', 'data', 'juga', 'sudah', 'tapi', 'hasilnya', 'tetap', 'nihil', 'lag', 'bikin', 'hp', 'jadi', 'cepat', 'panas', 'solusinya', 'katanya', 'harus', 'chat', 'admin', 'namun', 'tetep', 'ga', 'ada', 'solusi', 'yg', 'work', 'mohon', 'di', 'oprek', 'lagi', 'apk', 'nya', 'agar', 'lebih', 'enteng', 'dan', 'hemat', 'data', 'baru', 'aku', 'kasih', 'bintang']

4. Stopwords

Penghapusan kata yang tidak penting atau tidak dibutuhkan, seperti kata keterangan dan kata sambung. Berikut merupakan hasil dari proses *Stopwords*:

Tabel 6. Hasil Stopwords

Sebelum	Sesudah
['shopee', 'dari', 'dulu', 'ngga', 'ada', 'perubahan', 'sama', 'sekali', 'katanya', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'pun', 'nggak', 'ada', 'bedanya', 'tetep', 'berat', 'padahal', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'sudah', 'hapus', 'data', 'juga', 'sudah', 'tapi', 'hasilnya', 'tetap', 'nihil', 'lag', 'bikin', 'hp', 'jadi', 'cepat', 'panas', 'solusinya', 'katanya', 'harus', 'chat', 'admin', 'namun', 'tetep', 'ga', 'ada', 'solusi', 'yg', 'work', 'mohon', 'di', 'oprek', 'lagi', 'apk', 'nya', 'agar', 'lebih', 'enteng', 'dan', 'hemat', 'data', 'baru', 'aku', 'kasih', 'bintang']	['shopee', 'ngga', 'perubahan', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'nggak', 'bedanya', 'tetep', 'berat', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'hapus', 'data', 'hasilnya', 'nihil', 'lag', 'bikin', 'hp', 'cepat', 'panas', 'solusinya', 'chat', 'admin', 'tetep', 'ga', 'solusi', 'yg', 'work', 'mohon', 'oprek', 'apk', 'nya', 'enteng', 'hemat', 'data', 'kasih', 'bintang']

5. Stemming

Proses untuk mendapatkan bentuk dasar kata dengan cara menghilangkan semua imbuhan pada kata tersebut. Berikut merupakan hasil dari proses *Stemming*:

Tabel 7. Hasil Stemming

Sebelum	Sesudah
['shopee', 'ngga', 'perubahan', 'suruh', 'update', 'sampe', 'yg', 'terbaru', 'nggak', 'bedanya', 'tetep', 'berat', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'hapus', 'data', 'hasilnya', 'nihil', 'lag', 'bikin', 'hp', 'cepat', 'panas', 'solusinya', 'chat', 'admin', 'tetep', 'ga', 'solusi', 'yg', 'work', 'mohon', 'oprek', 'apk', 'nya', 'enteng', 'hemat', 'data', 'kasih', 'bintang']	['shopee', 'ngga', 'ubah', 'suruh', 'update', 'sampe', 'yg', 'baru', 'nggak', 'beda', 'tetep', 'berat', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'hapus', 'data', 'hasil', 'nihil', 'lag', 'bikin', 'hp', 'cepat', 'panas', 'solusi', 'chat', 'admin', 'tetep', 'ga', 'solusi', 'yg', 'work', 'mohon', 'oprek', 'apk', 'nya', 'enteng', 'hemat', 'data', 'kasih', 'bintang']

a. Pembobotan Kata

Proses *pre-processing* bertujuan untuk memastikan data siap digunakan untuk analisis lebih lanjut. Setelah dataset dibersihkan, langkah selanjutnya adalah memberi nilai pada setiap kata dalam teks tersebut untuk memudahkan klasifikasi. Nilai kata diberikan dengan menggunakan metode TF-IDF.

i. Term Frequency

Mengukur frekuensi kemunculan suatu istilah dalam suatu dokumen. Berikut adalah contoh perhitungan nilai TF berdasarkan hasil dari stemming, yaitu kumpulan kata-kata berikut: ['shopee', 'ngga', 'ubah', 'suruh', 'update', 'sampe', 'yg', 'baru', 'nggak', 'beda', 'tetep', 'berat', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'hapus', 'data', 'hasil', 'nihil', 'lag', 'bikin', 'hp', 'cepat', 'panas', 'solusi', 'chat', 'admin', 'tetep', 'ga', 'solusi', 'yg', 'work', 'mohon', 'oprek', 'apk', 'nya', 'enteng', 'hemat', 'data', 'kasih', 'bintang']

Berdasarkan perhitungan dengan menggunakan persamaan (1), diperoleh hasil sebagai berikut.

$$shopee = \frac{1}{45} = 0,0222222222222222$$

$$yg = \frac{2}{45} = 0,04444444444444445$$

$$data = \frac{3}{45} = 0,06666666666666667$$

Jika dilihat dari dokumen-1 maka kata yang muncul 1 kali adalah ['shopee', 'ngga', 'ubah', 'suruh', 'update', 'sampe', 'baru', 'nggak', 'beda', 'tetep', 'berat', 'wifi', 'paket', 'log', 'out', 'hapus', 'hasil', 'nihil', 'lag', 'bikin', 'hp', 'cepat', 'panas', 'chat', 'admin', 'tetep', 'ga', 'work', 'mohon', 'oprek', 'apk', 'nya', 'enteng', 'hemat', 'kasih', 'bintang']. Lalu kata yang muncul 2 kali adalah ['yg', 'tetep', 'stabil', 'solusi']. Dan kata yang muncul 3 kali adalah ['data'].

Karena setiap kata dalam dokumen tersebut muncul dengan jumlah yang berbeda – beda, maka setiap kata tersebut memiliki nilai bobot TF yang sama sesuai dengan jumlah kemunculan tersebut. Nilai bobot TF untuk setiap kata

adalah sebagai berikut: ['shopee'=0.02, 'ngga'=0.02, 'ubah'=0.02, 'suruh'=0.02, 'update'=0.02, 'sampe'=0.02, 'baru'=0.02, 'nggak'=0.02, 'beda'=0.02, 'tetep'=0.02, 'berat'=0.02, 'wifi'=0.02, 'paket'=0.02, 'log'=0.02, 'out'=0.02, 'hapus'=0.02, 'hasil'=0.02, 'nihil'=0.02, 'lag'=0.02, 'bikin'=0.02, 'hp'=0.02, 'cepat'=0.02, 'panas'=0.02, 'chat'=0.02, 'admin'=0.02, 'tetep'=0.02, 'ga'=0.02, 'work'=0.02, 'mohon'=0.02, 'oprek'=0.02, 'apk'=0.02, 'nya'=0.02, 'enteng'=0.02, 'hemat'=0.02, 'kasih'=0.02, 'bintang'=0.02] ['yg'=0.04, 'tetep'=0.04, 'stabil'=0.04, 'solusi'=0.04] ['data'=0.06]

ii. Inverse Document Frequency

Memberikan bobot yang lebih rendah terhadap istilah yang umum digunakan. Contoh penerapan IDF untuk dokumen-1 ['shopee', 'ngga', 'ubah', 'suruh', 'update', 'sampe', 'yg', 'baru', 'nggak', 'beda', 'tetep', 'berat', 'wifi', 'stabil', 'paket', 'data', 'stabil', 'log', 'out', 'hapus', 'data', 'hasil', 'nihil', 'lag', 'bikin', 'hp', 'cepat', 'panas', 'solusi', 'chat', 'admin', 'tetep', 'ga', 'solusi', 'yg', 'work', 'mohon', 'oprek', 'apk', 'nya', 'enteng', 'hemat', 'data', 'kasih', 'bintang']. Setelah kata – kata dalam dataset dikoding, didapatkan jumlah kemunculan masing-masing kata dalam dataset tersebut. Berikut adalah hasil pengkodean tersebut.

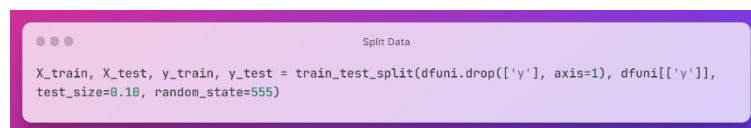
['shopee'=1218, 'ngga'=45, 'ubah'=90, 'suruh'=102, 'update'=303, 'sampe'=167, 'yg'=686, 'baru'=99, 'nggak'=55, 'beda'=75, 'tetep'=118, 'berat'=227, 'wifi'=73, 'stabil'=52, 'paket'=260, 'data'=81, 'stabil'=52, 'log'=78, 'out'=158, 'hapus'=158, 'data'=81, 'hasil'=76, 'nihil'=6, 'lag'=37, 'bikin'=189, 'hp'=264, 'cepat'=111, 'panas'=4, 'solusi'=69, 'chat'=106, 'admin'=51, 'tetep'=118, 'ga'=530, 'solusi'=69, 'yg'=686, 'work'=2, 'mohon'=132, 'oprek'=1, 'apk'=217, 'nya'=879, 'enteng'=3, 'hemat'=19, 'data'=81, 'kasih'=141, 'bintang'=151].

Hasil pembobotan kata dari dokumen tersebut adalah sebagai berikut. ['shopee'=0,001, 'ngga'=0,112, 'ubah'=0,049, 'suruh'=0,042, 'update'=0,010, 'sampe'=0,023, 'yg'=0,003, 'baru'=0,044, 'nggak'=0,088, 'beda'=0,061, 'tetep'=0,035, 'berat'=0,015, 'wifi'=0,063, 'stabil'=0,095, 'paket'=0,013, 'data'=0,056, 'stabil'=0,095, 'log'=0,115, 'out'=0,058, 'hapus'=0,024, 'data'=0,056, 'hasil'=0,060, 'nihil'=1,008, 'lag'=0,141, 'bikin'=0,019, 'hp'=0,012, 'cepat'=0,038, 'panas'=1,479, 'solusi'=0,067, 'chat'=0,040, 'admin'=0,097, 'tetep'=0,035, 'ga'=0,005, 'solusi'=0,067, 'yg'=0,003, 'work'=2,636, 'mohon'=0,030, 'oprek'=4,156, 'apk'=0,016, 'nya'=0,002, 'enteng'=1,905, 'hemat'=0,300, 'data'=0,056, 'kasih'=0,028, 'bintang'=0,026].

Selanjutnya, nilai bobot TF-IDF dapat dihitung dengan mengalikan nilai TF dengan IDF. Hasil perhitungannya dapat dilihat sebagai berikut. ['shopee'=0,042, 'ngga'=0,115, 'ubah'=0,099, 'suruh'=0,097, 'update'=0,073, 'sampe'=0,086, 'yg'=0,109, 'baru'=0,097, 'nggak'=0,110, 'beda'=0,103, 'tetep'=0,187, 'berat'=0,079, 'wifi'=0,104, 'stabil'=0,223, 'paket'=0,076, 'data'=0,306, 'stabil'=0,223, 'log'=0,115, 'out'=0,103, 'hapus'=0,087, 'data'=0,306, 'hasil'=0,103, 'nihil'=1,156, 'lag'=0,119, 'bikin'=0,083, 'hp'=0,076, 'cepat'=0,095, 'panas'=0,164, 'solusi'=0,211, 'chat'=0,096, 'admin'=0,112, 'tetep'=0,187, 'ga'=0,060, 'solusi'=0,211, 'yg'=0,109, 'work'=0,175, 'mohon'=0,091, 'oprek'=0,184, 'apk'=0,080, 'nya'=0,049, 'enteng'=0,169, 'hemat'=0,133, 'data'=0,306, 'kasih'=0,090, 'bintang'=0,088].

3.1.4 Modeling

Sebelum membangun model, data terlebih dahulu dibagi menjadi dua bagian, yaitu *data training* dan *data testing*. *Data training* digunakan untuk melatih model, sedangkan *data testing* digunakan untuk mengevaluasi kinerja model. Pembagian data dilakukan dengan menggunakan bahasa pemrograman Python, dengan kode sebagai berikut:



```
X_train, X_test, y_train, y_test = train_test_split(dfuni.drop(['y'], axis=1), dfuni[['y']],
test_size=0.10, random_state=555)
```

Gambar 5. Sintaks Split Data

Setelah data dibagi menjadi *data training* dan *data testing*, data tersebut digunakan untuk melatih dan menguji model algoritma Support Vector Machine. Proses ini dapat dilihat pada gambar berikut.

```

Support Vector Machine

clf_svm = svm.SVC(max_iter=-1,
                  C=1000,
                  kernel='linear')
clf_svm.fit(X_unitrain, y_unitrain.values.ravel())
    
```

Gambar 6. Sintaks Modeling Support Vector Machine

Setelah pemodelan dengan algoritma Support Vector Machine, langkah selanjutnya adalah menerapkan algoritma Random Forest pada data yang telah dibagi menjadi data training dan data testing, seperti yang ditunjukkan pada Gambar 7.

```

Random Forest

# Membuat objek model Random Forest
clf_rf = RandomForestClassifier(n_estimators=100, random_state=555)
# Melatih model dengan data pelatihan
clf_rf.fit(X_unitrain, y_unitrain.values.ravel())
    
```

Gambar 7. Sintaks Modeling Random Forest

Setelah pemodelan dengan algoritma Support Vector Machine dan Random Forest, langkah selanjutnya adalah menerapkan algoritma Logistic Regression pada data yang telah dibagi menjadi data training dan data testing, seperti yang ditunjukkan pada Gambar 8.

```

Logistic Regression

# Membuat objek model Regresi Logistik
clf_lr = LogisticRegression(solver='lbfgs', max_iter=1000, random_state=555)
# Melatih model dengan data pelatihan
clf_lr.fit(X_unitrain, y_unitrain.values.ravel())
    
```

Gambar 8. Sintaks Modeling Logistic Regression

3.1.5 Evaluation

Setelah menjalani tahap pemodelan menggunakan tiga algoritma berbeda, peneliti melanjutkan dengan tahap evaluasi untuk menilai kinerja masing - masing model dalam mengklasifikasikan sentimen ulasan pengguna Shopee. Evaluasi ini menggunakan confusion matrix dan selanjutnya dilakukan pengujian evaluasi untuk mendapatkan nilai hasil *accuracy*, *f1-score*, *recall*, serta *precision*. Gambar 9 berikut menampilkan kode untuk mendapatkan confusion matrix dan pengujian evaluasi untuk Support Vector Machine.

```

Support Vector Machine

# Hitung dan tampilkan confusion matrix
cm_svm = confusion_matrix(y_unittest, y_unipred_svm)
print("Confusion Matrix:")
print(cm_svm)
# Visualisasi confusion matrix menggunakan heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(cm_svm, annot=True, fmt="d", cmap="Blues", cbar=False,
            xticklabels=sentimen_mapping.keys(), yticklabels=sentimen_mapping.keys())
plt.title("Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.show()
    
```

Gambar 9. Sintaks Evaluation Support Vector Machine

Langkah berikutnya, setelah evaluasi algoritma Support Vector Machine, adalah mengevaluasi algoritma Random Forest untuk mendapatkan mendapatkan confusion matrix dan pengujian evaluasi. Berikut kode untuk evaluasi Random Forest.

```

Random Forest
# Hitung dan tampilkan confusion matrix
cm_rf = confusion_matrix(y_unittest, y_unipred_rf)
print("Confusion Matrix:")
print(cm_rf)
# Visualisasi confusion matrix menggunakan heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(cm_rf, annot=True, fmt="d", cmap="Blues", cbar=False,
            xtickLabels=sentimen_mapping.keys(), ytickLabels=sentimen_mapping.keys())
plt.title("Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.show()
    
```

Gambar 10. Sintaks Evaluation Random Forest

Selanjutnya, setelah algoritma Support Vector Machine dan Random Forest, langkah terakhir adalah menerapkan kode python untuk mengevaluasi algoritma Random Forest pada klasifikasi ulasan pengguna Shopee, sebagaimana terlihat pada Gambar 11.

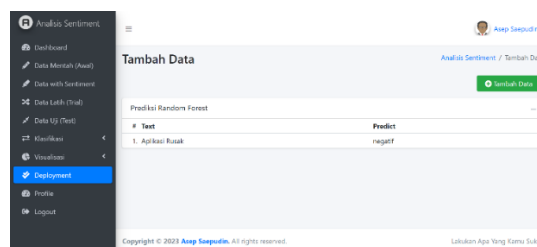
```

Logistic Regression
# Hitung dan tampilkan confusion matrix
cm_lr = confusion_matrix(y_unittest, y_unipred_lr)
print("Confusion Matrix:")
print(cm_lr)
# Visualisasi confusion matrix menggunakan heatmap
plt.figure(figsize=(8, 6))
sns.heatmap(cm_lr, annot=True, fmt="d", cmap="Blues", cbar=False,
            xtickLabels=sentimen_mapping.keys(), ytickLabels=sentimen_mapping.keys())
plt.title("Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.show()
    
```

Gambar 11. Sintaks Evaluation Logistic Regression

3.1.6 Deployment

Sebuah sistem hasil pengembangan algoritma dengan akurasi tertinggi telah diimplementasikan menggunakan Python dan PHP. Sistem ini dapat mengklasifikasikan kalimat opini baru menjadi positif, negatif, atau netral. Sistem ini bekerja dengan baik, meskipun dibatasi oleh jumlah karakter maksimal 150.



Gambar 12. Antarmuka Hasil Deployment

3.2 Pembahasan

3.2.1 Penerapan Algoritma Support Vector Machine, Random Forest, dan Logistic Regression dalam Klasifikasi Ulasan Pengguna Shopee

Dalam penerapan algoritma pada penelitian ini, tiga model klasifikasi utama digunakan yaitu Support Vector Machine, Random Forest, dan Logistic Regression. Pemilihan algoritma didasarkan pada pertimbangan kelebihan masing – masing. Algoritma *Support Vector Machine* (SVM) digunakan karena menggunakan *kernel linear* untuk membuat batas pemisah linier antara dua kelas data[12]. Random Forest diterapkan karena memiliki akurasi yang bagus dan sifatnya yang sederhana dan mudah diterapkan[13]. Dan Logistic Regression ketika menggunakan rasio data 90:10 didapatkan hasil yang terbaik [5].

Berdasarkan penerapan ketiga algoritma tersebut dalam klasifikasi ulasan pengguna shopee didapatkan hasil klasifikasi sebagai berikut.

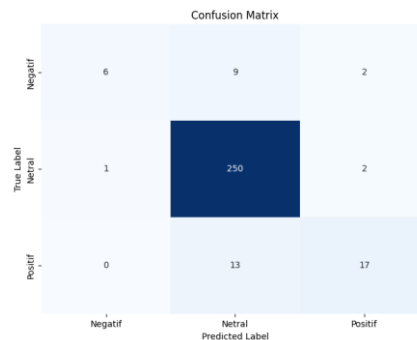
Tabel 8. Hasil Klasifikasi Ulasan Pengguna Shopee

Algoritma	Prediksi Sentime		
	Positif	Netral	Negatif
Support Vector Machine	21	272	7
Random Forest	20	267	13
Logistic Regression	5	293	2

3.2.2 Evaluasi Hasil Klasifikasi Ulasan Pengguna Shopee menggunakan Support Vector Machine, Random Forest, dan Logistic Regression

a. Support Vector Machine

Berdasarkan Gambar 9 pada tahap evaluasi, maka didapatkan confusion matrix seperti berikut:



Gambar 13. Confusion Matrix Support Vector Machine

Dari Gambar 12 tersebut berhasil mengklasifikasikan secara benar pada data positif (True Positive) sebesar 17, lalu data benar netral (True Neutral) sebanyak 250 dan data benar negatif (True Negative) sebanyak 6. Adapun hasil dari perhitungan nilai accuracy, precision, recall dan F1-Score Support Vector Machine dapat dilihat pada Tabel 9 berikut:

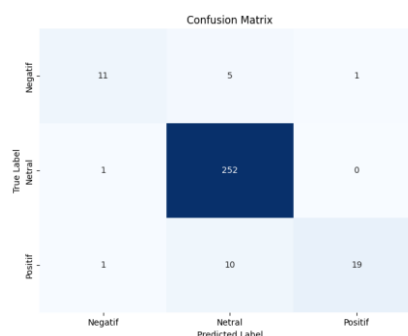
Tabel 9. Hasil Pengujian Support Vector Machine

	Precision	Recall	F1-Score	Support
Negatif	0,857	0,353	0,500	17
Netral	0,919	0,988	0,952	253
Positif	0,810	0,567	0,667	30
Accuracy			0,910	300

Dalam tabel diatas hasil precision terhadap kelas positif 81%, kelas negatif 85% dan kelas netral 91%. Dari hasil tersebut menunjukkan bahwa label netral memiliki akurasi klasifikasi yang lebih tinggi daripada label positif dan negatif. Hal ini menunjukkan bahwa sistem lebih baik dalam menemukan kembali informasi yang berlabel netral[3]. Penelitian ini menggunakan algoritma Support Vector Machine untuk mengklasifikasikan analisis sentimen Shopee dengan akurasi 91%.

b. Random Forest

Selanjutnya berdasarkan Gambar 10 pada tahap evaluasi model Random Forest, maka didapatkan confusion matrix seperti berikut:



Gambar 14. Confusion Matrix Random Forest

Dari Gambar 13 diatas berhasil mengklasifikasikan secara benar pada data positif (True Positive) sebesar 19, data benar netral (True Neutral) sebanyak 252 dan data benar negatif (True Negative) sebanyak 11. Sedangkan pada kelas yang seharusnya positif (False positive negative) sebanyak 1 dan kelas seharusnya positif yang lain (False positive neutral) sebanyak 10. Setelah pengujian menggunakan confusion matrix, langkah selanjutnya adalah

menghitung nilai akurasi, presisi, recall, dan F1-score[14]. Hasil perhitungan Random Forest dapat dilihat pada Tabel 10 berikut:

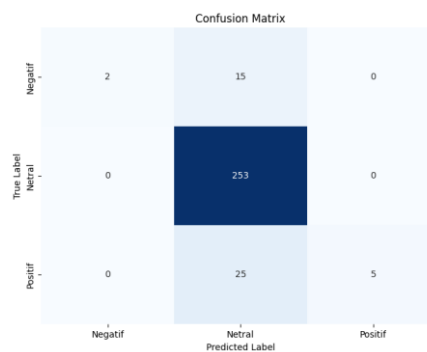
Tabel 10. Hasil Pengujian Random Forest

	Precision	Recall	F1-Score	Support
Negatif	0,846	0,647	0,733	17
Netral	0,944	0,996	0,969	253
Positif	0,950	0,633	0,760	30
Accuracy			0,940	300

Dalam tabel diatas hasil precision terhadap kelas positif 84%, kelas negatif 95% dan kelas netral 94%. Hasil recall terhadap kelas positif 63%, kelas negatif 64% dan kelas netral 96%. Hasil F1-score kelas positif 76%, kelas negatif 73% dan kelas netral 96%. Lalu akurasi didapat menggunakan algoritma Random Forest sebesar 94%.

c. Logistic Regression

Kemudian berdasarkan Gambar 11 pada tahap evaluasi, maka didapatkan hasil confusion matrix sebagai berikut:



Gambar 15. Confusion Matrix Logistic Regression

Dari Gambar 14 diatas berhasil mengklasifikasikan secara benar pada data positif (True Positive) sebesar 19, data benar netral (True Neutral) sebanyak 252 dan data benar negatif (True Negative) sebanyak 11. Sedangkan pada kelas yang seharusnya positif (False positive negative) sebanyak 1 dan kelas seharusnya positif yang lain (False positive neutral) sebanyak 10. Lalu pada kelas yang seharusnya negatif (False negative positive) sebanyak 11, kelas seharusnya negatif (False negative neutral) sebanyak 11, pada kelas yang seharusnya netral (False neutral positive) sebanyak dan kelas yang seharusnya netral (false neutral negative). Setelah pengujian menggunakan confusion matrix, langkah selanjutnya adalah menghitung nilai akurasi, presisi, recall, dan F1-score[8]. Nilai akurasi, presisi, recall, dan F1-score dari perhitungan Logistic Regression dapat dilihat pada Tabel 17 berikut:

Tabel 11. Hasil Pengujian Logistic Regression

	Precision	Recall	F1-Score	Support
Negatif	0,846	0,647	0,733	17
Netral	0,944	0,996	0,969	253
Positif	0,950	0,633	0,760	30
Accuracy			0,940	300

Dalam tabel diatas hasil precision terhadap kelas positif 100%, kelas negatif 100% dan kelas netral 86%. Hasil recall terhadap kelas positif 16%, kelas negatif 11% dan kelas netral 100%. Hasil F1-score kelas positif 28%, kelas negatif 21% dan kelas netral 92%. Lalu akurasi didapat menggunakan algoritma Logistic Regression sebesar 86%.

3.2.3 Hasil Perbandingan Klasifikasi Ulasan Pengguna Shopee menggunakan Support Vector Machine, Random Forest, dan Logistic Regression

Perbandingan nilai accuracy dari algoritma klasifikasi Support Vector Machine, Random Forest dan Logistic Regression dapat dilihat pada Tabel 14.

Tabel 12. Perbandingan Accuracy Algoritma Klasifikasi

Algoritma	Accuracy
Support Vector Machine	91%

Random Forest	94%
Logistic Regression	86%

Berdasarkan Tabel 14 tersebut dapat diketahui bahwa nilai akurasi 91% untuk algoritma Support Vector Machine, lalu untuk algoritma Random Forest memiliki nilai akurasi 94% dan untuk algoritma Logistic Regression memiliki nilai akurasi 86%. Akurasi adalah ukuran seberapa baik model pengujian dapat mengklasifikasikan data dengan benar. Akurasi dihitung dengan membagi jumlah klasifikasi kelas yang benar dengan jumlah total data yang diuji[3]. Berdasarkan hasil pengujian, tingkat kemampuan sistem dalam mengklasifikasikan ulasan pengguna Shopee adalah 91% untuk algoritma Support Vector Machine, 94% untuk algoritma Random Forest, dan 86% untuk algoritma Logistic Regression. Dari hasil tersebut dipastikan bahwa Random Forest memiliki akurasi tertinggi diikuti oleh Support Vector Machine dan Logistic Regression.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan dalam mengklasifikasikan ulasan pengguna Shopee menggunakan algoritma Support Vector Machine, Random Forest dan Logistic Regression dapat disimpulkan bahwa :

1. Algoritma klasifikasi Support Vector Machine, Random Forest dan Logistic Regression dapat diterapkan dalam konteks ini ulasan pengguna Shopee.
2. Hasil evaluasi algoritma Support Vector Machine mendapatkan akurasi 91%, lalu Random Forest mendapatkan 94% akurasi dan akurasi Logistic Regression sebesar 86%.
3. Dari hasil perbandingan algoritma dipastikan bahwa Random Forest memiliki akurasi tertinggi diikuti oleh Support Vector Machine dan Logistic Regression.

UCAPAN TERIMAKASIH

Penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada orangtua, dosen pembimbing, dan seluruh pihak yang telah membantu dan mendukung penulis dalam menyelesaikan penelitian dan publikasi jurnal ini.

REFERENCES

- [1] Detika Yossy Pramesti, Sri Widyastuti, and Dian Riskarini, "Pengaruh Kualitas Pelayanan, Keragaman Produk, Dan Promosi E-Commerce Terhadap Kepuasan Konsumen Shopee," *JIMP J. Ilm. Manaj. Pancasila*, vol. 1, no. 1, pp. 27–39, 2021, doi: 10.35814/jimp.v1i1.2065.
- [2] A. S. Widagdo, B. S. W. A, and A. Nasiri, "Analisis Tingkat Kepopuleran E-Commerce di Indonesia Berdasarkan Sentimen Sosial Media Menggunakan Metode Naive Bayes," *J. Inf.*, vol. 6, pp. 1–5, 2020.
- [3] T. P. R. Sanjaya, A. Fauzi, and A. F. N. Masruriyah, "Analisis sentimen ulasan pada e-commerce shopee menggunakan algoritma naive bayes dan support vector machine," *INFOTECH J. Inform. Teknol.*, vol. 4, no. 1, pp. 16–26, 2023, doi: 10.37373/infotech.v4i1.422.
- [4] A. C. . Nurhaliza, D. H. Citra, W. Purnama, and R. A. Kurnia, "The Implementation of Naïve Bayes Algorithm for Sentiment Analysis of Shopee Reviews on Google Play Store," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 2, no. April, pp. 47–54, 2022.
- [5] Y. Pratama, D. T. Murdiansyah, and K. M. Lhaksana, "Analisis Sentimen Kendaraan Listrik Pada Media Sosial Twitter Menggunakan Algoritma Logistic Regression dan Principal Component Analysis," *J. Media Inform. Budidarma*, vol. 7, no. 1, pp. 529–535, 2023, doi: 10.30865/mib.v7i1.5575.
- [6] T. A.M and A. Yaqin, "Perbandingan Algoritma Naïve Bayes, K-Nearest Neighbors dan Random Forest untuk Klasifikasi Sentimen Terhadap BPJS Kesehatan pada Media Twitter," *InComTech J. Telekomun. dan Komput.*, vol. 12, no. 1, p. 01, 2022, doi: 10.22441/incomtech.v12i1.13642.
- [7] N. Herlinawati, Y. Yuliani, S. Faizah, W. Gata, and S. Samudi, "Analisis Sentimen Zoom Cloud Meetings di Play Store Menggunakan Naive Bayes dan Support Vector Machine," *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 5, no. 2, p. 293, 2020, doi: 10.24114/cess.v5i2.18186.
- [8] I. R. Afandi, I. F. Hanif, F. N. Hasan, E. Sinduningrum, Z. Halim, and N. Pratiwi, "Analisis Sentimen Opini Masyarakat Terkait Penyelenggaraan Sistem Elektronik Menggunakan Metode Logistic Regression," *J. Linguist. Komputasional*, vol. 5, no. 2, pp. 77–84, 2022, [Online]. Available: <https://inacl.id/journal/index.php/jlk/article/view/103>
- [9] A. Baita, Y. Pristyanto, and N. Cahyono, "Analisis Sentimen Mengenai Vaksin Sinovac Menggunakan Algoritma Support Vector Machine (SVM) dan K-Nearest Neighbor (KNN)," *Inf. Syst. J.*, vol. 4, no. 2, pp. 42–46, 2021, [Online]. Available: <https://jurnal.amikom.ac.id/index.php/infos/article/view/687>
- [10] D. Kurniawan and M. Yasir, "Optimization Sentimen Analysis using CRISP-DM and Naive Bayes Methods Implemented on Social Media," *Cybersp. J. Pendidik. Teknol. Inf.*, vol. 6, no. 2, p. 74, 2022, doi: 10.22373/cj.v6i2.12793.
- [11] S. Navisa, Luqman Hakim, and Aulia Nabilah, "Komparasi Algoritma Klasifikasi Genre Musik pada Spotify Menggunakan CRISP-DM," *J. Sist. Cerdas*, vol. 4, no. 2, pp. 114–125, 2021, doi: 10.37396/jsc.v4i2.162.
- [12] S. F. Pane, A. Owen, and C. Prianto, "Analisis Sentimen UU Omnibus Law pada Twitter Menggunakan Metode

- Support Vector Machine,” *J. Telekomun. dan Komput.*, vol. 11, no. 2, p. 130, 2021, doi: 10.22441/incomtech.v11i2.10874.
- [13] E. Fitri, “Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naive Bayes, Random Forest Dan Support Vector Machine,” *J. Transform.*, vol. 18, no. 1, p. 71, 2020, doi: 10.26623/transformatika.v18i1.2317.
- [14] A. B. Wibisono and A. Fahrurozi, “Perbandingan Algoritma Klasifikasi Dalam Pengklasifikasian Data Penyakit Jantung Koroner,” *J. Ilm. Teknol. dan Rekayasa*, vol. 24, no. 3, pp. 161–170, 2019, doi: 10.35760/tr.2019.v24i3.2393.