

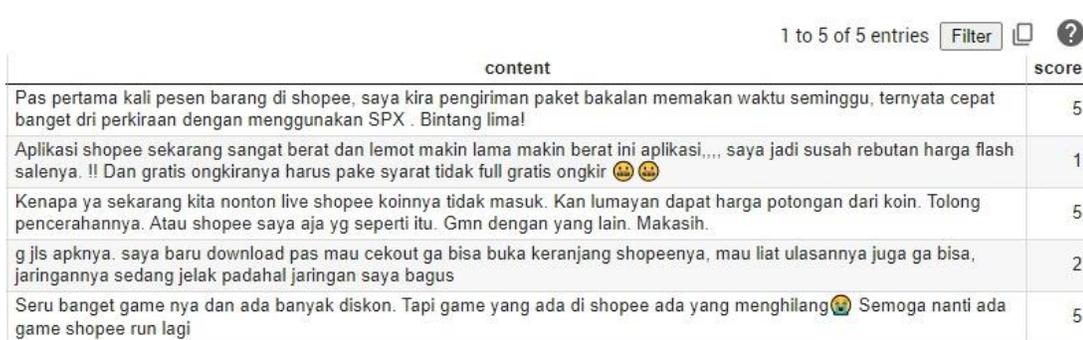
BAB IV

HASIL DAN PEMBAHASAN

4.1 Pengumpulan Data

Dataset penelitian ini terdiri dari ulasan pengguna aplikasi Shopee yang diambil dari *Google Play Store* melalui metode *scraping* dengan bantuan *google-play-scraper*. *Google Play Scraper* menggunakan API-nya untuk mengumpulkan data dan menyimpannya dalam format CSV. Sebanyak 8229 data dikumpulkan dan digunakan dalam penelitian ini. Data yang digunakan adalah data *rating* Februari 2024 – Agustus 2024.

Untuk melakukan *scraping* data, digunakan variabel terbaru dari *Google Play Scraper*, mencakup waktu pengambilan data dan seluruh *rating* untuk mendapatkan data lengkap. Dalam penelitian ini, aplikasi yang digunakan adalah Shopee, dengan ID aplikasi dari alamat: <https://play.google.com/store/apps/details?id=com.shopee.id>. Data yang telah di-*scraping* kemudian dikonversi ke dalam format CSV.



The image shows a screenshot of a web scraping result. At the top right, it says "1 to 5 of 5 entries" with a "Filter" button and a help icon. Below this is a table with two columns: "content" and "score". The table contains five rows of data, each representing a review and its corresponding score.

content	score
Pas pertama kali pesen barang di shopee, saya kira pengiriman paket bakalan memakan waktu seminggu, ternyata cepat banget dri perkiraan dengan menggunakan SPX . Bintang lima!	5
Aplikasi shopee sekarang sangat berat dan lemot makin lama makin berat ini aplikasi,,, saya jadi susah rebutan harga flash salenya. !! Dan gratis ongkirnya harus pake syarat tidak full gratis ongkir 😞😞	1
Kenapa ya sekarang kita nonton live shopee koinnya tidak masuk. Kan lumayan dapat harga potongan dari koin. Tolong pencerahannya. Atau shopee saya aja yg seperti itu. Gmn dengan yang lain. Makasih.	5
g jls apknya. saya baru download pas mau cekout ga bisa buka keranjang shopeenya, mau liat ulasannya juga ga bisa, jaringannya sedang jelak padahal jaringan saya bagus	2
Seru banget game nya dan ada banyak diskon. Tapi game yang ada di shopee ada yang menghilang🥲 Semoga nanti ada game shopee run lagi	5

Gambar 4. 1 Hasil *Web Scraping* Ulasan Shopee

Pada gambar 4.1 berisikan data *rating* pengguna Shopee dimana data tersebut berisikan tanggal, isi komentar dalam Bahasa Indonesia, dan penilaian dalam bentuk bintang 1 sampai 5.

Setelah data *rating* pengguna Shopee sudah didapat, tahap selanjutnya membandingkan data *rating* pengguna aplikasi Shopee yg sudah diperoleh dengan *rating* aplikasi *e-commerce* lainnya. Peneliti akan membandingkan dengan *rating* pengguna aplikasi Tokopedia, Lazada, dan Bukalapak. Untuk pengumpulan datanya, sama seperti mengumpulkan data *rating* pengguna Shopee.

Tabel 4. 1 Jumlah *Rating* Setiap *E-Commerce*

	Shopee	Tokopedia	Lazada	Bukalapak
Jumlah <i>Rating</i>	8229	1369	1148	995

Pada tabel 4.1 dapat dilihat bahwa Shopee mempunyai data *rating* sebesar 8229. Tokopedia mempunyai data *rating* sebesar 1369. Lazada mempunyai data *rating* sebesar 1148. Bukalapak mempunyai data *rating* sebesar 995.

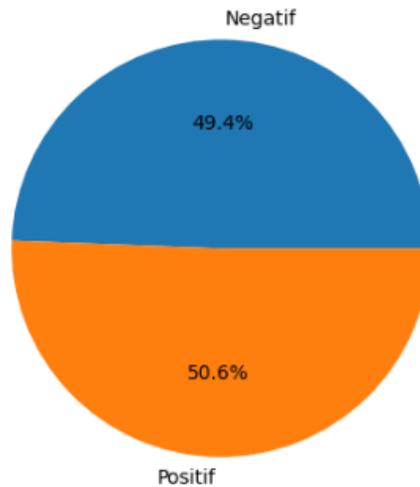
4.2 Pelabelan Data

Setelah data dikumpulkan, langkah berikutnya adalah pelabelan data. Ini mengategorikan data menjadi ulasan positif dan negatif. Ulasan positif menunjukkan kepuasan pengguna atau dukungan terhadap aplikasi Shopee, sedangkan ulasan negatif menunjukkan ketidakpuasan atau kritik terhadap layanan yang ditawarkan. Contoh hasil pelabelan dapat dilihat pada gambar 4.2.

Seneng banget punya Shopee karna bisa belanja sepuasnya juga harganya murah-murah juga datang selalu tepat waktu makasih Shopee	5	Positif
Suka banget sma shopee. Tp minusnya masih bnyk akun penjual bodong. Beli handpon yg datang sendal. Tp untung masih bisa di kembalikan, dengan rugi ongkir. Yaaa. Semoga kedepan bisa lebih selektif dan tidak merugikan pembeli. Tp sejauh ini aku suka sekali belanja di shopee.	5	Positif
Membantu bagi kami yang jauh dari kota untuk belanja tpi sayang pengiriman dari jasa SPX express tidak memuaskan barang selalu telat melebihi dari estimasih yang di tentukan.	5	Positif
beli apa aja ada, harga juga sesuai kantong, barang dtng tepat waktu, bnyk dpt diskon, yg butuh pinjaman dana juga ada, lengkap, bnyk bgt keuntungan dr aplikasi ini, sangat membantu sekali, smoga makin sukses 🙏, terimakasih Shopee 😊😊	5	Positif
Mohon di update lagi soal pengiriman paket jangan terlalu lama- lama di pusat penyortiran, yang membuat saya jadi tanda tanya, soalnya langsung di antar ke alamat saya dan itupun saya kira masih di sortir ternyata sudah di antar oleh kurir ke alamat saya yang membuat saya kaget laa kok bisa🙏 tolong di update-update-update-update lagi soal pengiriman barang atau paket atau produk atau apalah yang penting update upupupupupup,	5	Positif
Bagus banget belanja banyak dapt spaylater juga. Spinjam dengan cicilan yang sangat ringan,, bisa membantu disaat ada kebutuhan. Terima kasih shoppe	5	Positif
Saya seller dishopee. Sekarang pakai platform shopee sangat tidak fleksible untuk jualan. Sama seperti teman penjual yang lain. Saya gabisa belanja produk saya gabisa dibeli saya gabisa tarik uang. Terus gimana kalo seller yg ngandelin penghasilan dari shopee terus jadi dibekukan 4 minggu??? Lebih better aplikasi ecommerce sebelah untuk jualan. Masih fleksible dan ga rewel. Saya minta banding harus 7 hari kerja. Sangat menyulitkan pekerjaan begini.	1	Negatif

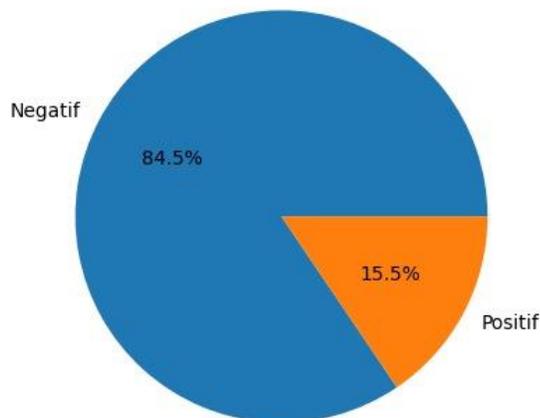
Gambar 4. 2 Contoh Hasil Pelabelan

Berikut ini adalah perbandingan hasil pelabelan data ulasan pengguna pada aplikasi Shopee, Tokopedia, Lazada dan Bukalapak.



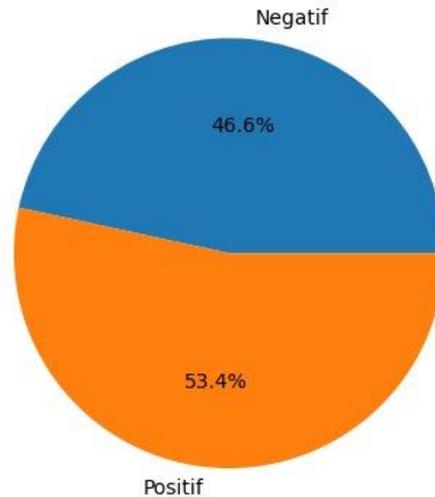
Gambar 4. 3 *Rating* Pengguna pada Aplikasi Shopee

Pada gambar 4.3 berisikan data *rating* pengguna pada aplikasi Shopee yang terdiri dari 49,4 persen ulasan negatif dan 50,6 persen ulasan positif. Dapat dilihat bahwa jumlah ulasan positif lebih besar daripada ulasan negatif.



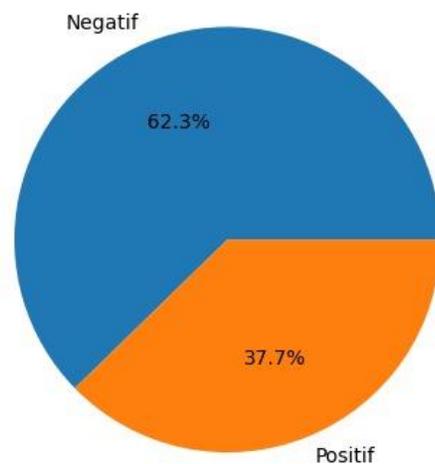
Gambar 4. 4 *Rating* pengguna pada aplikasi Tokopedia

Pada gambar 4.4 berisikan data *rating* pengguna pada aplikasi Tokopedia yang terdiri dari 84,5 persen ulasan negatif dan 15,5 persen ulasan positif. Dapat dilihat bahwa jumlah ulasan negatif lebih besar daripada ulasan positif.



Gambar 4. 5 *Rating* pengguna pada aplikasi Lazada

Pada gambar 4.5 berisikan data *rating* pengguna pada aplikasi Lazada yang terdiri dari 46,6 persen ulasan negatif dan 53,4 persen ulasan positif. Dapat dilihat bahwa jumlah ulasan positif lebih besar daripada ulasan negatif.



Gambar 4. 6 *Rating* pengguna pada aplikasi Bukalapak

Pada gambar 4.6 berisikan data *rating* pengguna pada aplikasi Bukalapak yang terdiri dari 62,3 persen ulasan negatif dan 37,7 persen ulasan positif. Dapat dilihat bahwa jumlah ulasan negatif lebih besar daripada ulasan positif.

4.3 Preprocessing Data

Setelah pengumpulan data ulasan serta pelabelan data, dilakukan tahapan preprocessing data dengan menggunakan beberapa metode untuk menghilangkan *noise* dan perkataan yang tidak memiliki arti. Pada tahap ini memanfaatkan *library* pada *python*. Metode yang digunakan antara lain adalah *case folding*, *cleaning data*, *stopword removal*, *tokenizing*, dan *stemming*. Data *rating* dapat dibersihkan dan disiapkan untuk analisis selanjutnya melalui tahap ini.

4.3.1 Case Folding

Case Folding adalah proses mengubah huruf dalam teks menjadi huruf kecil untuk menghindari perbedaan makna. Proses ini mencakup pengubahan semua huruf dari "A" hingga "Z" menjadi "a" hingga "z". Berikut ini contoh penerapan *case folding*.

Tabel 4. 2 Contoh Penerapan *Case Folding*

<i>Input</i>	<i>Output</i>
Aplikasi sering lag/error. Beberapa menu tidak bisa diakses, sudah dipencet tapi stuck atau tidak langsung menuju menu yang dituju padahal jaringan bagus.	aplikasi sering lag/error beberapa menu tidak bisa diakses sudah dipencet tapi stuck atau tidak langsung menuju menu yang dituju padahal jaringan bagus.
Selama belanja lewat shopee saya sangat terbantu sekali, jadi lebih gampang untuk mencari barang-barang yang dibutuhkan dan yang sesuai dengan budget, menghemat waktu juga	selama belanja lewat shopee saya sangat terbantu sekali, jadi lebih gampang untuk mencari barang-barang yang dibutuhkan dan yang sesuai dengan budget, menghemat waktu juga

4.3.2 *Cleaning Data*

Setelah tahap *case folding* selesai, proses pembersihan data dilakukan untuk menghilangkan elemen analisis seperti tanda baca, simbol, angka, dan karakter huruf lainnya dari *dataset*. Tujuan dari pembersihan data ini adalah agar *dataset* menjadi bersih. Berikut ini contoh penerapan *cleaning data*.

Tabel 4. 3 Contoh Penerapan *Cleaning Data*

<i>Input</i>	<i>Output</i>
aplikasi sering lag/error. beberapa menu tidak bisa diakses, sudah dipencet tapi stuck atau tidak langsung menuju menu yang dituju padahal jaringan bagus.	aplikasi sering lag error beberapa menu tidak bisa diakses sudah dipencet tapi stuck atau tidak langsung menuju menu yang dituju padahal jaringan bagus
selama belanja lewat shopee saya sangat terbantu sekali, jadi lebih gampang untuk mencari barang-barang yang dibutuhkan dan yang sesuai dengan budget, menghemat waktu juga	selama belanja lewat shopee saya sangat terbantu sekali jadi lebih gampang untuk mencari barang barang yang dibutuhkan dan yang sesuai dengan budget menghemat waktu juga

4.3.3 *Stopword Removal*

Stopword removal adalah prosedur untuk menghapus kata-kata dalam dokumen yang tidak memiliki arti atau sambungannya. Kata-kata ini biasanya berupa kata sambung, kata ganti orang, atau kata-kata yang tidak memiliki arti saat melakukan tahapan *stopword*. Berikut ini contoh penerapan *stopword removal*.

Tabel 4. 4 Contoh Penerapan *Stopword Removal*

<i>Input</i>	<i>Output</i>
aplikasi sering lag error beberapa menu tidak bisa diakses sudah dipencet tapi	aplikasi lag error menu diakses dipencet stuck langsung menu dituju jaringan bagus

stuck atau tidak langsung menuju menu yang dituju padahal jaringan bagus	
selama belanja lewat shopee saya sangat terbantu sekali jadi lebih gampang untuk mencari barang-barang yang dibutuhkan dan yang sesuai dengan budget menghemat waktu juga	belanja shopee terbantu gampang mencari barang dibutuhkan sesuai budget menghemat

4.3.4 Tokenizing

Setelah melakukan tahap *stopword*, tahap selanjutnya melakukan *tokenizing*. Proses ini mengacu pada proses mengubah rangkaian teks menjadi bagian-bagian yang lebih kecil dikenal sebagai token. Berikut ini contoh penerapan *tokenizing*.

Tabel 4. 5 Contoh Penerapan *Tokenizing*

<i>Input</i>	<i>Output</i>
aplikasi lag error menu diakses dipencet stuck langsung menu dituju jaringan bagus	[aplikasi, lag, error, menu, diakses, dipencet, stuck ,langsung, menu, dituju, jaringan, bagus]
belanja shopee terbantu gampang mencari barang dibutuhkan sesuai budget menghemat	[belanja, shopee, terbantu, gampang, mencari, barang, dibutuhkan, sesuai, budget, menghemat]

4.3.5 Stemming

Pada tahap ini, kata dasar diperoleh dengan menghilangkan imbuhan dari setiap kata. Berikut ini contoh penerapan *stemming*.

Tabel 4. 6 Contoh Penerapan *Stemming*

<i>Input</i>	<i>Output</i>
--------------	---------------

[aplikasi, lag, error, menu, diakses, dipencet, stuck ,langsung, menu, dituju, jaringan, bagus]	[aplikasi lag error menu akses pencet stuck langsung menu tuju jaring bagus]
[belanja, shopee, terbantu, gampang, mencari, barang, dibutuhkan, sesuai, budget, menghemat]	[belanja shopee bantu gampang cari barang butuh sesuai budget hemat]

4.4 Klasifikasi

Dalam penelitian ini, data *review* pengguna akan dibagi menjadi data *training* dan data *testing* untuk melakukan pengujian akurasi dengan rasio 80:20. Pembagian data memanfaatkan *library* pada *python* yaitu `from sklearn.model_selection import train_test_split`.

Setelah data melewati tahapan pembagian data, dilanjutkan dengan proses pembobotan kata. Tujuan dari proses ini adalah untuk memberi bobot pada setiap kata sehingga bisa diproses oleh model *machine learning* seperti *Decision Tree* yang hanya menerima data numerik. Tahapan ini melibatkan pembobotan menggunakan metode *CountVectorizer* memanfaatkan *library* pada *python* yaitu `from sklearn.feature_extraction.text import CountVectorizer`.

4.4.1 Decision Tree

Setelah melakukan pembagian data dan pembobotan kata, dilanjutkan dengan pemodelan data menggunakan *Decision Tree* dengan memanfaatkan *library* pada *python* yaitu `from sklearn.tree import DecisionTreeClassifier`. Berikut adalah tabel implementasi *Decision Tree* pada Shopee, Tokopedia, Lazada dan Bukalapak.

Tabel 4. 7 Implementasi *Decision Tree*

	Shopee	Tokopedia	Lazada	Bukalapak
Data Training	1226	1014	822	732
Data Testing	307	254	206	182
Hasil Akurasi	73,74%	83,07%	68,44%	74,86%

4.4.2 *Decision Tree* berbasis SMOTE

Selanjutnya, peneliti akan menggunakan SMOTE dalam metode *decision tree* untuk menyeimbangkan distribusi kelas. SMOTE membantu mengatasi masalah ketidakseimbangan data dengan membuat contoh sintesis dari kelas minoritas, yang memungkinkan model *Decision Tree* untuk belajar dari jumlah data yang lebih seimbang. Penggunaan metode *Decision Tree* berbasis SMOTE memanfaatkan *library* dari *python* yaitu *from imblearn.over_sampling import SMOTE*. Berikut adalah tabel implementasi *Decision Tree* berbasis SMOTE pada Shopee, Tokopedia, Lazada dan Bukalapak.

Tabel 4. 8 Implementasi *Decision Tree* berbasis SMOTE

	Shopee	Tokopedia	Lazada	Bukalapak
Data Training	1226	1014	822	732
Data Testing	307	254	206	182
Hasil Akurasi	73,13%	78,34%	71,35%	70,49%

4.5 Evaluasi

Tujuan dari evaluasi ini adalah untuk menentukan skenario mana yang akan berhasil menggunakan model *maching learning* yang telah dibuat. Tabel akan digunakan untuk menampilkan hasil pengklasifikasian dan disertakan dengan perhitungan matematika menggunakan rumus yang telah ditetapkan. Selain itu, penggunaan tabel *confusion matrix* akan membuat proses perhitungan yang diperlukan untuk pengambilan kesimpulan penelitian ini lebih mudah. Nilai *True Negative* (TN), *True Positive* (TP), *False Negative* (FN), dan *False Positive* (FP) dari tabel *confusion matrix* dapat digunakan untuk menilai akurasi model *Decision Tree* dan model *Decision Tree* berbasis SMOTE yang telah dibuat ini.

4.5.1 Evaluasi *Decision Tree*

4.5.1.1 Evaluasi *Decision Tree* pada Aplikasi Shopee

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* pada aplikasi Shopee dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.9.

Tabel 4. 9 Evaluasi *Decision Tree* pada Aplikasi Shopee

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	558 (TN)	166 (FP)
	Positive	217 (FN)	518 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.9 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{518+558}{(518+558)+(166+217)} = 0.7374$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{518}{518+166} = 0.7573$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{518}{518+217} = 0.7047$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.7047 \times 0.7573)}{0.7047 + 0.7573} = 0.73$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- Nilai *accuracy* 0,7374 dengan persentase 73,74% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.
- Nilai *precision* sebesar 0.7573 dengan persentase 75,73% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- Nilai *recall* sebesar 0,7047 dengan persentase 70,47% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.

- d. Nilai *f1-score* sebesar 0.73 dengan persentase 73% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.1.2 Evaluasi Decision Tree pada Aplikasi Tokopedia

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* pada aplikasi Tokopedia dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.10.

Tabel 4. 10 Evaluasi *Decision Tree* pada Aplikasi Tokopedia

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	203 (TN)	18 (FP)
	Positive	25 (FN)	8 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.10 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{8+203}{(8+203)+(18+25)} = 0.8307$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{8}{8+18} = 0.3076$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{8}{(8+25)} = 0.2424$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.2424 \times 0.3076)}{0.2424 + 0.3076} = 0.2711$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- Nilai *accuracy* 0,8307 dengan persentase 83,07% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.
- Nilai *precision* sebesar 0.3076 dengan persentase 30,76% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.

- c. Nilai *recall* sebesar 0,2424 dengan persentase 24,24% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- d. Nilai *f1-score* sebesar 0,2711 dengan persentase 27,11% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.1.3 Evaluasi Decision Tree pada Aplikasi Lazada

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* pada aplikasi Lazada dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.11.

Tabel 4. 11 Evaluasi *Decision Tree* pada Aplikasi Lazada

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	60 (TN)	28 (FP)
	Positive	37 (FN)	81 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.11 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{81+60}{(81+60)+(28+37)} = 0.6844$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{81}{81+28} = 0.7431$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{81}{(81+37)} = 0.6864$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.6864 \times 0.7431)}{0.6864 + 0.7431} = 0.7138$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- a. Nilai *accuracy* 0.6844 dengan persentase 68,44% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.

- b. Nilai *precision* sebesar 0.7431 dengan persentase 74,31% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- c. Nilai *recall* sebesar 0.6864 dengan persentase 68,64% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- d. Nilai *f1-score* sebesar 0.7138 dengan persentase 71,38% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.1.4 Evaluasi Decision Tree pada Aplikasi Bukalapak

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* pada aplikasi Bukalapak dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.12.

Tabel 4. 12 Evaluasi *Decision Tree* pada Aplikasi Bukalapak

	Label Prediksi		
	<i>Class</i>	<i>Negative</i>	<i>Positive</i>
Label Aktual	<i>Negative</i>	90 (TN)	18 (FP)
	<i>Positive</i>	28 (FN)	47 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.12 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{47+90}{(47+90)+(18+28)} = 0.7486$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{47}{47+18} = 0.7230$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{47}{(47+28)} = 0.6266$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.6266 \times 0.7230)}{0.6266 + 0.7230} = 0.6713$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- a. Nilai *accuracy* 0,7486 dengan persentase 74,86% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.

- b. Nilai *precision* sebesar 0.7230 dengan persentase 72,30% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- c. Nilai *recall* sebesar 0,6266 dengan persentase 62,66% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- d. Nilai *f1-score* sebesar 0,6713 dengan persentase 67,13% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.2 Evaluasi *Decision Tree* berbasis SMOTE

4.5.2.1 Evaluasi *Decision Tree* berbasis SMOTE pada Aplikasi Shopee

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* berbasis SMOTE pada aplikasi Shopee dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.13.

Tabel 4. 13 Evaluasi *Decision Tree* berbasis SMOTE pada Aplikasi Shopee

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	544 (TN)	180 (FP)
	Positive	212 (FN)	523 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.13 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{523+544}{(523+544)+(180+212)} = 0.7313$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{523}{523+180} = 0.7439$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{523}{(523+212)} = 0.7115$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.7115 \times 0.7439)}{0.7115 + 0.7439} = 0.7273$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- Nilai *accuracy* 0,7313 dengan persentase 73,13% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.
- Nilai *precision* sebesar 0.7439 dengan persentase 74,39% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- Nilai *recall* sebesar 0.7115 dengan persentase 71,15% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- Nilai *f1-score* sebesar 0.7273 dengan persentase 72,73% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.2.2 Evaluasi Decision Tree berbasis SMOTE pada Aplikasi Tokopedia

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* berbasis SMOTE pada aplikasi Tokopedia dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.14.

Tabel 4. 14 Evaluasi *Decision Tree* berbasis SMOTE pada Aplikasi Tokopedia

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	186 (TN)	35 (FP)
	Positive	20 (FN)	13 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.14 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{13+186}{(13+186)+(35+20)} = 0.7834$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{13}{13+35} = 0.2708$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{13}{(13+20)} = 0.3939$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.3939 \times 0.2708)}{0.3939 + 0.2708} = 0.3209$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- Nilai *accuracy* 0,7834 dengan persentase 78,34% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.
- Nilai *precision* sebesar 0.2708 dengan persentase 27,08% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- Nilai *recall* sebesar 0,3939 dengan persentase 39,39% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- Nilai *f1-score* sebesar 0,3209 dengan persentase 32,09% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.2.3 Evaluasi Decision Tree berbasis SMOTE pada Aplikasi Lazada

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* berbasis SMOTE pada aplikasi Lazada dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.15.

Tabel 4. 15 Evaluasi *Decision Tree* Berbasis SMOTE pada Aplikasi Lazada

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	62 (TN)	26 (FP)
	Positive	33 (FN)	85 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.15 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$Accuracy = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{85+62}{(85+62)+(26+33)} = 0.7135$$

$$Precision = \frac{TP}{(TP+FP)} = \frac{85}{85+26} = 0.7657$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{85}{(85+33)} = 0.7203$$

$$F1\text{-score} = \frac{(2 \times \text{Recall} \times \text{Precision})}{\text{Recall} + \text{Precision}} = \frac{(2 \times 0.7203 \times 0.7657)}{0.7203 + 0.7657} = 0.7423$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- Nilai *accuracy* 0,7135 dengan persentase 71,35% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.
- Nilai *precision* sebesar 0.7657 dengan persentase 76,57% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- Nilai *recall* sebesar 0,7203 dengan persentase 72,03% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- Nilai *f1-score* sebesar 0,7423 dengan persentase 74,23% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.2.4 Evaluasi Decision Tree berbasis SMOTE pada Aplikasi Bukalapak

Data *confusion matrix* yang didapat setelah pengujian model *Decision Tree* berbasis SMOTE pada aplikasi Bukalapak dengan perbandingan 80:20 antara data *training* dan data *testing* dapat dilihat pada tabel 4.16.

Tabel 4. 16 Evaluasi *Decision Tree* Berbasis SMOTE pada Aplikasi Bukalapak

	Label Prediksi		
	Class	Negative	Positive
Label Aktual	Negative	84 (TN)	24 (FP)
	Positive	30 (FN)	45 (TP)

Selanjutnya, data yang diperoleh dari *confusion matrix* pada tabel 4.16 ini akan digunakan sebagai nilai-nilai untuk menghitung performa model, yang mencakup *accuracy*, *precision*, *recall*, dan *f1-score*. Prosedur penghitungan untuk menilai performa model sebagai berikut:

$$\text{Accuracy} = \frac{TP+TN}{(TP+TN)+(FP+FN)} = \frac{45+84}{(45+84)+(24+30)} = 0.7049$$

$$\text{Precision} = \frac{TP}{(TP+FP)} = \frac{45}{45+24} = 0.6521$$

$$Recall = \frac{TP}{(TP+FN)} = \frac{45}{(45+30)} = 0.60$$

$$F1-score = \frac{(2 \times Recall \times Precision)}{Recall + Precision} = \frac{(2 \times 0.60 \times 0.6521)}{0.60 + 0.6521} = 0.6249$$

Hasil yang diperoleh dari evaluasi *confusion matrix* dapat disimpulkan sebagai berikut:

- a. Nilai *accuracy* 0,7049 dengan persentase 70,49% menunjukkan bahwa rasio prediksi benar dari semua data yang diuji.
- b. Nilai *precision* sebesar 0.6521 dengan persentase 65,21% menunjukkan bahwa rasio prediksi benar positif dibandingkan dengan seluruh data aktual.
- c. Nilai *recall* sebesar 0,60 dengan persentase 60% menunjukkan bahwa hasil tersebut benar-benar positif dibandingkan seluruh hasil yang diprediksi sebesar.
- d. Nilai *f1-score* sebesar 0,6249 dengan persentase 62,49% adalah hasil dari nilai keseimbangan antara *recall* dan *precision*.

4.5.3 Analisis Hasil Evaluasi

Pada tahap ini dilakukan analisis terhadap model yang sudah dibuat yaitu model *Decision Tree* tanpa SMOTE dan model *Decision Tree* berbasis SMOTE.

4.5.3.1 Analisis Hasil Evaluasi pada Aplikasi Shopee

Tabel 4. 17 Analisis Hasil Evaluasi pada Aplikasi Shopee

Model	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
<i>Decision Tree</i> tanpa SMOTE	73,74%	75,73%	70,47%	73%
<i>Decision Tree</i> berbasis SMOTE	73,13%	74,39%	71,15%	72,73%

Berdasarkan tabel 4.17 dapat diperhatikan bahwa pada kasus analisis sentimen ulasan aplikasi Shopee, hasil performa pada model yang sudah dibuat mendapat peningkatan dan penurunan pada performa menggunakan SMOTE. Dimana model *Decision Tree* berbasis SMOTE mendapatkan penurunan *accuracy* dari 73,74 persen

menjadi 73,13 persen, penurunan *precision* dari 75,73 persen menjadi 74,39 persen, kenaikan *recall* dari 70,47 persen menjadi 71,15 persen, dan penurunan *f1-score* dari 73 persen menjadi 72,73 persen.

4.5.3.2 Analisis Hasil Evaluasi pada Aplikasi Tokopedia

Tabel 4. 18 Analisis Hasil Evaluasi pada Aplikasi Tokopedia

Model	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
<i>Decision Tree</i> tanpa SMOTE	83,07%	30,76%	24,24%	27,11%
<i>Decision Tree</i> berbasis SMOTE	78,34%	27,08%	39,39%	32,09%

Berdasarkan tabel 4.18 dapat diperhatikan bahwa pada kasus analisis sentimen ulasan aplikasi Tokopedia, hasil performa pada model yang sudah dibuat mendapat peningkatan dan penurunan pada performa menggunakan SMOTE. Dimana model *Decision Tree* berbasis SMOTE mendapatkan penurunan *accuracy* dari 83,07 persen menjadi 78,34 persen, penurunan *precision* dari 30,76 persen menjadi 27,08 persen, kenaikan *recall* dari 24,24 persen menjadi 39,39 persen, dan kenaikan *f1-score* dari 27,11 persen menjadi 32,09 persen.

4.5.3.3 Analisis Hasil Evaluasi pada Aplikasi Lazada

Tabel 4. 19 Analisis Hasil Evaluasi pada Aplikasi Lazada

Model	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
<i>Decision Tree</i> tanpa SMOTE	68,44%	74,31%	68,64%	71,38%
<i>Decision Tree</i> berbasis SMOTE	71,35%	76,57%	72,03%	74,23%

Berdasarkan tabel 4.19 dapat diperhatikan bahwa pada kasus analisis sentimen ulasan aplikasi Lazada, hasil performa pada model yang sudah dibuat mendapat peningkatan pada performa menggunakan SMOTE. Dimana model *Decision Tree*

berbasis SMOTE mendapatkan kenaikan *accuracy* dari 68,44 persen menjadi 71,35 persen, kenaikan *precision* dari 74,31 persen menjadi 76,57 persen, kenaikan *recall* dari 68,64 persen menjadi 72,03 persen, dan kenaikan *f1-score* dari 71,38 persen menjadi 74,23 persen.

4.5.3.4 Analisis Hasil Evaluasi pada Aplikasi Bukalapak

Tabel 4. 20 Analisis Hasil Evaluasi pada Aplikasi Bukalapak

Model	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
<i>Decision Tree</i> tanpa SMOTE	74,86%	72,30%	62,66%	67,13%
<i>Decision Tree</i> berbasis SMOTE	70,49%	65,21%	60%	62,49%

Berdasarkan tabel 4.20 dapat diperhatikan bahwa pada kasus analisis sentimen ulasan aplikasi Bukalapak, hasil performa pada model yang sudah dibuat mendapat penurunan pada performa menggunakan SMOTE. Dimana model *Decision Tree* berbasis SMOTE mendapatkan penurunan *accuracy* dari 74,86 persen menjadi 70,49 persen, penurunan *precision* dari 72,30 persen menjadi 65,21 persen, penurunan *recall* dari 62,66 persen menjadi 60 persen, dan penurunan *f1-score* dari 67,13 persen menjadi 62,49 persen.

4.6 Visualisasi

Tahap visualisasi dilakukan untuk menggambarkan hasil penelitian berupa *wordcloud* sentimen positif dan *wordcloud* sentimen negatif mengenai aplikasi Shopee. *Wordcloud* menampilkan kata-kata yang paling banyak muncul pada data yang digunakan, di mana ukuran kata menunjukkan seberapa sering kata tersebut muncul dalam data tersebut. Pada tahapan ini memanfaatkan *library* pada *python*.

4.6.1 Wordcloud Sentimen Negatif

Berdasarkan jumlah kemunculan kata pada ulasan dalam data, didapatkan hasil wordcloud dari sentimen negatif ulasan pengguna aplikasi Shopee yang dapat dilihat pada gambar berikut.



Gambar 4. 7 Wordcloud Sentimen Negatif

Dari gambar di atas, kata-kata sentimen negatif yang paling banyak muncul didasarkan pada banyaknya kemunculan kata pada korpus kelas negatif. Kata yang paling banyak muncul berdasarkan besarnya ukuran kata pada *wordcloud*.

4.6.2 Wordcloud Sentimen Positif

Berdasarkan jumlah kemunculan kata pada ulasan dalam data, didapatkan hasil *wordcloud* dari sentimen positif ulasan pengguna aplikasi Shopee yang dapat dilihat pada gambar berikut.

