

## **BAB III**

### **ANALISIS DAN PERANCANGAN SISTEM**

#### **3.1 Analisis Sistem**

Sistem informasi sentimen masyarakat terkait aplikasi PLN Mobile merupakan suatu sistem yang dirancang untuk mengelola dan menganalisis data komentar pengguna aplikasi PLN Mobile dalam rentang waktu tertentu. Tujuan sistem untuk memberikan wawasan yang lebih mendalam mengenai tingkat kepuasan pengguna serta berbagai masalah yang dihadapi selama menggunakan aplikasi. Sistem ini menggunakan bahasa pemrograman Python, dengan memanfaatkan teknik web crawling yang didukung oleh auth token API X untuk mengambil data komentar secara otomatis dari platform tertentu. Data yang diambil mencakup komentar pengguna aplikasi PLN Mobile yang dikumpulkan selama periode 01 Januari 2024 hingga 10 September 2024, yang mencakup sekitar sembilan bulan. Melalui analisis sentimen, sistem ini dapat membantu mengidentifikasi tren kepuasan pengguna, keluhan, dan masalah yang sering muncul, sehingga pengembang dapat mengambil langkah-langkah yang diperlukan untuk meningkatkan kualitas layanan dan fitur aplikasi. Informasi yang dihasilkan dari sistem ini diharapkan dapat menjadi acuan bagi pengambilan keputusan dalam pengembangan aplikasi PLN Mobile ke depannya, serta memperkuat hubungan antara pengguna dan pengembang dalam menciptakan pengalaman pengguna yang lebih baik.

#### **3.2 Hasil Analisis**

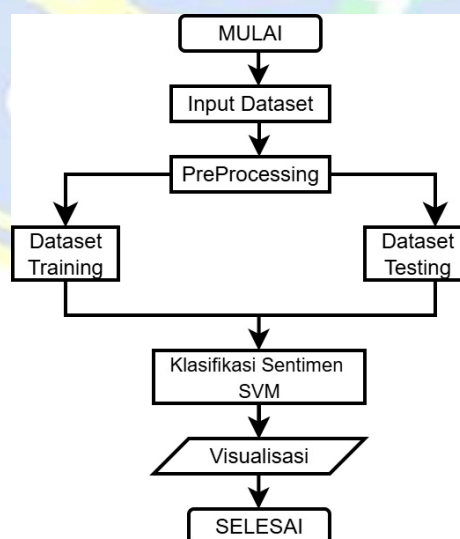
Dalam penelitian ini dihasilkan analisis sistem yang dapat dilakukan dari pengklasifikasian sentimen tweet yang membahas PLN dapat melakukan klasifikasi sentimen yang terkandung dalam tweet di media sosial X. Pembuatan sistem klasifikasi sentimen untuk PLN Mobile menggunakan metode *Support Vector Machine*. Metode ini membutuhkan data tweet yang sudah diberi label sentimen terlebih dahulu sebagai data latih dengan data latih ini berfungsi sebagai panduan dalam menentukan kelas sentimen dari tweet baru yang belum diberi label. Setelah dilakukan klasifikasi dengan *Support Vector Machine*, setiap tweet akan diberi

kelas seperti sentimen positif, netral, atau negatif. Sehingga masing-masing tweet akan memiliki kelas sentimen yang dapat digunakan untuk mengukur opini publik terkait penggunaan dan keluhan pada aplikasi PLN Mobile. Observasi dilakukan untuk mengumpulkan informasi dan data melalui pengamatan serta tinjauan langsung terhadap topik penelitian, guna memperoleh pemahaman yang lebih mendalam mengenai objek atau fenomena yang sedang diteliti. Topik tersebut adalah tanggapan dari pengguna media sosial X terhadap penggunaan pada aplikasi PLN Mobile. Observasi dilakukan pada tanggal 01 Januari 2024 hingga 10 September 2024 dengan jumlah sebanyak 1400 komentar masyarakat di media sosial X.

### 3.3 Perancangan Sistem

#### 3.3.1 Flowchart Sistem

Sistem yang akan dibangun adalah aplikasi untuk menganalisis sentimen terhadap aplikasi PLN Mobile di media sosial X menggunakan metode SVM untuk klasifikasi tweet. Aplikasi ini dirancang untuk menganalisis sentimen publik terkait PLN Mobile melalui platform media sosial X. Aplikasi ini, diharapkan dapat membantu pemerintah dan pihak terkait dengan lebih memahami opini publik secara lebih efektif, yang nantinya dapat mendukung pengambilan keputusan yang lebih baik. Gambaran umum sistem yang akan dibangun seperti gambar 3.1.



**Gambar 3.1** Gambaran Flowchart Sistem

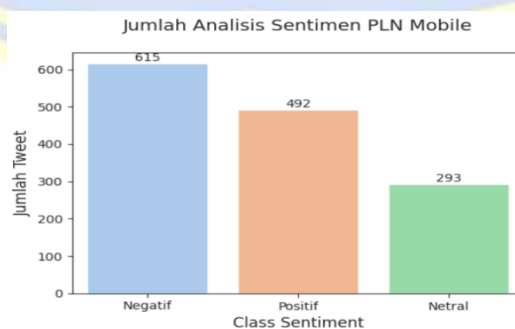
### 3.3.2 Representasi Dataset

Pada tahapan ini dilakukan pengumpulan data dengan cara crawling menggunakan Python melalui komponen Search X API. Tweet yang diambil berisi opini mengenai penggunaan aplikasi PLN Mobile dengan menggunakan query "@pln\_123". Proses crawling ini dilakukan mulai dari tanggal 01 Januari 2024 hingga 10 September 2024 dengan jumlah sebanyak 1400 data tweet. Proses crawling dilakukan dengan memanfaatkan Tweepy library untuk mengakses data dari X, dengan kata kunci dan filter tertentu yang relevan dengan topik penelitian. Berikut adalah contoh hasil crawling dataset dengan menggunakan bahasa python.

**Tabel 3.1** Contoh Hasil Crawling Menggunakan Python

Username	Review
plnup3pbun	Pertama dowload karena ada masalah kelistrikan. Chat cs via aplikasi langsung ditindak hari itu juga. Gak sampe 1 jam dari laporan, pengerjaan sudah teratasi dan selesai. Pelayanan OKE, TOP deh.
kawmeisya	Error teruss, otp lama, loading tiap halaman lama, catat meter errornya id tidak di temukan, full masalah ini aplikasi
Ratratratra	Proses transaksi cepat dan mudah, sayang menggunakan App punya PLN tetapi masih harus bayar administrasi, sebaiknya betul2 bebas biaya. Terimakasih

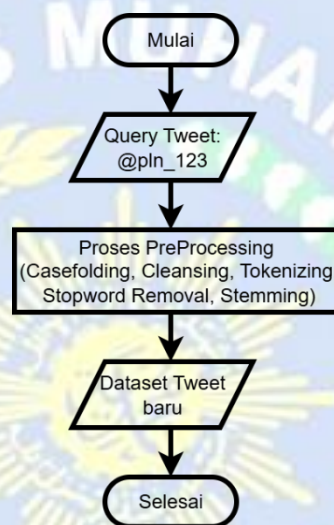
Pada tahap ini analisis data, sentimen dari setiap komentar diidentifikasi dan dikalsifikasikan menjadi tiga kategori: positif, netral dan negatif. Hasil dari klasifikasi sentimen tersebut ditunjukkan pada gambar 3.2.



**Gambar 3.2** Hasil Data Sentimen terhadap PLN

### 3.3.3 Preprocessing Data

Tahap *Preprocessing* adalah langkah penting dalam analisis teks untuk membersihkan dan menyiapkan data sebelum diolah oleh model. Proses ini dilakukan secara manual dengan memastikan data lebih terstruktur, sehingga meningkatkan akurasi dan efisiensi model dalam pemrosesan teks. Pada penelitian ini, preprocessing diterapkan pada 1.400 data ulasan, yang dibagi menjadi 1.120 data training dan 280 data testing. Langkah-langkah ditunjukkan pada Gambar 3.3.



**Gambar 3.3** Alur Preprocessing

#### 3.3.3.1 Casefolding

Pada tahap ini bertujuan untuk mengubah semua huruf dalam teks menjadi bentuk huruf kecil (lowercase). Langkah ini penting untuk mengurangi variasi dalam data teks, terutama ketika suatu analisis tidak memperhitungkan perbedaan antara huruf kapital dan huruf kecil. Sebagai teks review dari proses *case folding*, contoh *tweet* dapat dilihat sebelum dan sesudah pada tabel 3.2.

**Tabel 3.2** Contoh Tahap Casefolding

Casefolding	
Sebelum	Sesudah
Pertama dowload karena ada masalah kelistrikan. Chat cs via aplikasi langsung ditindak hari itu juga. Gak sampe 1 jam dari laporan, pengerjaan	pertama dowload karena ada masalah kelistrikan. chat cs via aplikasi langsung ditindak hari itu juga. gak sampe 1 jam dari laporan, pengerjaan

Casefolding	
Sebelum	Sesudah
sudah teratasi dan selesai. Pelayanan OKE, TOP deh.	sudah teratasi dan selesai. pelayanan oke, top deh.
Error terusss, otp lama, loading tiap halaman lama, catat meter errornya id tidak di temukan, full masalah ini aplikasi	error terusss, otp lama, loading tiap halaman lama, catat meter errornya id tidak di temukan, full masalah ini aplikasi
Proses transaksi cepat dan mudah, sayang menggunakan App punya PLN tetapi masih harus bayar administrasi, sebaiknya betul2 bebas biaya. Terimakasih	proses transaksi cepat dan mudah, sayang menggunakan app punya pln tetapi masih harus bayar administrasi, sebaiknya betul2 bebas biaya. terimakasih

### 3.3.3.2 Cleansing

Tahap *cleansing* adalah proses membersihkan data dari elemen-elemen yang tidak diperlukan, seperti tanda baca, simbol, URL, tag HTML, emotikon, koma, dan spasi tambahan, dan karakter khusus lainnya. Selain itu, pada tahap ini dilakukan beberapa proses, diantaranya penghapusan missing values data, menghapus simbol seperti '!"#\$\$%&\'()\*+,-./:;<=>?@[\\]^\_`{|}~', karakter spesial, hingga angka. Tujuan *cleansing* adalah untuk meningkatkan kualitas data guna memudahkan analisis dan meningkatkan akurasi hasil pada tahap pemrosesan selanjutnya seperti klasifikasi sentimen. Sebagai teks review dari proses *cleansing*, contoh *tweet* dapat dilihat sebelum dan sesudah pada tabel 3.3.

**Tabel 3.3** Contoh Tahap Cleansing

Cleansing	
Sebelum	Sesudah
pertama dowload karena ada masalah kelistrikan. chat cs via aplikasi langsung ditindak hari itu juga. gak sampe 1 jam dari laporan, pengerjaan	pertama dowload karena ada masalah kelistrikan chat cs via aplikasi langsung ditindak hari itu juga gak sampe jam dari laporan pengerjaan sudah teratasi dan selesai pelayanan oke top deh

Cleansing	
Sebelum	Sesudah
sudah teratasi dan selesai. pelayanan oke, top deh.	
error terusss, otp lama, loading tiap halaman lama, catat meter errornya id tidak di temukan, full masalah ini aplikasi	error terusss otp lama loading tiap halaman lama catat meter errornya id tidak di temukan full masalah ini aplikasi
proses transaksi cepat dan mudah, sayang menggunakan app punya pln tetapi masih harus bayar administrasi, sebaiknya betul2 bebas biaya. terimakasih	proses transaksi cepat dan mudah sayang menggunakan app punya pln tetapi masih harus bayar administrasi sebaiknya betul bebas biaya terimakasih

### 3.3.3.3 Tokenizing

Tokenizing adalah proses memecah teks menjadi unit-unit kecil yang disebut token, seperti kata, frasa, atau simbol, yang digunakan dalam analisis sentimen. Pada umumnya, teks dipecah menjadi token kata (word tokenization) di mana setiap kata dianggap sebagai satu unit analisis. Proses ini memungkinkan algoritma untuk mengidentifikasi dan memproses setiap kata secara terpisah, membantu dalam memahami struktur kalimat dan makna yang terkandung sehingga model dapat menganalisis sentimen secara lebih efisien dan akurat. Sebagai teks review dari proses *tokenizing*, contoh *tweet* dapat dilihat sebelum dan sesudah pada tabel 3.4.

**Tabel 3.4** Contoh Tahap Tokenizing

Tokenizing	
Sebelum	Sesudah
pertama dowload karena ada masalah kelistrikan chat cs via aplikasi langsung ditindak hari itu juga gak sampe jam dari laporan pengerjaan sudah teratasi dan selesai pelayanan oke top deh	['pertama', 'dowload', 'karena', 'ada', 'masalah', 'kelistrikan', 'chat', 'cs', 'via', 'aplikasi', 'langsung', 'ditindak', 'hari', 'itu', 'juga', 'gak', 'sampe', 'jam', 'dari', 'laporan', 'pengerjaan', 'sudah', 'teratasi',

Tokenizing	
Sebelum	Sesudah
	'dan', 'selesai', 'pelayanan', 'oke', 'top', 'deh']
error teruss otp lama loading tiap halaman lama catat meter errornya id tidak di temukan full masalah ini aplikasi	['error', 'teruss', 'otp', 'lama', 'loading', 'tiap', 'halaman', 'lama', 'catat', 'meter', 'errornya', 'id', 'tidak', 'di', 'temukan', 'full', 'masalah', 'ini', 'aplikasi']
proses transaksi cepat dan mudah sayang menggunakan app punya pln tetapi masih harus bayar administrasi sebaiknya betul bebas biaya terimakasih	['proses', 'transaksi', 'cepat', 'dan', 'mudah', 'sayang', 'menggunakan', 'app', 'punya', 'pln', 'tetapi', 'masih', 'harus', 'bayar', 'administrasi', 'sebaiknya', 'betul', 'bebas', 'biaya', 'terimakasih']

#### 3.3.3.4 Stopword Removal

Tahap *stopword* adalah proses menghapus kata-kata yang sering muncul dalam teks tetapi tidak memiliki nilai signifikan dalam analisis sentimen. Kata-kata seperti "dan," "yang," "atau," dan "di" biasanya tidak berkontribusi pada pemahaman konteks atau makna suatu kalimat. Dalam preprocessing, *stopword* removal membantu mengurangi *noise* atau gangguan dalam data, sehingga hanya kata-kata yang penting untuk analisis yang tersisa. Menghapus *stopwords*, ukuran data yang diolah menjadi lebih kecil dan model analisis dapat fokus pada kata-kata yang relevan, sehingga meningkatkan akurasi dalam penentuan sentimen. Sebagai teks review dari proses *stopword*, contoh *tweet* dapat dilihat sebelum dan sesudah pada tabel 3.5.

**Tabel 3.5** Contoh Tahap Stopword Removal

Stopword Removal	
Sebelum	Sesudah
['pertama', 'download', 'karena', 'ada', 'masalah', 'kelistrikan', 'chat', 'cs', 'via', 'aplikasi', 'langsung', 'ditindak', 'hari', 'itu', 'juga', 'gak', 'sampe', 'jam', 'dari',	['download', 'kelistrikan', 'chat', 'cs', 'via', 'aplikasi', 'langsung', 'ditindak', 'gak', 'sampe', 'jam', 'laporan', 'pengerjaan',

Stopword Removal	
Sebelum	Sesudah
'laporan', 'pengerjaan', 'sudah', 'teratasi', 'dan', 'selesai', 'pelayanan', 'oke', 'top', 'deh']	'teratasi', 'selesai', 'pelayanan', 'oke', 'top', 'deh']
['error', 'terusss', 'otp', 'lama', 'loading', 'tiap', 'halaman', 'lama', 'catat', 'meter', 'errornya', 'id', 'tidak', 'di', 'temukan', 'full', 'masalah', 'ini', 'aplikasi']	['error', 'terusss', 'otp', 'loading', 'halaman', 'catat', 'meter', 'errornya', 'id', 'temukan', 'full', 'aplikasi']
['proses', 'transaksi', 'cepat', 'dan', 'mudah', 'sayang', 'menggunakan', 'app', 'punya', 'pln', 'tetapi', 'masih', 'harus', 'bayar', 'administrasi', 'sebaiknya', 'betul', 'bebas', 'biaya', 'terimakasih']	['proses', 'transaksi', 'cepat', 'mudah', 'sayang', 'app', 'pln', 'bayar', 'administrasi', 'bebas', 'biaya', 'terimakasih']

### 3.3.3.5 Stemming

Tahap stemming adalah proses mengubah kata menjadi bentuk dasar atau akarnya dengan menghilangkan akhiran kata, seperti "-ing," "-ed," atau "-an." Tujuan stemming adalah menyederhanakan variasi kata sehingga model analisis sentimen dapat mengenali kata-kata dengan makna yang sama meskipun memiliki bentuk yang berbeda. Misalnya, kata "makan," "memakan," dan "makanan" akan diubah menjadi akar kata "makan." Oleh karena itu stemming, maka jumlah kata unik dalam data berkurang, sehingga model lebih efisien dalam mengolah teks dan mengidentifikasi pola sentimen yang lebih akurat. Sebagai teks review dari proses *Stemming*, contoh *tweet* dapat dilihat sebelum dan sesudah pada tabel 3.6.

**Tabel 3.6** Contoh Tahap Stemming

Stemming	
Sebelum	Sesudah
['dowload', 'kelistrikan', 'chat', 'cs', 'via', 'aplikasi', 'langsung', 'ditindak', 'gak', 'sampe', 'jam', 'laporan', 'pengerjaan',	dowload listrik chat cs via aplikasi langsung tindak gak sampe jam lapor kerja atas selesai layan oke top deh

Stemming	
Sebelum	Sesudah
'teratasi', 'selesai', 'pelayanan', 'oke', 'top', 'deh']	
['error', 'teruss', 'otp', 'loading', 'halaman', 'catat', 'meter', 'errornya', 'id', 'temukan', 'full', 'aplikasi']	error teruss otp loading halaman catat meter errornya id temu full aplikasi
['proses', 'transaksi', 'cepat', 'mudah', 'sayang', 'app', 'pln', 'bayar', 'administrasi', 'bebas', 'biaya', 'terimakasih']	proses transaksi cepat mudah sayang app pln bayar administrasi bebas biaya terimakasih

### 3.3.4 Ekstraksi Fitur

Pada tahapan ekstraksi fitur menggunakan pembobotan algoritma TF-IDF, tujuannya adalah digunakan untuk mengukur seberapa penting sebuah kata dalam sebuah dokumen relatif terhadap kumpulan dokumen untuk di ubah menjadi sebuah vektor. Fitur yang diekstrak menggunakan TF-IDF berfungsi mengidentifikasi kata-kata unik dan penting dalam suatu dokumen, sehingga sering digunakan dalam tugas-tugas seperti klasifikasi teks, clustering, atau pencarian informasi. Metode ini efektif karena mampu menekan pengaruh kata-kata umum yang banyak muncul di dalam suatu dokumen.

1. Tweet yang sudah melalui tahap *preprocessing* untuk perhitungan TF-IDF

**Tabel 3.7** Review Tweet

<b>D1</b>	aplikasi bagus bantu upgrade prabayar
<b>D2</b>	aplikasi trouble bikin susah rakyat
<b>D3</b>	metode bayar sedia aplikasi pln mobile
<b>D4</b>	aplikasi bagus transaksi beli token masuk informasi
<b>D5</b>	sulit daftar masuk aplikasi
<b>D6</b>	aplikasi pln mobile update proses load
<b>D7</b>	aplikasi pln mobile tanggap tangani keluhan voltase rendah
<b>D8</b>	aplikasi bagus bantu konsum

Proses pembobotan kata dimulai dengan mengambil kata-kata yang telah melalui tahap *Preprocessing* dari database. Kemudian, frsekuensi kemunculan

setiap kata dalam tiap *tweet* dihitung. Perhitungan ini menggunakan kata-kata hasil *Preprocessing* dan menghasilkan bobot untuk masing-masing kata seperti pada tabel 3.8 berikut.

## 2. Menghitung pembobotan TF-IDF

**Tabel 3.8** Menghitung TF-IDF

TERM	TF								$IDF = (\log \frac{6}{df})$
	D1	D2	D3	D4	D5	D6	D7	D8	
aplikasi	1	1	1	1	1	1	1	1	0
metode	0	0	1	0	0	0	0	0	0,77815125
bagus	1	0	0	1	0	0	0	1	0,47715125
trouble	0	1	0	0	0	0	0	0	0,77815125
bayar	0	0	1	0	0	0	0	0	0,77815125
bantu	1	0	0	0	0	0	0	1	0,77815125
bikin	0	1	0	0	0	0	0	0	0,77815125
sedia	0	0	1	0	0	0	0	0	0,77815125
upgrade	1	0	0	0	0	0	0	0	0,77815125
susah	0	1	0	0	0	0	0	0	0,77815125
prabayar	1	0	0	0	0	0	0	0	0,77815125
rakyat	0	1	0	0	0	0	0	0	0,77815125
pln	0	0	1	0	0	1	1	0	0,47715125
mobile	0	0	1	0	0	1	1	0	0,47715125
transaksi	0	0	0	1	0	0	0	0	0,77815125
beli	0	0	0	1	0	0	0	0	0,77815125
token	0	0	0	1	0	0	0	0	0,77815125
masuk	0	0	0	1	1	0	0	0	0,47715125
informasi	0	0	0	1	0	0	0	0	0,77815125
sulit	0	0	0	0	1	0	0	0	0,77815125
daftar	0	0	0	0	1	0	0	0	0,77815125
update	0	0	0	0	0	1	0	0	0,77815125
proses	0	0	0	0	0	1	0	0	0,77815125
load	0	0	0	0	0	1	0	0	0,77815125
tanggap	0	0	0	0	0	0	1	0	0
tangani	0	0	0	0	0	0	1	0	0
keluh	0	0	0	0	0	0	1	0	0
voltase	0	0	0	0	0	0	1	0	0
rendah	0	0	0	0	0	0	1	0	0
konsum	0	0	0	0	0	0	0	1	0

**Tabel 3.9** Hasil Pembobotan TF-IDF

TERM	TF*IDF							
	D1	D2	D3	D4	D5	D6	D7	D8
aplikasi	0	0	0	0	0	0	0	0
metode	0	0	0,7781	0	0	0	0	0
bagus	0,4771	0	0	0,4771	0	0	0	0,4771
trouble	0	0,7781	0	0	0	0	0	0
bayar	0	0	0,7781	0	0	0	0	0

TERM	TF*IDF							
	D1	D2	D3	D4	D5	D6	D7	D8
bantu	0,7781	0	0	0	0	0	0	0,7781
bikin	0	0,7781	0	0	0	0	0	0
sedia	0	0	0,7781	0	0	0	0	0
upgrade	0,7781	0	0	0	0	0	0	0
susah	0	0,7781	0	0	0	0	0	0
prabayar	0,7781	0	0	0	0	0	0	0
rakyat	0	0,7781	0	0	0	0	0	0
pln	0	0	0,4771	0	0	0,4771	0,4771	0
mobile	0	0	0,4771	0	0	0,4771	0,4771	0
transaksi	0	0	0	0,7781	0	0	0	0
beli	0	0	0	0,7781	0	0	0	0
token	0	0	0	0,7781	0	0	0	0
masuk	0	0	0	0,4771	0,4771	0	0	0
informasi	0	0	0	0,7781	0	0	0	0
sulit	0	0	0	0	0,7781	0	0	0
daftar	0	0	0	0	0,7781	0	0	0
update	0	0	0	0	0	0,7781	0	0
proses	0	0	0	0	0	0,7781	0	0
load	0	0	0	0	0	0,7781	0	0
tanggap	0	0	0	0	0	0	0	0
tangani	0	0	0	0	0	0	0	0
keluh	0	0	0	0	0	0	0	0
voltase	0	0	0	0	0	0	0	0
rendah	0	0	0	0	0	0	0	0
konsum	0	0	0	0	0	0	0	0

### 3.3.5 Klasifikasi *Support Vector Machine*

Pada tahap ini setelah dilakukan data ulasan diubah menjadi vektor, dataset kemudian dibagi menjadi dua bagian menggunakan fungsi `train_test_split` dalam model dilatih menggunakan algoritma *Support Vector Machine*, langkah terakhir adalah menguji model tersebut menggunakan data uji untuk melakukan klasifikasi sentimen. Kata-kata yang telah melalui tahap *Preprocessing* dan dihitung frekuensi kemunculannya, seperti yang ditunjukkan pada tabel 3.9, digunakan sebagai input awal. Langkah – langkah klasifikasi *Support Vector Machine* adalah sebagai berikut:

#### 1. Menghitung Klasifikasi

Setelah mendapatkan nilai dari perhitungan TF\*IDF dari perdokumen dapat dimasukkan nilai tersebut pada tabel 3.10 untuk perhitungan SVM.

**Tabel 3.10** Perhitungan Klasifikasi SVM

TERM	PERHITUNGAN SVM							
	D1	D2	D3	D4	D5	D6	D7	D8
aplikasi	0	0	0	0	0	0	0	0
metode	0	0	0,7781	0	0	0	0	0
bagus	0,4771	0	0	0,4771	0	0	0	0,4771
trouble	0	0,7781	0	0	0	0	0	0
bayar	0	0	0,7781	0	0	0	0	0
bantu	0,7781	0	0	0	0	0	0	0,7781
bikin	0	0,7781	0	0	0	0	0	0
sedia	0	0	0,7781	0	0	0	0	0
upgrade	0,7781	0	0	0	0	0	0	0
susah	0	0,7781	0	0	0	0	0	0
prabayar	0,7781	0	0	0	0	0	0	0
rakyat	0	0,7781	0	0	0	0	0	0
pln	0	0	0,4771	0	0	0,4771	0,4771	0
mobile	0	0	0,4771	0	0	0,4771	0,4771	0
transaksi	0	0	0	0,7781	0	0	0	0
beli	0	0	0	0,7781	0	0	0	0
token	0	0	0	0,7781	0	0	0	0
masuk	0	0	0	0,4771	0,4771	0	0	0
informasi	0	0	0	0,7781	0	0	0	0
sulit	0	0	0	0	0,7781	0	0	0
daftar	0	0	0	0	0,7781	0	0	0
update	0	0	0	0	0	0,7781	0	0
proses	0	0	0	0	0	0,7781	0	0
load	0	0	0	0	0	0,7781	0	0
tanggap	0	0	0	0	0	0	0	0
tangani	0	0	0	0	0	0	0	0
keluh	0	0	0	0	0	0	0	0
voltase	0	0	0	0	0	0	0	0
rendah	0	0	0	0	0	0	0	0
konsum	0	0	0	0	0	0	0	0

## 2. Menentukan nilai bias

Nilai bias didapatkan dari hasil sentimen yang dimisalkan menjadi angka.

Setelah itu nilai bias dijumlahkan semua untuk mendapatkan nilai bias

**Tabel 3.11** Menghitung Bias

No.	Dokumen Review	Label
<b>D1</b>	aplikasi bagus bantu upgrade prabayar	1
<b>D2</b>	aplikasi trouble bikin susah rakyat	-1
<b>D3</b>	metode bayar sedia aplikasi pln mobile	0
<b>D4</b>	aplikasi bagus transaksi beli token masuk informasi	1
<b>D5</b>	sulit daftar masuk aplikasi	-1

No.	Dokumen Review	Label
D6	aplikasi pln mobile update proses load	0
	<b>Bias</b>	<b>0</b>

### 3. Menghitung nilai W

Langkah menghitung bobot  $w$  dimulai dengan mengalikan nilai TF-IDF setiap term pada dokumen latih dengan label kelas (misalnya +1 untuk kelas target, -1 untuk lainnya). Hasil perkalian tiap term dijumlahkan per term di seluruh dokumen.

**Tabel 3.12** Menghitung W

TERM	W
aplikasi	0
metode	0
bagus	0,95424251
trouble	-0,77815125
bayar	0
bantu	0,77815125
bikin	-0,77815125
sedia	0
upgrade	0,77815125
susah	-0,77815125
prabayar	0,77815125
rakyat	-0,77815125
pln	0
mobile	0
transaksi	0,77815125
beli	0,77815125
token	0,77815125
masuk	0
informasi	0,77815125
sulit	-0,77815125
daftar	-0,77815125
update	0
proses	0
load	0
tanggap	0
tangani	0
keluh	0

### 4. Hasil klasifikasi SVM

Langkah hasil klasifikasi dilakukan dengan menjumlahkan nilai pada tabel 3.10 dan  $w$  dokumen data uji pada tiap term D7 dan D8, kemudian hasilnya dijumlahkan secara keseluruhan. Setelah itu, tambahkan nilai bias ke total

tersebut. Nilai akhir disebut margin, dan kelas dengan margin tertinggi adalah hasil prediksi.

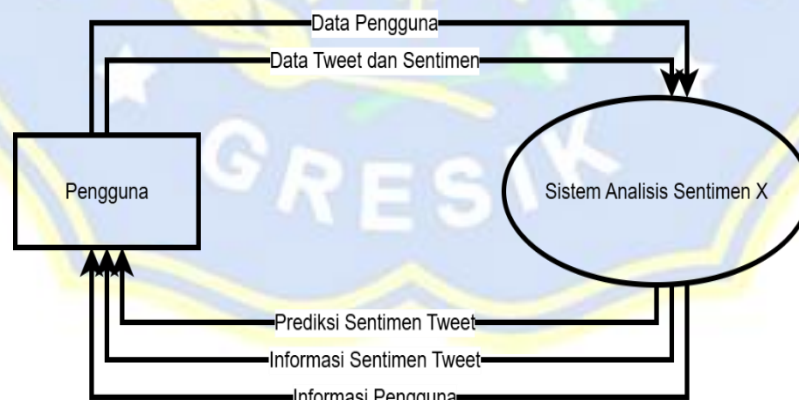
**Tabel 3.13** Prediksi Klasifikasi SVM

DATA UJI	PREDIKSI	HASIL PREDIKSI	SENTIMEN
aplikasi pln mobile tanggap tangani keluh voltase rendah	0	0	NETRAL
aplikasi bagus bantu konsum	1,060808752	+1	POSITIF

Untuk klasifikasi menggunakan Support Vector Machine (SVM) pada dataset ulasan dilakukan dengan mengubah teks menjadi vektor melalui proses pembobotan kata menggunakan metode TF-IDF. Hasil klasifikasi ditampilkan dalam kolom Sentimen, yang menunjukkan bagaimana kata-kata tertentu berkontribusi terhadap sentimen dalam masing-masing dokumen. Dengan demikian, SVM memanfaatkan pola distribusi bobot kata untuk menentukan kategori sentimen secara otomatis berdasarkan parameter tertentu.

### 3.4 Diagram Konteks

Diagram konteks adalah diagram yang dapat menggambarkan satu proses utama dalam suatu sistem, menunjukkan interaksi antara sistem tersebut dengan elemen-elemen eksternal terkait, serta aliran informasi di antaranya. Dalam penelitian ini, rangkaian diagram konteks yang digunakan ditunjukkan pada Gambar 3.4.



**Gambar 3.4** Diagram Konteks

Diagram konteks menggambarkan alur interaksi antara pengguna dan sistem analisis sentimen X. Pengguna memasukkan data profil dan tweet sebagai input utama. Sistem kemudian melakukan preprocessing data untuk menyiapkan informasi sebelum tahap analisis sentimen yang menentukan apakah tweet bersifat positif, negatif, atau netral. Hasil akhirnya berupa prediksi sentimen, sehingga menunjukkan aliran data dan fungsi sistem secara menyeluruh.

### 3.5 Perancangan Antarmuka

#### a. Antar Muka Halaman Dashboard

Halaman Dashboard adalah halaman menampilkan halaman informasi mengenai nama sistem dan panjang data ulasan. Tampilan halaman dapat dilihat pada Gambar 3.5.

ANALISIS SENTIMEN APLIKASI PLN MOBILE	
	<b>DASHBOARD</b>
<b>Dashboard</b>	Selamat Datang di Sistem Analisis Sentimen Aplikasi PLN MOBILE
Dataset	
Klasifikasi SVM	
Visualisasi	
Logout	

**Gambar 3.5** Tampilan Halaman Dashboard

#### b. Antar Muka Halaman Dataset

Halaman Dataset menampilkan data ulasan analisis sentimen, di mana dapat melihat, menambah, mengubah, atau menghapus data tweet. Data mencakup teks tweet, label, dan tipe data. Tampilan halaman ini ada pada Gambar 3.6.

ANALISIS SENTIMEN APLIKASI PLN MOBILE			
	<b>DATASET</b>		
Dashboard	<input type="button" value="Input File"/>	<input type="button" value="Reset Data"/>	
<b>Dataset</b>	Data Ulasan	Label	Tipe Data
Klasifikasi SVM	1. Teks Ulasan	Positif	Testing
Visualisasi	2. Teks Ulasan	Negatif	Training
Logout	3. Teks Ulasan	Netral	Testing

**Gambar 3.6** Tampilan Halaman Dataset

c. Antar Muka Halaman Klasifikasi

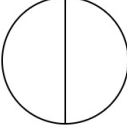
Halaman Klasifikasi dapat menampilkan data ulasan, label sentimen asli, hasil prediksi model SVM, serta tombol akurasi untuk evaluasi kinerja model. Tampilan halaman ini ada pada Gambar 3.7.

ANALISIS SENTIMEN APLIKASI PLN MOBILE						
	<b>KLASIFIKASI</b>					
Dashboard	<input type="button" value="Akurasi"/>					
<b>Dataset</b>	Data Ulasan	Positif	Negatif	Netral	Label	Prediksi
<b>Klasifikasi</b>	1. Teks Ulasan	10	9	11	Positif	Netral
Visualisasi	2. Teks Ulasan	10	11	14	Negatif	Positif
Logout	3. Teks Ulasan	13	12	10	Netral	Negatif

**Gambar 3.7** Tampilan Halaman Klasifikasi SVM

d. Antar Muka Halaman Visualisasi

Halaman Visualisasi menampilkan diagram data aktual, hasil prediksi klasifikasi SVM sentimen pengguna aplikasi PLN Mobile di media sosial X, serta tabel confusion matrix terhadap data training dan testing. Tampilan halaman ini ada pada Gambar 3.8.

ANALISIS SENTIMEN APLIKASI PLN MOBILE														
	<b>VISUALISASI</b>													
Dashboard	Diagram Hasil Analisis Sentimen	Tabel Confusion Matrix												
Dataset		<table border="1"> <thead> <tr> <th colspan="3">Tabel</th> </tr> </thead> <tbody> <tr> <td>9</td> <td>12</td> <td>8</td> </tr> <tr> <td>11</td> <td>11</td> <td>9</td> </tr> <tr> <td>8</td> <td>9</td> <td>10</td> </tr> </tbody> </table>	Tabel			9	12	8	11	11	9	8	9	10
Tabel														
9		12	8											
11		11	9											
8		9	10											
Klasifikasi SVM														
<b>Visualisasi</b>														
Logout														

**Gambar 3.8** Tampilan Halaman Visualisasi

### 3.6 Skenario Pengujian

Sebuah pengujian dalam pembuatan sistem analisis perlu dilakukan. Pengujian dalam hal ini digunakan untuk menentukan ambang batas akurasi regresi menggunakan metode klasifikasi *Support Vector Machine*. Skenario pengujian dalam penelitian ini menggunakan *Confusion Matrix*. Situasi pengujian dalam analisis sentimen harus dilakukan untuk mendapatkan nilai akurasi yang relevan. Penulis akan menyusun dataset sebanyak 1400 tweet dengan rasio 80 : 20 untuk menguji akurasi yang diberikan oleh model. Adapun rasio pembagian tersebut disajikan pada table di bawah sebagai berikut:

**Tabel 3.14** Rasio Pembagian Dataset

Rasio Pembagian Data Latih : Data Uji	Jumlah Data Latih	Jumlah Data Uji
80:20	1120	280

### 3.7 Evaluasi Model

Evaluasi model adalah proses penting dalam mengevaluasi sistem *machine learning*, yang bertujuan untuk menilai kinerja model terhadap data yang tidak terlihat atau data uji. Evaluasi ini membantu memastikan bahwa model yang telah dilatih mampu bekerja dengan baik dalam situasi nyata, bukan hanya pada data latihannya. Beberapa metrik yang sering digunakan dalam evaluasi model termasuk *accuracy*, *precision*, *recall*, dan *f1-score*, tergantung pada jenis masalah yang

dihadapi, seperti klasifikasi atau regresi data yang diuji. Selama proses evaluasi, data uji yang belum pernah dilihat oleh model digunakan untuk mengukur kemampuannya dalam membuat prediksi. Untuk mengetahui performa dari Algoritma *Support Vector Machine*, maka dilakukan pengujian terhadap model. Hasil klasifikasi akan ditampilkan dalam bentuk *confusion matrix*. Tabel *confusion matrix* terdiri dari kelas predicted dan kelas actual.



