

## Perbandingan Metode *Decision Tree* dan *Naive Bayes Classifier* pada Analisis Sentimen Pengguna Layanan PT Perusahaan Listrik Negara (PLN)

Abiyoga Bagus Mustriyanto\*<sup>1</sup>, Muhammad Habibi<sup>2</sup>, Dayat Subekti<sup>3</sup>, Fajar Syahrudin<sup>4</sup>

<sup>1,2,3</sup>Informatika, FTI UNJAYA, Yogyakarta, Indonesia

<sup>4</sup>Teknologi Informasi, FTI UNJAYA, Yogyakarta, Indonesia

e-mail: \*<sup>1</sup>bagusmustriyanto@gmail.com, <sup>2</sup>muhammadhabibi17@gmail.com, <sup>3</sup>bekti74@gmail.com, <sup>4</sup>fajarsyahrudin08@gmail.com

**Abstract - Background :** PLN is a state-owned company that is tasked with supplying electricity to all regions of Indonesia which certainly cannot be separated from the various obstacles experienced, to find out public sentiment on the services that have been provided, an analysis is carried out to determine public sentiment. The results of these sentiments are created in the dashboard using the Flask framework by comparing the Naive Bayes and Decision tree methods. To create a sentiment analysis dashboard for PT. PLN and make a research analysis model using a comparison of the Naive Bayes Classification and Decision tree methods. The method used in this research is Naive Bayes and Decision tree. The data obtained with a total of 40,745 Tweet data taken in the period 1 May 2022 - 4 June 2022 with the keyword "PLN". Making a dashboard that displays the results of the analysis where there is a menu to display the data and each analysis process. The use of 900 training data and 300 testing data resulted in the Naive Bayes method getting an accuracy of 83% on the training data and 80% for the Testing data, while the Decision tree method got an accuracy of 77% on the Training data and 56% on the Testing data. The analysis obtained for the method in this study also shows that the Naive Bayes method is better for classifying large amounts of data than the Decision tree. The sentiment generated by the highest number is negative, with most of the Tweets being complaints about the response to complaints and handling of damage reported by the public.

**Keywords -** PLN, Sentiment Analysis, Decision tree, Naive Bayes

**Abstrak - Latar Belakang:** PLN merupakan perusahaan milik negara yang bertugas untuk menyuplai pasokan listrik ke seluruh daerah Indonesia yang pastinya tidak lepas dari berbagai kendala yang dialami, untuk mengetahui sentimen masyarakat atas pelayanan yang telah diberikan maka dilakukan analisis untuk mengetahui sentimen masyarakat. Hasil dari sentimen tersebut dibuat di dalam dashboard dengan menggunakan framework Flask dengan membandingkan metode Naive Bayes dan Decision tree.

**Membuat dashboard analisis sentimen terhadap PT. PLN serta membuat model analisis penelitian**

**menggunakan perbandingan metode Naive Bayes Classification dan Decision tree.**

**Metode yang digunakan pada penelitian ini adalah Naive Bayes dan Decision tree. Hasil Penelitian: Data yang di dapat dengan jumlah total 40.745 data Tweet yang diambil pada periode 1 mei 2022 – 4 juni 2022 dengan kata kunci "PLN". Pembuatan sebuah dashboard yang menampilkan hasil dari analisis dimana terdapat menu untuk menampilkan data dan masing-masing proses analisis. Penggunaan data training sebanyak 900 data dan testing sebanyak 300 menghasilkan metode Naive Bayes mendapatkan akurasi sebesar 83% pada data training dan 80% untuk data Testing, sedangkan metode Decision tree mendapatkan akurasi sebesar 77% pada data Training dan 56% pada data Testing." Analisis yang didapatkan untuk metode Pada penelitian ini juga menunjukkan bahwa metode Naive Bayes lebih bagus untuk mengklasifikasikan data dengan jumlah banyak dibandingkan dengan Decision tree. Sentimen yang dihasilkan dengan jumlah terbanyak adalah negatif dengan kebanyakan Tweet adalah keluhan terhadap respon dalam menghadapi komplain dan penanganan kerusakan yang diajukan masyarakat.**

**Kata kunci -** PLN, Analisis Sentimen, Decision tree, Naive Bayes.

### I. PENDAHULUAN

PT. Perusahaan Listrik Negara (PLN) merupakan perusahaan pemerintah yang bertujuan untuk menyelenggarakan usaha penyediaan tenaga listrik bagi kepentingan umum dalam jumlah dan mutu yang memadai serta memupuk keuntungan. Salah satu misi dari PT. PLN adalah menjalankan bisnis kelistrikan dan bidang lain yang terkait, berorientasi pada kepuasan pelanggan [1]. Perusahaan yang sudah mengembangkan sistemnya sampai ke seluruh negeri pastinya tidak lepas dari berbagai kendala dan komplain masyarakat terhadap kinerja yang telah diberikan. Untuk mengetahui kepuasan pelanggan terhadap pelayanan yang diberikan PT.

Twitter merupakan situs web layanan jejaring sosial yang banyak diminati pengguna internet sebagai media komunikasi dan mendapatkan informasi. Informasi yang terdapat pada Twitter berupa pertanyaan, opini atau komentar, baik yang bersifat

positif maupun negatif [2]. Twitter juga menjadi tempat untuk mengutarakan keluhan terkait berbagai layanan, salah satunya keluhan dan komentar terhadap pelayanan PT Perusahaan Listrik Negara PLN. *Tweet* yang biasanya diberikan mengenai pelayanan yang diberikan kurang memuaskan, gangguan terhadap jaringan listrik, lamanya respon petugas terhadap komplain yang diberikan oleh masyarakat, serta terkait isu mengenai kenaikan harga listrik yang akan terjadi pada tahun 2022. Seperti yang terdapat dalam beberapa media masa seperti Kompas.

Analisis sentimen di Twitter mengenai PT. PLN menjadi penting untuk dilakukan untuk menganalisis mengenai opini masyarakat terhadap kinerja perusahaan dan pelayanan yang telah diberikan. Data yang dianalisis berupa *Tweet*, *re-Tweet* serta komentar yang ada, kemudian data akan dibagi menjadi komentar positif, negatif dan netral sehingga didapat hasil sentimen Twitter mengenai pelayanan PT.

Berdasarkan permasalahan yang ada sehingga menjadi bahan penelitian untuk melakukan analisis sentimen yang ada di Twitter mengenai pelayanan PT. PLN, untuk mengetahui tingkat kepuasan pelanggan terkait layanan yang diberikan oleh PT. PLN serta respon masyarakat mengenai kenaikan harga listrik yang akan dilakukan dan menjadi sebuah evaluasi agar lebih baik lagi dalam melakukan pelayanan, nantinya hasil dari analisis ini bisa menjadi bahan pertimbangan untuk menetapkan kebijakan tersebut. Analisis yang dilakukan menggunakan metode *Naive Bayes Classification* dan *Decision tree*. Dua metode ini dipilih karena di beberapa penelitian sebelumnya metode ini merupakan metode dengan hasil akurasi yang sama-sama tinggi. *Naive Bayes Classification* merupakan metode pengklasifikasian *probabilistik* sederhana. Metode ini akan menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari *dataset* yang diberikan. Metode *Naive Bayes* menganggap semua atribut pada setiap kategori tidak memiliki ketergantungan satu sama lain. Keuntungan penggunaan *Naive Bayes Classification* yaitu metode ini hanya memerlukan sejumlah kecil data latih untuk menentukan parameter *mean* dan *varians* dari variabel yang diperlukan untuk klasifikasi. Sedangkan *Decision tree* merupakan sebuah metode klasifikasi yang mudah diinterpretasikan, metode ini terdiri dari tiga tipe dari simpul, yaitu simpul *leaf* yang memuat suatu akhir atau kelas target untuk suatu pohon keputusan, simpul *root* yang merupakan titik awal dari suatu *Decision tree*, dan simpul perantara yang berhubungan dengan suatu pertanyaan atau pengujian [3].

Data yang akan di ambil dari Twitter nantinya akan dikelompokkan berdasarkan komentar negatif dan komentar positif, kemudian akan dilakukan proses pengolahan data dan dilanjutkan untuk memvisualisasikan datanya agar lebih mudah dipahami dan dilihat, sehingga akan menghasilkan analisis sentimen yang sesuai dengan data yang diperoleh dari hasil *Tweet* di Twitter. Proses visualisasi data nantinya akan dibuatkan sebuah *dashboard*

dengan menggunakan *framework Flask*, dengan adanya *dashboard* diharapkan memudahkan dalam melihat hasil dari sentimen yang telah dibuat.

## II. METODE PENELITIAN

Penelitian ini adalah penelitian analisis sentimen positif, negatif dan netral para media sosial Twitter yang menggunakan metode *Naive Bayes Classification* dan *Decision tree*. Dalam penelitian ini membutuhkan data *Tweet* yang diperoleh dari Twitter yang berkaitan dengan PT. PLN, selanjutnya akan dilakukan proses *preprocessing* untuk mendapatkan informasi yang diinginkan. Data yang telah berhasil didapatkan dan diolah nantinya digunakan untuk memetakan informasi sentimen di Twitter mengenai PT. PLN sehingga memperoleh informasi yang sesuai mengenai pelayanan, keluhan dan tanggapan masyarakat mengenai kebijakan serta pelayanan PT. PLN.

### A. BAHAN DAN ALAT PENELITIAN

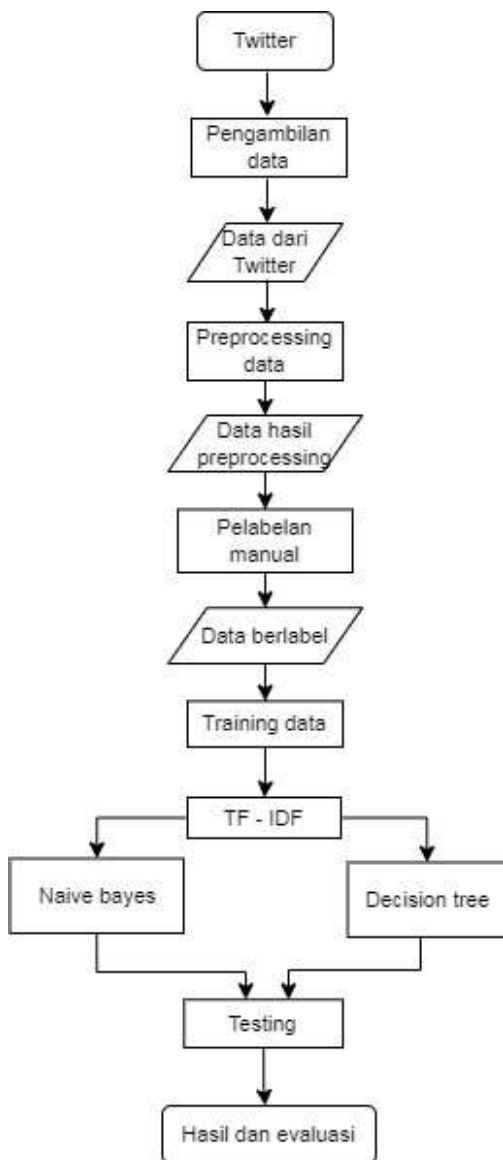
Bahan penelitian yang akan digunakan dalam penelitian ini adalah data dari *Tweet*, *re-Tweet* maupun komentar di Twitter yang berhubungan mengenai pelayanan, keluhan serta tanggapan masyarakat mengenai PT. PLN. Alat mencantumkan piranti-piranti yang dipakai untuk melakukan pengolahan atau pemberian perlakuan terhadap bahan penelitian.

Sistem Operasi dan program-program aplikasi yang dipergunakan dalam pengembangan aplikasi ini adalah:

1. Sistem Operasi: *Windows 7 Ultimate 64-bit*.
2. Bahasa Pemrograman *Python 3.7.3*
3. *Microsoft Office Excel 2016*.
4. *Anaconda 3 64-bit*.
5. *Jupyter Notebook 6.0.1*.
6. *Sublime Text*.

### B. JALAN PENELITIAN

Penelitian ini menggunakan bahasa pemrograman *Python*, *Anaconda 3* dan *Jupyter Notebook* untuk melakukan pengambilan data dari Twitter yang akan ditampilkan di *Microsoft Office Excel* kemudian akan dimodelkan dengan bantuan *library* pada bahasa pemrograman *Python*, jalan penelitian seperti pada Gambar 1.



**Gambar 1.** Jalan Penelitian

Berikut merupakan tahapan- tahapan yang akan dilakukan dalam penelitian ini yaitu :

### 1. Pengambilan Data

Pengambilan data merupakan tahap awal dalam penelitian ini. Dalam tahap ini data diambil dari Twitter mengenai *Tweet* tentang PT. PLN dengan menggunakan *Anaconda Prompt* dengan editor *Jupyter Notebook*, kemudian data akan ditampilkan dengan Microsoft Excel dengan format .csv. Pengambilan data dilakukan mulai dari bulan 1 sampai mei 4 juni 2022 dan didapatkan sekitar 40.745 data. Pengambilan data dilakukan dengan menggunakan API Twitter.



**Gambar 2.** Twitter Developer

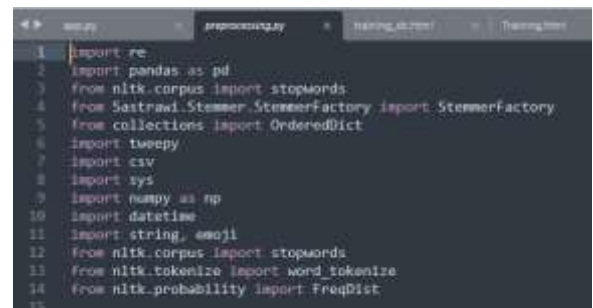
Kemudian dilanjutkan dengan pengambilan data yang dilakukan pada *Jupyter Notebook*. Ada beberapa *library* yang digunakan dalam pengambilan data seperti pada Gambar 3.



**Gambar 3.** Library Pengambilan data

### 2. Preprocessing data

*Preprocessing* data merupakan tahap pengolahan data teks yang baru saja di ambil dengan melakukan tahapan – tahapan untuk memperbaiki data teks yang belum sesuai. *Preprocessing* data bertujuan untuk mempermudah dalam melakukan analisis. *Preprocessing* dilakukan di dalam sistem dengan *framework Flask* dan berikut beberapa *library* yang diperlukan seperti pada Gambar 4.



**Gambar 4.**Library Preprocessing

Berikut merupakan beberapa tahapan dalam proses *preprocessing* :

#### a. Case Folding

*Case folding* merupakan konversi dari bentuk awal menjadi bentuk standar (huruf kecil atau *lowercase*). Berikut merupakan kode yang digunakan untuk melakukan *Case Folding* seperti pada Gambar 5.



**Gambar 5.** Case Folding

b. *Punctuation removal*

*Punctual removal* yaitu penghapusan terhadap karakter khusus yang tidak memiliki nilai dalam melakukan analisis teks. Berikut merupakan kode yang digunakan untuk melakukan *Punctuation removal* seperti pada Gambar 6.

```
def remove_punctuation(data):
    # happy emoticons
    emoticons_happy = re.compile(
        "(?:\ \ |:)\ \ |\ \ \ \ |\ \ \ \ \ |\ \ \ \ \ \ |\ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ \ \ \ |"
    )

    # sad emoticons
    emoticons_sad = re.compile(
        "(?:\ \ |:)\ \ |\ \ \ \ |\ \ \ \ \ |\ \ \ \ \ \ |\ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ \ |\ \ \ \ \ \ \ \ \ \ \ \ \ |"
    )

    # all emoticons (happy + sad)
    emoticons = emoticons_happy + emoticons_sad

    data = re.sub(emoticons, "", data)
    data = re.sub(" ", "", data)
    data = re.sub("\n", "", data)
    data = re.sub("\t", "", data)
    data = re.sub("\r", "", data)
    data = re.sub("\f", "", data)
    data = re.sub("\x0b", "", data)
    data = re.sub("\x0c", "", data)
    data = re.sub("\x0d", "", data)
    data = re.sub("\x0e", "", data)
    data = re.sub("\x0f", "", data)
    data = re.sub("\x10", "", data)
    data = re.sub("\x11", "", data)
    data = re.sub("\x12", "", data)
    data = re.sub("\x13", "", data)
    data = re.sub("\x14", "", data)
    data = re.sub("\x15", "", data)
    data = re.sub("\x16", "", data)
    data = re.sub("\x17", "", data)
    data = re.sub("\x18", "", data)
    data = re.sub("\x19", "", data)
    data = re.sub("\x1a", "", data)
    data = re.sub("\x1b", "", data)
    data = re.sub("\x1c", "", data)
    data = re.sub("\x1d", "", data)
    data = re.sub("\x1e", "", data)
    data = re.sub("\x1f", "", data)
    return data
```

Gambar 6. *Punctuation removal*

c. *Tokenize*

*Tokenize* merupakan proses yaitu melakukan pemecahan atau pemisahan karakter menjadi token atau potongan kata tunggal. Berikut merupakan kode untuk melakukan proses *Tokenize* seperti pada Gambar 7.

```
def tokenize(data):
    data = re.sub("[^a-zA-Z]", " ", data)
    return data
```

Gambar 7. *Tokenize*

d. *Stopword removal*

*Stopword removal* yaitu pemisahan kata –kata penting dari hasil *tokenizing* dan merupakan kata – kata yang akan digunakan untuk mewakili dokumen. Berikut merupakan kode yg digunakan untuk melakukan *stopword* seperti pada Gambar 8.

```
def remove_stopwords(data):
    list_stopwords = [
        "a", "an", "and", "are", "as", "at", "be", "but", "by", "can", "cannot", "could", "do", "does", "for", "from", "has", "have", "he", "her", "his", "in", "is", "it", "me", "my", "no", "not", "of", "on", "or", "over", "she", "so", "that", "the", "to", "was", "we", "were", "with", "you", "your", "youre", "yours"
    ]

    list_stopwords = re.compile("|".join(list_stopwords))
    data = re.sub(list_stopwords, "", data)
    return data
```

Gambar 8. *Stopword removal*

e. *Normalization*

*Normalization* adalah sebuah proses untuk mengembalikan *term* atau kata ke dalam bentuk yang baku, berikut merupakan kode yang digunakan untuk melakukan proses *normalization* seperti pada Gambar 9.

```
def normalize(word):
    word = word.lower()
    word = re.sub("[^a-zA-Z]", " ", word)
    word = re.sub(" ", "", word)
    return word

def normalize_data(data):
    data = [normalize(word) for word in data]
    return data
```

Gambar 9. *Normalization*

f. *Stemming*

*Steming* yaitu pengembalian kata menjadi bentuk dasarnya dengan cara penghilangan infleksinya. Berikut merupakan kode yang digunakan untuk melakukan proses *steming* seperti pada Gambar 10.

```
def stem_word(data):
    data = re.sub(".*", lambda m: m.group(1).stem(), data)
    return data
```

Gambar 10. *Stemming*

3. Pelabelan manual

Pelabelan manual adalah proses memberikan label pada kalimat ataupun kata yang ada pada dokumen sehingga dapat dilakukan analisis kalimat tersebut positif atau negatif, contoh pelabelan seperti pada Gambar 11.



Gambar 11. Pelabelan Manual

4. Training data

*Training data* adalah proses melatih pada data dengan menggunakan metode *Naive Bayes Classification* dan *Decision tree*. Nantinya data yang telah melewati proses latih akan digunakan untuk perhitungan probabilitas dari data berdasarkan data latih sebelumnya. Proses *training data* akan diawali dengan ekstraksi pada teks dengan menggunakan TF-IDF (*Term Frequency-Inverse Document Frequency*), Contoh perhitungan TF-IDF manual dapat dilihat pada Tabel 1.

Tabel 1. Dokumen TF-IDF

Dokumen(d)	Kalimat
d1	padam sekarang padam payah
d2	boleh admin saya sudah dm akun yang ini ya terima kasih
d3	aku harus makan opor dengan suasana candle light dinner
d4	bisa malam takbir begini daerah rumah mati lampu pln cupu
d5	ini gimana sih ini mau lebaran kok tambah mati lampu

Kemudian *Training data* agar dapat membuat model klasifikasi yang dapat digunakan untuk melihat

sentimen dari data secara otomatis seperti pada Gambar 12.

Gambar 12. Training dan Model Klasifikasi

5. Testing

Testing adalah tahapan untuk mengetahui keakuratan dari pemodelan yang sudah dibangun pada tahap training untuk memprediksi label atau kelas dari data uji yang telah dibuat. Model yang telah didapatkan akan dihitung dengan menggunakan metode yang ada dalam confusion matrix untuk mengetahui persentase yang dilakukan pada setiap pengujian. Metode dalam confusion matrix meliputi beberapa tahap, Berikut merupakan kode testing seperti pada Gambar 13.

Gambar 13. Testing

- a. Accuracy didapatkan dengan cara perbandingan antara data yang telah terklasifikasi benar dengan keseluruhan data.
- b. Precision didapatkan dengan cara jumlah data yang diklasifikasi benar dibagi dengan total data yang diklasifikasi positif.

- c. Recall digunakan untuk menunjukkan seberapa besar persentase data yang terkategori positif yang terklasifikasi dengan benar oleh sistem.
- d. F-measure digunakan untuk menghitung rata-rata antara precision dan recall.
- e. Berikut merupakan kode perhitungan confusion matrix yang digunakan seperti pada Gambar 14.

Gambar 14. Confusion Matrix

6. Hasil dan Evaluasi

Hasil dan Evaluasi adalah tahapan terakhir dimana hasil data yang telah diolah akan divisualisasikan menggunakan dashboard dengan framework Flask. Pada tampilan dashboard terdapat fitur untuk melihat hasil perbandingan dari kedua metode, jumlah data, data training dan data testing.

III. HASIL DAN PEMBAHASAN

Hasil dari penelitian penerapan model klasifikasi menggunakan metode Naive Bayes dan Decision tree untuk melakukan analisis terhadap sentimen di media sosial Twitter dengan kata kunci “PLN” menggunakan data yang diambil dari tanggal 1 Mei 2022 sampai dengan 4 Juni 2022 dengan jumlah total data 40.745, baik Tweet maupun re-Tweet. Dalam proses training digunakan sebanyak 900 data dengan 300 data berlabel positif, 300 data berlabel negatif dan 300 data berlabel netral. Sedangkan untuk proses testing menggunakan 300 data dengan rincian 100 data berlabel positif, 100 data berlabel negatif dan 100 data berlabel netral. Hasil dari analisis ini ditampilkan ke dalam sebuah dashboard dengan menggunakan framework Flask. Penelitian ini diharapkan dapat memberikan gambaran mengenai respon masyarakat mengenai pelayanan yang telah diberikan oleh PT. PLN dan juga untuk membandingkan antara metode Naive Bayes dan Decision tree.

A. AKURASI MODEL DECISION TREE

Evaluasi model Decision tree dilakukan dengan menggunakan data dengan jumlah 900 data dengan rincian 300 data dengan label positif, 300 data berlabel negatif dan 300 data berlabel netral, kemudian dilakukan pengujian confusion matrix untuk mendapatkan nilai aktual dan prediksi pada data, hasil dari confusion matrix dapat dilihat pada Tabel 2.

**Tabel 2.** Confusion Matrix Training Decision tree

Kelas Aktual	Kelas Predikasi		
	positif		positif
positif	49	positif	49
negatif	1	negatif	1
netral	3	netral	3

Dari percobaan yang telah dilakukan didapatkan bahwa akurasi dari model yang menggunakan metode *Decision tree* sebesar 77 % kemudian mendapatkan nilai *precision* sebesar 80%, *recall* 77 % dan *f1-score* 77%, seperti pada Gambar 15.

**Gambar 15.** Akurasi Training Decision tree

### B. TESTING DECISION TREE

*Testing* yang dilakukan dengan menggunakan 300 data *testing* dibagi menjadi 100 data netral, 100 data negatif dan 100 data positif kemudian dilakukan penghitungan *confusion matrix* dan menghasilkan data seperti pada Tabel 3.

**Tabel 3.** Confusion Matrix Training Decision tree

Kelas Aktual	Kelas Predikasi		
	positif		positif
positif	8	positif	8
negatif	9	negatif	9
netral	12	netral	12

menghasilkan akurasi sebesar 56 %, *precision* 73 %, *recall* 56% dan *f1score* 60% seperti pada Gambar 16.

**Gambar 16.** Testing Decision tree

### C. AKURASI MODEL NAÏVE BAYES CLASSIFIER

Evaluasi model dilakukan dengan menggunakan data dengan jumlah 900 data dengan rincian 300 data dengan label positif, 300 data berlabel negatif dan 300 data berlabel netral, kemudian dilakukan pengujian *confusion matrix* untuk mendapatkan nilai aktual dan prediksi pada data, hasil dari *confusion matrix* dapat dilihat pada Tabel 4.

**Tabel 4.** Confusion Matrix Training Naive Bayes

Kelas Aktual	Kelas Predikasi		
	positif		positif
positif	56	positif	56
negatif	3	negatif	3
netral	2	netral	2

Dari percobaan kedua yang telah dilakukan didapatkan bahwa akurasi dari model yang menggunakan metode *Naive Bayes* sebesar 83 %, *f1-score* 83 %, *Precision* 83% dan *recall* 83% seperti pada Gambar 17.

**Gambar 17.** Akurasi Training Naive Bayes

### D. TESTING NAÏVE BAYES CLASSIFIER

*Testing* yang dilakukan dengan menggunakan 300 data *testing* dibagi menjadi 100 data netral, 100 data negatif dan 100 data positif dan dilakukan penghitungan *confusion matrix* dan menghasilkan data seperti pada Tabel 5.

**Tabel 5.** Confusion Matrix Testing Naive Bayes

Kelas Aktual	Kelas Predikasi		
	positif		positif
positif	7	positif	7
negatif	0	13	0
netral	2	2	28

Kemudian menghasilkan akurasi sebesar 80 %., *f1-score* 78 %, *precision* 80% dan *recall* 80% seperti pada Gambar 18.

**Gambar 18.** Testing Naive Bayes

### E. PERBANDINGAN HASIL AKURASI DECISION TREE DAN NAÏVE BAYES CLASSIFIER

Berdasarkan analisis yang dilakukan bahwa akurasi metode *Naive Bayes* lebih tinggi dibandingkan dengan akurasi metode *Decision tree* didapatkan bahwa nilai dari *training Naive Bayes* berada angka 83 % sedangkan untuk *Decision tree* berada pada angka 77% dengan jumlah data *training* sebanyak 900 data dan menghasilkan perbandingan seperti pada Tabel 6.

**Tabel 6.** Hasil Akurasi Training

Accuracy	Naive Bayes	Decision tree
		83%

Kemudian untuk data *testing* sebanyak 300 data *Tweet* dengan label positif sebanyak 100 data, negatif 100 data dan netral 100 data sehingga menghasilkan perbandingan akurasi seperti pada Tabel 7.

**Tabel 7.** Hasil Akurasi *Testing*

Accuracy	Naive Bayes	Decision tree
	80%	56%

## F. HASIL KLASIFIKASI TWEET

Klasifikasi dilakukan dengan menggunakan metode dengan akurasi *training* dan *testing* yang memiliki akurasi paling tinggi yaitu menggunakan metode *Naive Bayes*, Seperti pada Gambar 19.

**Gambar 19.** Klasifikasi

Dari hasil klasifikasi dengan menggunakan 40.475 data *Tweet* untuk metode *Naive Bayes* didapatkan jumlah sentimen positif sebanyak 12.029 dengan *Tweet* berisi tentang apresiasi masyarakat mengenai kesiapan dalam menjaga pasokan tenaga listrik, untuk sentimen positif metode *Decision tree* mendapatkan sebanyak 9.189 data *Tweet* dengan pembahasan yang sama, sentimen negatif pada metode *Naive Bayes* mendapat 13.363 dengan pembahasan mengenai lamanya penanganan ketika ada masalah jaringan kelistrikan dan pemadaman listrik secara tiba-tiba sedangkan untuk metode *Decision tree* mendapatkan sebanyak 11.097 data *Tweet* dengan pembahasan yang sama, untuk data netral metode *Naive Bayes* mendapatkan 15.352 data *Tweet* dan metode *Decision tree* mendapat 20.458 data *Tweet* dengan pembahasan mengenai berita dan pertanyaan pelanggan kepada admin PLN. Untuk rincian data sentimen seperti pada Tabel 8.

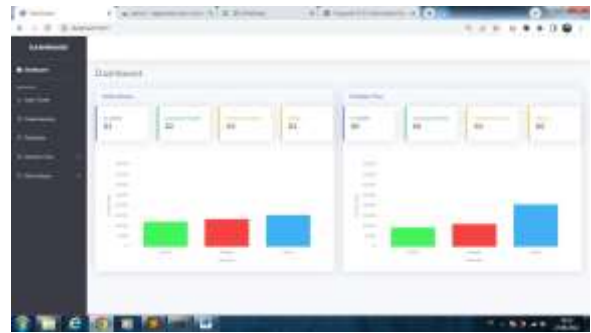
**Tabel 8.** Jumlah Total Sentimen

Sentimen	Naive Bayes	Decision tree
Positif	13.403	9.364
Negatif	14.811	16.649
Netral	12.530	14.731

## G. IMPLEMENTASI APLIKASI

### 1. Dashboard

*Dashboard* merupakan tampilan awal dalam sistem dimana di dalam halamannya akan menampilkan grafik perbandingan antara klasifikasi dengan menggunakan metode *Naive Bayes* dan *Decision tree* seperti pada Gambar 20.

**Gambar 20.** Halaman *Dashboard*

### 2. Data *Tweet*

Data *Tweet* merupakan halaman yang berfungsi untuk meng-*upload* data yang akan di *preprocessing* dan menampilkan data dengan format csv. Data akan tampil di dalam tabel. Seperti pada Gambar 21.

**Gambar 21.** Halaman Data *Tweet*

### 3. *Preprocessing*

*Preprocessing* merupakan halaman yang berfungsi untuk membersihkan mengubah data *Tweet* menjadi struktur data yang baku sehingga mudah dalam melakukan analisis. Berikut merupakan tampilan dari halaman *Preprocessing* seperti pada Gambar 22.

**Gambar 22.** Halaman *Preprocessing*

### 4. Pelabelan

Pelabelan merupakan halaman yang menampilkan data yang telah melalui proses *preprocessing* dan telah memiliki label sentimen, penambahan label dilakukan dengan cara manual menggunakan Microsoft Excel, data ini nantinya akan digunakan sebagai data *training* dalam *tesing* dan klasifikasi nantinya. Seperti pada Gambar 23.



Gambar 23. Halaman Pelabelan Manual



Gambar 26. Halaman Klasifikasi

## 5. Training

*Training* merupakan halaman yang berfungsi untuk melatih data yang akan digunakan untuk melakukan klasifikasi, pada halaman *training* juga dilakukan proses  $tf - idf$ . Proses training dilakukan dengan menggunakan dua metode, yaitu metode *Naive Bayes* dan *Decision tree*. Dapat dilihat pada Gambar 24.



Gambar 24. Halaman Training

## 6. Testing

Halaman *testing* merupakan halaman uji coba untuk model yang telah dibuat, dalam halaman *testing* digunakan sebanyak 300 data *testing*, seperti pada Gambar 25.



Gambar 25. Halaman Testing

## 7. Klasifikasi

Halaman klasifikasi merupakan halaman yang digunakan untuk melakukan proses pelabelan secara otomatis dengan menggunakan data *training* yang telah dilakukan sebelumnya. Dalam halaman klasifikasi menggunakan semua data yang telah diambil dengan jumlah total datanya 40.745. Kemudian menampilkan hasil akurasinya seperti pada Gambar 26.

## IV. KESIMPULAN

Berdasarkan dari penelitian yang dilakukan mengenai analisis sentimen di Twitter dengan studi kasus PLN dapat diambil beberapa kesimpulan :

Penelitian ini berhasil melakukan pengujian untuk dua metode dengan hasil metode *Naive Bayes* mendapatkan akurasi *training* sebesar 83 % sedangkan untuk metode *Decision tree* mandatkan akurasi sebesar 77 % . Kemudian untuk testing metode *Naive Bayes* mendapatkan nilai 80% sedangkan *Decision tree* mendapatkan akurasi sebesar 56%. Penelitian ini juga menunjukkan bahwa Metode *Naive Bayes* lebih bagus untuk mengklasifikasi data dengan jumlah banyak dibandingkan dengan metode *Decision tree*.

Penelitian ini berhasil membuat sebuah *dashboard* tampilan dari hasil analisis sentimen dan melakukan beberapa proses di dalam sistem dengan menggunakan *framework Flask*.

Penelitian ini memberikan hasil dari kedua metode bahwa lebih banyak sentimen negatif daripada sentimen positif. Terdapat banyak keluhan masyarakat mengenai pelayanan yang dilakukan PT. PLN baik dari segi respon dalam menghadapi komplain dan penanganan terhadap kerusakan yang di adukan oleh masyarakat.

## DAFTAR PUSTAKA

- [1] M. S. Alrajak, I. Ernawati, dan I. Nurlaili, "Analisis Sentimen Terhadap Pelayanan PT. PLN Di Jakarta Pada Twitter Dengan Algoritma K-Nearest Neighbor (K-NN)," dipresentasikan pada Prosiding Seminar Nasional Mahasiswa Bidang Ilmu Komputer dan Aplikasinya, 2020, hlm. 110–122.
- [2] W. E. Nurjanah, R. S. Perdana, dan M. A. Fauzi, "Analisis Sentimen Terhadap Tayangan Televisi Berdasarkan Opini Masyarakat pada Media Sosial Twitter menggunakan Metode K-Nearest Neighbor dan Pembobotan Jumlah Retweet," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 1, no. 12, hlm. 1750–1757, 2017.
- [3] R. Puspita dan A. Widodo, "Perbandingan Metode KNN, Decision Tree, dan Naive Bayes Terhadap Analisis Sentimen Pengguna Layanan BPJS," *J. Inform. Univ. Pamulang*, vol. 5, no. 4, hlm. 646, 2021.
- [4] M. Putra dan M. Putera, "Analisis Perbandingan metode SOAP dan REST yang digunakan pada Framework Flask untuk membangun Web Service," *Scan: Jurnal Teknologi Informasi dan Komunikasi*, vol. 14, no. 2, hlm. 1–7, 2019.



- 
- [5] D. N. Batubara, A. P. Windarto, dan E. Irawan, "Analisis Prediksi Keterlambatan Pembayaran Listrik Menggunakan Komparasi Metode Klasifikasi Decision Tree dan Support Vector Machine," *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 1, hlm. 102–108, 2022.
- [6] F. F. Rachman dan S. Pramana, "Analisis sentimen pro dan kontra masyarakat Indonesia tentang vaksin COVID-19 pada media sosial Twitter," *Indonesian of Health Information Management Journal (INOHIM)*, vol. 8, no. 2, hlm. 100–109, 2020.
- [7] N. Ruhyana, "Analisis Sentimen Terhadap Penerapan Sistem Plat Nomor Ganjil/Genap Pada Twitter Dengan Metode Klasifikasi Naive Bayes," *IKRA-ITH Informatika: Jurnal Komputer dan Informatika*, vol. 3, no. 1, hlm. 94–99, 2019.
- [8] F. Ratnawati, "Implementasi Algoritma Naive Bayes Terhadap Analisis Sentimen Opini Film Pada Twitter," *INOVTEK Polbeng-Seri Informatika*, vol. 3, no. 1, hlm. 50–59, 2018.
- [9] D. Darwis, N. Siskawati, dan Z. Abidin, "Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Review Data Twitter Bmkg Nasional," *Jurnal Tekno Kompak*, vol. 15, no. 1, hlm. 131–145, 2021.
- [10] B. P. Pratiwi, A. S. Handayani, dan S. Sarjana, "Pengkukuran Kinerja Sistem Kualitas Udara Dengan Teknologi Wsn Menggunakan Confusion Matrix," *Jurnal Informatika Upgris*, vol. 6, no. 2, 2020.