

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Hasil penelitian ini berupa data mentah *tweet* sebesar 7.861 data. Pada langkah berikutnya yaitu *data preprocessing* dimana data mentah mengalami penyusutan dari 7.861 data menjadi 6.954 data. Setelah itu dilakukan *labeling* secara otomatis dengan Vader Lexicon. Data yang sudah dilabeli akan masuk ke tahap *split data* dimana pembagian data adalah 80% dan 20% yang menghasilkan 5.563 data merupakan data latih dan 1.391 data merupakan data uji. Setelah itu data diklasifikasikan dengan algoritma SVM dan mendapatkan akurasi sebesar 91%. Setelah itu dilakukan evaluasi sehingga mendapatkan *recall 87%*, *accuracy 90%*, *precision 82%*, dan *f1-score 85%*.

4.2 IMPLEMENTASI DESAIN ANTARMUKA

Setelah merancang sebuah desain tampilan menggunakan *tool* Figma langkah selanjutnya adalah melakukan implementasi ke dalam sebuah tampilan *website*. Ada lima tampilan dalam pembangunan desain antarmuka ini yaitu implementasi desain halaman *scrapping*, implementasi desain halaman *preprocessing*, implementasi desain halaman *labeling*, implementasi desain halaman klasifikasi dan yang terakhir adalah implementasi desain halaman evaluasi.

4.2.1 Implementasi Fitur *Scrapping*

Halaman pertama dari *website* ini adalah halaman yang dipergunakan untuk *scrapping data* pada Twitter yang menggunakan Snsrape. Halaman ini memiliki sebuah *sidebar* dengan beberapa fitur seperti *Scrapping*, *Preprocessing*, *Labeling*, Visualisasi SVM, dan Evaluasi. Disamping fitur utama terdapat fitur tambahan seperti *download* data dengan format CSV dan Excel. Data *tweet* yang diambil meliputi *id tweet*, *datetime*, dan *text*. Selanjutnya hasil dari *scrapping* akan

ditampilkan pada halaman tersebut. Implementasi ini dapat dilihat pada gambar dibawah ini yaitu Gambar 4.1.

Gambar 4.1 Implementasi fitur *scraping*

ada Gambar 4.1 implementasi halaman *scraping* dengan kata kunci “jokowi 3 periode” yang dimulai dari tanggal 1 September 2022 sampai 31 Desember 2022. Data *tweet* yang berhasil didapatkan sebesar 7.861 data yang merupakan data yang masih mentah atau belum siap olah.

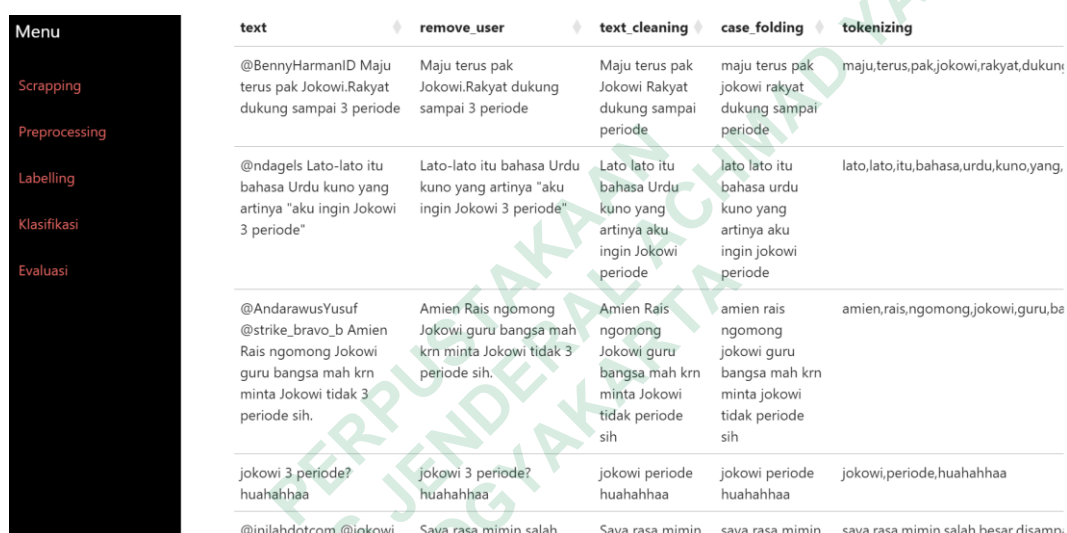
4.2.2 Implementasi Fitur *Preprocessing*

Setelah melakukan *scraping* data *tweet* berhasil diambil dilanjutkan dengan implementasi tampilan untuk data *preprocessing*. Data hasil *scraping* akan di panggil kembali dan memencet tombol proses *text* untuk memproses data.

Gambar 4.2 Implementasi fitur preprocessing bagian hasil *scraping*

Pada Gambar 4.2 tampilan menampilkan hasil *scrapping* yang ditampilkan dengan 10 data perhalaman. Data hasil *scrapping* pada halaman ini juga bisa diunduh dengan format CSV dan Excel.

Setelah mengklik tombol proses *text* maka data akan diproses dalam proses pembersihan data. Dalam pemrosesannya dibutuhkan waktu sekitar 30 menit dan untuk menampilkan data yang sudah tersimpan dari basis data hanya membutuhkan waktu 1 menit tergantung kecepatan internet yang dipakai. Tampilan tersebut dapat dilihat pada Gambar 4.3.



text	remove_user	text_cleaning	case_folding	tokenizing
@BennyHarmanID Maju terus pak Jokowi.Rakyat dukung sampai 3 periode	Maju terus pak Jokowi.Rakyat dukung sampai 3 periode	Maju terus pak Jokowi Rakyat dukung sampai periode	maju terus pak jokowi rakyat dukung sampai periode	maju,terus,pak,jokowi,rakyat,dukun,
@ndagels Lato-lato itu bahasa Urdu kuno yang artinya "aku ingin Jokowi 3 periode"	Lato-lato itu bahasa Urdu kuno yang artinya "aku ingin Jokowi 3 periode"	Lato lato itu bahasa Urdu kuno yang artinya aku ingin Jokowi periode	lato lato itu bahasa urdu kuno yang artinya aku ingin jokowi periode	lato,lato,itu,bahasa,urdu,kuno,yang,
@AndarawusYusuf @strike_bravo_b Amien Rais ngomong Jokowi guru bangsa mah krn minta Jokowi tidak 3 periode sih.	Amien Rais ngomong Jokowi guru bangsa mah krn minta Jokowi tidak 3 periode sih.	Amien Rais ngomong Jokowi guru bangsa mah krn minta Jokowi tidak periode sih	amien rais ngomong jokowi guru bangsa mah krn minta jokowi tidak periode sih	amien,rais,ngomong,jokowi,guru,ba
jokowi 3 periode? huahahaha	jokowi 3 periode? huahahaha	jokowi periode huahahaha	jokowi periode huahahaha	jokowi,periode,huahahaha
@inilahdotcom @jokowi	Saya rasa mimin salah	Saya rasa mimin	saya rasa mimin	saya,rasa,mimin,salah,besar,disamp.

Gambar 4.3 Implementasi fitur *preprocessing* bagian hasil *preprocessing*

Pada Gambar 4.3 merupakan implementasi dari proses *text* pada halaman *preprocessing*. Data *scrapping* akan ditampilkan pada halaman *preprocessing*. Setelah mengklik tombol proses *text* hasil dari *preprocessing* akan ditampilkan. Hasil setelah data diproses dari 7.861 data menjadi 6.954 data. Pada proses ini juga terjadi penambahan kolom seperti *text cleaning*, *remove user*, *case folding*, *tokenizing*, *stop forward*, dan *stemming*. Data hasil *preprocessing* dapat diunduh dengan format CSV dan Excel.

4.2.3 Implementasi Fitur *Labeling*

Setelah implementasi tampilan *preprocessing* dilanjutkan dengan implementasi halaman untuk *labeling*. pada bagian ini data dari hasil *preprocessing* data akan dipanggil pada halaman ini. Proses *labeling* ini memakan waktu sekitar

10 menit dengan data 6.954 data dan untuk menampilkan data *labeling* yang sudah tersimpan di basis data hanya membutuhkan waktu 1 menit tergantung kecepatan internet yang dipakai. Tampilan dapat dilihat pada Gambar 4.4.

The screenshot shows a web application interface for data labeling. On the left is a dark sidebar menu with the following items: Menu, Scraping, Preprocessing, Labeling, Klasifikasi, and Evaluasi. The main content area has a 'Proses text' button at the top. Below it are 'Download csv' and 'Download excel' buttons. The title is 'Hasil process twitter'. There is a 'Show 10 entries' dropdown and a search bar. The table below has the following columns: id, tweet_id, datetime, username, text, and remove_user. The first row of data is as follows:

id	tweet_id	datetime	username	text	remove_user	
0	1	1609325627211145218	31/12/2022	combro_dingin	@syahdansaladin @TarunaAdjiet @YanHarahap @SBVudhoyono @jokowi Apa'a yg stabil...?? Janji 10.000...gaktau'a Pertalite tembus 10.000... 2 periode ..minyak goreng ilang...solar ilang...Sinyal Tv ilang.... 3 periode Sinyal HP lu ilang... mikiirrrrr....pake otakmu biar gakliatan DUNGU..... ilang...Sinyal Tv ilang.... 3	

Gambar 4.4 Implementasi fitur *labeling* bagian data *preprocessing*

Pada Gambar 4.4 merupakan tampilan dari data hasil *preprocessing data* yang ditampilkan kembali pada halaman *labeling*. Setelah dipanggil lalu mengklik proses *text* untuk segera dilakukan proses *labeling*. Hasil dari data yang sudah berlabel bisa dilihat dari Gambar 4.5.

The screenshot shows a web application interface for data labeling. On the left is a dark sidebar menu with the following items: Menu, Scraping, Preprocessing, Labeling, Klasifikasi, and Evaluasi. The main content area shows a table with the following columns: stemming, compound, score, and sentimen. The table contains four rows of data:

stemming	compound	score	sentimen
maju jokowi rakyat dukung periode	0.8750	{'neg': 0.0, 'neu': 0.25, 'pos': 0.75, 'compound': 0.875}	positif
lato lato bahasa urdu kuno jokowi periode	-0.2500	{'neg': 0.25, 'neu': 0.75, 'pos': 0.0, 'compound': -0.25}	negatif
vi,periode amien rais ngomong jokowi guru bangsa mah jokowi periode	0.0000	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}	positif
jokowi periode	0.0000	{'neg': 0.0, 'neu': 1.0,	positif

Gambar 4.5 Implementasi fitur *labeling* bagian hasil *labeling*

Pada Gambar 4.5 setelah mengklik proses *text* maka akan muncul hasil dari *labeling* otomatis dengan Vader Lexicon dengan sentimen positif dan sentimen negatif. Vader Lexicon sendiri mempunyai empat sentimen yaitu positif, netral, negatif dan *compound* yang merupakan nilai metrik skor gabungan yang telah dinormalisasi. Pada penelitian ini sentimen yang digunakan hanya sentimen positif dan negatif karena penelitian berfokus pada hasil tetap tidak bercampur dengan sentimen netral agar dapat mengetahui seberapa orang yang mendukung isu tersebut secara jelas dan seberapa orang yang tidak mendukung isu tersebut secara jelas. Jika netral ditambahkan maka tidak akan menemukan jawaban dari seberapa banyak orang yang mendukung dan tidak mendukung secara jelas karena sentimen netral bersifat bisa iya dan bisa tidak. Pada proses ini ditambahkan sebuah kolom yaitu *compound*, *score*, dan sentimen.

4.2.4 Implementasi Fitur Klasifikasi

Setelah dilakukan implementasi tampilan pada halaman *labeling* maka langkah selanjutnya adalah implementasi tampilan pada halaman klasifikasi. Proses klasifikasi menggunakan metode SVM ini waktu 4 menit tergantung kecepatan internet yang dipakai. Data *labeling* akan dipanggil lagi untuk ditampilkan pada halaman klasifikasi. Tampilan dapat dilihat pada Gambar 4.6.

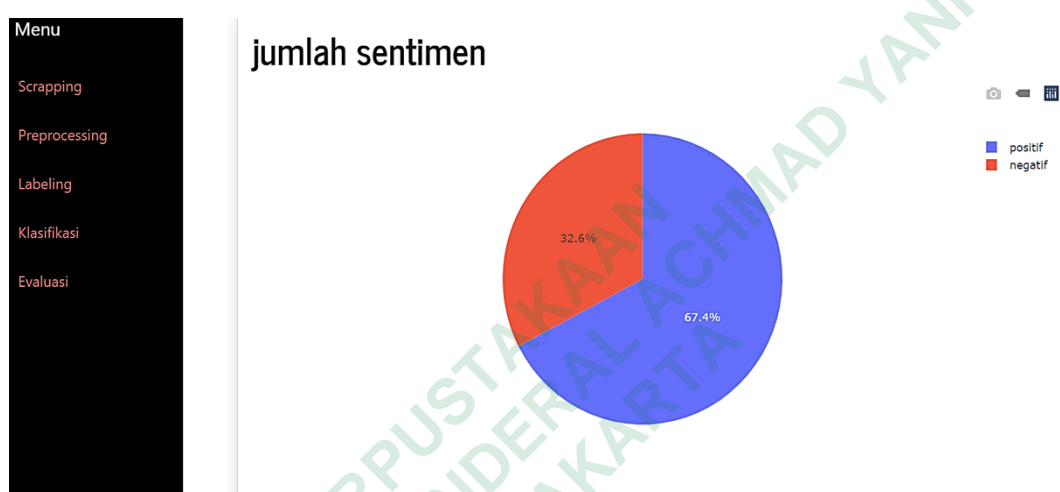
The screenshot displays a web application interface. On the left is a dark sidebar menu with the following items: Menu, Scapping, Preprocessing, Labelling, Klasifikasi, and Evaluasi. The main content area is titled 'Pelabelan dataset' and features two red buttons for 'Download csv' and 'Download excel'. Below the title, there is a 'Show 10 entries' dropdown and a search bar. A table with the following columns is shown: id, tweet_id, datetime, username, text, and remove_user. The table contains one row of data for a tweet from user 'combro_dingin' on 2022-12-31. The text of the tweet is a long, repetitive sentence about economic conditions and prices.

id	tweet_id	datetime	username	text	remove_user	
0	1	1609325627211145218	2022-12-31	combro_dingin	@syahdansaladin @TarunaAdjie1 @YanHarahap @SBYudhoyono @jokowi Apa'a yg stabil...?? Janji 1 Dollar 10.000...gaktau'a Pertalite tembus 10.000... 2 periode ..minyak goreng ilang...solar ilang...Sinyal Tv ilang... 3 periode Sinyal HP lu ilang... mikiirrrrr...pake otakmu biar gakliatan DUNGU.....	Apa'a yg stabil...?? Janji 1 Dollar 10.000...gaktau'a Pertalite tembus 10.000... 2 periode ..minyak goreng ilang...solar ilang...Sinyal Tv ilang... 3 periode Sinyal HP lu ilang... mikiirrrrr...pake otakmu biar gakliatan DUNGU.....

Gambar 4.6 Implementasi fitur klasifikasi bagian hasil *labeling*

Pada Gambar 4.6 tampilan untuk menampilkan data *labeling* sama seperti tampilan saat menampilkan sebuah data. Data tersebut nantinya akan diproses secara otomatis tanpa harus melakukan klik pada sebuah tombol.

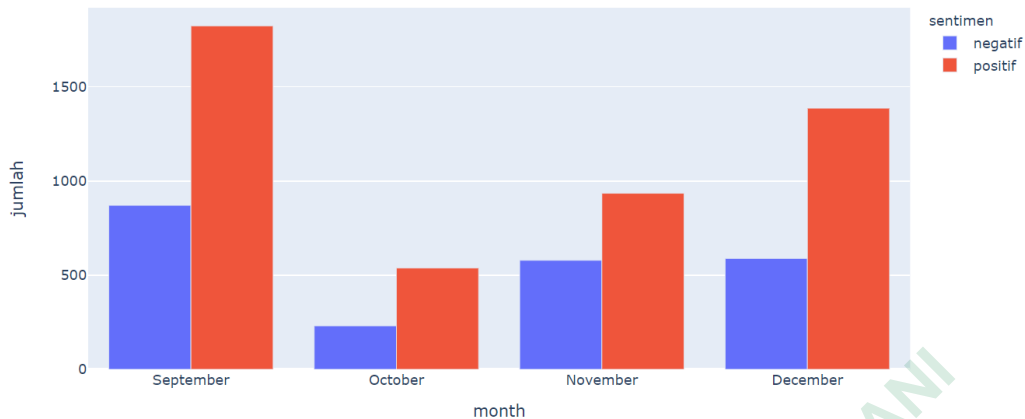
Pada bagian bawah data *labeling* ditampilkan sebuah *pie chart* untuk memudahkan dalam mengetahui seberapa banyak sentimen negatif dan seberapa banyak sentimen positif pada data tersebut. Tampilan dapat dilihat pada Gambar 4.7.



Gambar 4.7 Implementasi fitur klasifikasi bagian *chart labeling*

Pada Gambar 4.7 pada tampilan *pie chart* ditampilkan hasil pelabelan diatas diketahui bahwa 4.684 data memiliki data sentimen positif dan 2.270 data memiliki data sentimen negatif. Pada *pie chart* diatas juga dapat mengetahui berapa persen sentimen negatif dan berapa persen sentimen positif pada data tersebut. Sentimen positif memiliki persen sebesar 67.4% dan sentimen negatif memiliki persen sebesar 32.6%.

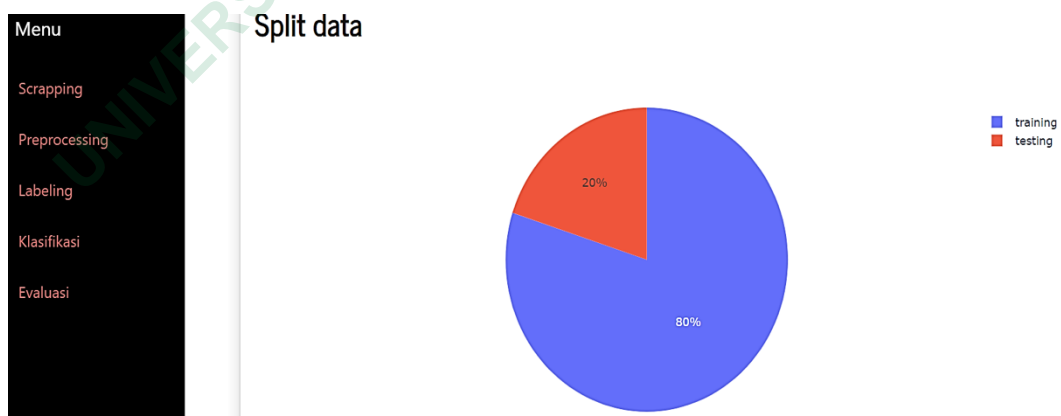
Selanjutnya agar dapat mengetahui lebih detail mengenai persebaran data yang memiliki sentimen positif dan sentimen negatif dibuat sebuah *chart* dengan dicantumkan data dari bulan September sampai Desember. *Chart* tersebut dapat menampilkan sentimen positif dan sentimen negatif dalam satu bulan. Tampilan dari *chart* ini dapat dilihat pada Gambar 4.8.



Gambar 4.8 Halaman klasifikasi bagian *chart* per bulan

Pada Gambar 4.8 ditampilkan sebuah *chart* dengan hasil bulan September sebanyak 1.824 data merupakan sentimen positif dan 871 data merupakan sentimen negatif. Selanjutnya pada bulan Oktober sebanyak 538 data merupakan sentimen positif dan 231 data merupakan sentimen negatif. Pada bulan November sebanyak 935 data merupakan sentimen positif dan 579 data merupakan sentimen negatif. Terakhir adalah bulan Desember sebanyak 1.387 data merupakan sentimen positif dan 589 data merupakan sentimen negatif.

Setelah mengetahui seberapa banyak sentimen positif dan negatif pada tampilan diatas langkah selanjutnya adalah melakukan *split data* dengan data yang dipisah yaitu data latih dan data uji. Tampilan dapat dilihat pada Gambar 4.9.



Gambar 4.9 *Split data*

Pada Gambar 4.9 dilakukan *split data* yang membagi data menjadi data latih dan data uji menjadi 80% dan 20%. Jumlah dari masing-masing data adalah 5.563

data sebagai data latih dan 1.391 data sebagai data uji. Data latih ini nantinya akan digunakan untuk pemodelan metode SVM dan data uji digunakan untuk perhitungan *confusion matrix*.

Setelah data dipisah selanjutnya adalah menghitung TF-IDF yang diperlukan untuk mengetahui sebuah istilah sangat penting dalam sebuah dokumen. Tampilan tersebut dapat dilihat pada Gambar 4.10.

words	tfidf
9240 zolimi	0.509620
9241 zon	0.509620
9242 zona	0.509620
9243 zonauang	0.509620
9244 zonk	0.509620

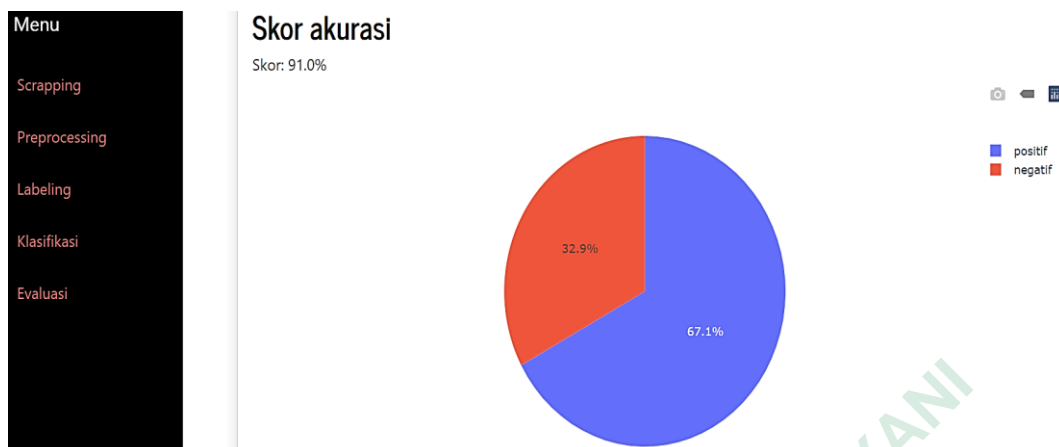
Gambar 4.10 TF-IDF

Pada Gambar 4.10 merupakan hasil dari perhitungan pembobotan TF-IDF data secara keseluruhan. Perhitungan TF-IDF diperlukan untuk menentukan sebuah nilai frekuensi pada sebuah kata pada dokumen. Setelah itu dilanjutkan dengan pemodelan SVM. Pada bagian ini dengan ditampilkan sebuah data bersentimen positif dan negatif yang dapat dilihat pada Gambar 4.11.

Sentimen	jumlah
Negatif	1830
Positif	3733

Gambar 4.11 Sentimen positif dan sentimen negatif

Pada data latih sebanyak 3.733 merupakan sentimen positif dan 1.830 data merupakan sentimen negatif. Selanjutnya adalah membuat tampilan halaman untuk memvisualisasikan hasil dari pemodelan tersebut dengan *pie chart* yang ditunjukkan pada Gambar 4.12.



Gambar 4.12 Halaman klasifikasi bagian visualisasi SVM

Gambar 4.12 merupakan representasi *pie chart* dari hasil pemodelan SVM dengan menggunakan data latih. Pada data latih sebanyak 3.733 adalah positif dan 1.830 data merupakan negatif. Berdasarkan hasil pemodelan SVM ini dihasilkan akurasi sebesar 91% dari yang semula sebesar 89%. Hasil akurasi ini didapat setelah penambahan pada *stopword list* dimana kata tersebut tidak ada pada *NLTK corpus* yang dilihat pada Gambar 4.13.

```
# Menambahkan stopwords tambahan yang umum dalam teks bahasa Indonesia
LIST_STOPWORDS.extend(['amp', 'biar', 'sih',
                       'si', 'n', 't', 'nyg', 'hehe', 'pen', 'u', 'nan', 'loh', 'rt',
                       'yah', 'no', 'je', 'om', 'pru', 'sch', 'injirrr', 'ah',
                       'oena', 'bu', 'eh', 'xac', 'anjir'])

LIST_STOPWORDS = set(LIST_STOPWORDS)
```

Gambar 4.13 Tambahan *Stopword List*

4.2.5 Implementasi Fitur Evaluasi

Tahap terakhir adalah tampilan untuk implementasi halaman yang dipergunakan untuk pemrosesan Evaluasi. Pada halaman ini dibuat sebuah *chart* yang digunakan untuk memvisualisasikan hasil dari perhitungan *confussion matrix*. Dalam perhitungan terdapat empat perhitungan yaitu *recall*, *Precision*, *accuracy*, dan *f1-score*. Perhitungan presentase tersebut dapat dilihat pada Persamaan (7), Persamaan (8), Persamaan (9), dan Persamaan (10) sebagai berikut.

1. Perhitungan *recall*

$$\text{Recall} = \frac{365}{415} = 0,87$$

2. Perhitungan *precision*

$$Precision = \frac{900}{1088} = 0,82$$

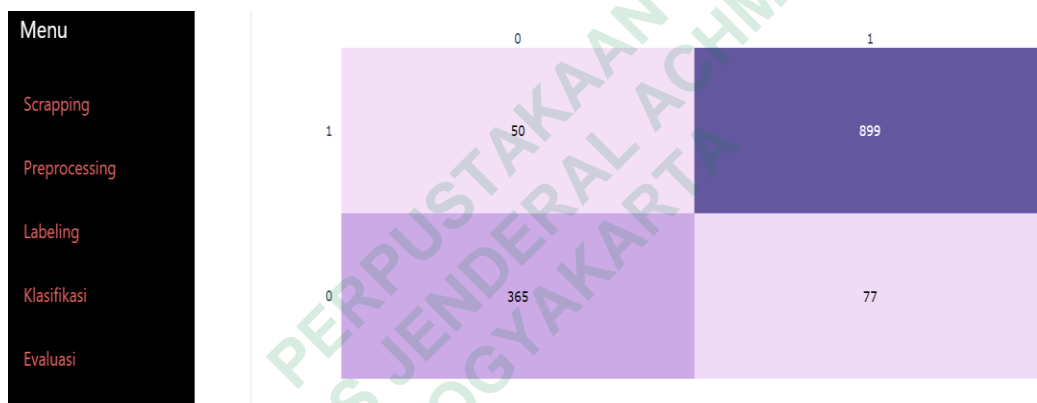
3. Perhitungan *accuracy*

$$Accuracy = \frac{1153}{1391} = 0,90$$

4. Perhitungan *f1-score*

$$F1-score = \frac{1,5416}{1,76} = 0,85$$

Pada tahap ini memerlukan proses waktu kurang lebih 3 menit sesuai dengan kecepatan internet yang dipakai untuk melakukan pemrosesan dalam *confussion matrix* yang dapat dilihat pada Gambar 4.14.



Gambar 4.14 Halaman implementasi *Confussion Matrix*

Pada Gambar 4.13 merupakan tampilan dari *chart* dari perhitungan *confussion matrix* yang berisi perhitungan *recall*, *precision*, *accuracy*, dan *f1-score*. Data yang digunakan untuk perhitungan *confussion matrix* diambil dari data uji. Hasil yang diperoleh adalah *accuracy* sebesar 90 %, *precision* sebesar 82%, *recall* sebesar 87 % dan *f-1 score* sebesar 85 %.

4.3 BASIS DATA

Dalam penelitian ini basis data yang digunakan untuk menyimpan data memanfaatkan SQLite Database. Dengan pemodelan dilakukan pada model.py. terdapat tiga tabel pada basis data ini. Tabel pertama adalah tabel *tweet* yang dapat dilihat pada Gambar 4.15 serta kode program sebagai berikut.

```
class Tweet(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    datetime = db.Column(db.String(200))
    tweet_id = db.Column(db.String(200))
    username = db.Column(db.String(200))
    text = db.Column(db.String(2000))
```

id	datetime	tweet_id	username	text
1	31/12/2022	1609325627211145218	combro_dingin	@syahdansaladin @T...
2	31/12/2022	1609322030914244608	ArRayah43832329	@myputun Katanya s...

Gambar 4.15 Representasi tabel *tweet*

Tabel pertama diberi nama *Tweet* yang menyimpan *id*, *datetime*, *tweet id*, *username*, *text* yang diambil dari proses *scrapping*. Selanjutnya merupakan tabel *preprocess* dengan kode program yang dapat dilihat pada Gambar 4.16.

```
class Preprocess(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    datetime = db.Column(db.String(200))
    tweet_id = db.Column(db.String(200))
    username = db.Column(db.String(200))
    text = db.Column(db.String(2000))
    remove_user = db.Column(db.String(2000))
    text_cleaning = db.Column(db.String(2000))
    case_folding = db.Column(db.String(2000))
    tokenizing = db.Column(db.String(2000))
    stop_words = db.Column(db.String(2000))
    stemming = db.Column(db.String(2000))
```

	remove_user	text_cleaning	case_folding	tokenizing	stop_words	stemming
6936	as , bapak jangan...	Mas bapak jangan b...	mas bapak jangan b...	mas,bapak,jangan,b...	mas,maju,periode,m...	mas maju periode h...
6937	ak , kami sayang ...	Pak kami sayang pa...	pak kami sayang pa...	pak,kami,sayang,pa...	sayang,periode	sayang periode
6938	ngomongnya sih spt...	ngomongnya sih spt...	ngomongnya sih spt...	ngomongnya,sih,spt...	ngomongnya,spt,mel...	ngomongnya spt lan...

Gambar 4.16 Representasi tabel *preprocess*

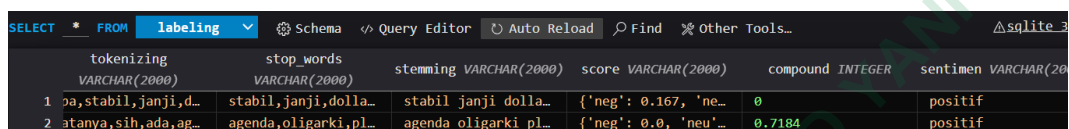
Tabel *preprocess* menyimpan *id*, *datetime*, *tweet id*, *username*, *text* dan tambahan *column* sebanyak lima *column* yaitu *text cleaning*, *remove user*, *case folding*, *tokenizing*, *stop word*, dan *stemming*. Lalu tabel terakhir adalah tabel *labeling* dengan kode program yang dapat dilihat pada bawah serta tampilan tabel *labeling* pada Gambar 4.17.

```
class Labeling(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    datetime = db.Column(db.String(200))
    tweet_id = db.Column(db.String(200))
```

```

username = db.Column(db.String(200))
text = db.Column(db.String(2000))
remove_user = db.Column(db.String(2000))
text_cleaning = db.Column(db.String(2000))
case_folding = db.Column(db.String(2000))
tokenizing = db.Column(db.String(2000))
stop_words = db.Column(db.String(2000))
stemming = db.Column(db.String(2000))
score = db.Column(db.String(2000))
compound = db.Column(db.Integer)
sentimen = db.Column(db.String(2000))

```



	tokenizing VARCHAR(2000)	stop_words VARCHAR(2000)	stemming VARCHAR(2000)	score VARCHAR(2000)	compound INTEGER	sentimen VARCHAR(2000)
1	pa, stabil, janji, d...	stabil, janji, dolla...	stabil janji dolla...	{'neg': 0.167, 'ne...	0	positif
2	atanya, sih, ada, ag...	agenda, oligarki, pl...	agenda oligarki pl...	{'neg': 0.0, 'neu'...	0.7184	positif

Gambar 4.17 Representasi tabel *Labeling*

Tabel *labeling* ini menyimpan *id*, *datetime*, *tweet id*, *username*, *text*, *text cleaning*, *remove user*, *case folding*, *tokenizing*, *stop word*, dan *stemming* dan terdapat penambahan *column* sebanyak tiga *column* yaitu *score*, *compound*, dan *sentimen*.

Tabel 4.15, Tabel 4.16, dan Tabel 4.17 dapat dilihat pada *Visual Studio Code* setelah melakukan *flask migrate* dengan modul *Alembic* dari *Sqlalchemy*. Pastikan *visual studio code* sudah memiliki *extention* *SQLite* agar dapat digunakan untuk menyimpan data yang akan digunakan untuk penelitian.

4.4 PEMBAHASAN

Hasil dari penelitian ini diawali dengan proses *scrapping* melalui *Snsrape* untuk *scrap* data pada *Twitter*. Data yang berhasil diambil sebesar 7.861 data. Selanjutnya dilakukan *preprocessing* dimana data mentah mengalami proses *text cleaning*, *case folding*, *tokenizing*, *stopword removal*, dan *stemming* sehingga data yang sudah bersih menyusut menjadi 6.954 data.

Pada tahap pelabelan data, data yang dilabeli merupakan data yang sudah bersih dan siap untuk dilakukan pelabelan dengan otomatis dengan *Vader Lexicon*. *Vader Lexicon* merupakan sebuah *package* dalam *NLTK* dalam Bahasa pemrograman *Python*. *Corpus* yang digunakan adalah *corpus* *Vader* berbahasa Indonesia *SentiStrength_id* seperti Gambar 4.18.

```
1 { "abadi": 5,  
2   "absen": -3,  
3   "abu-abu": -1,  
4   "acuh": 4,  
5   "adil": 5,  
6   "aduh": -2,
```

Gambar 4.18 File *_JSON_Sentiwords_id*

Pada Gambar 4.17 merupakan contoh kata *corpus* dalam bahasa Indonesia serta nilai dari masing-masing kata untuk menentukan sentimen positif dan sentimen negatif. Data yang sudah dilabeli memiliki sentimen positif sebesar 4.684 data dan data dengan sentimen negatif sebesar 2.270 data.

Setelah data dilabeli maka selanjutnya adalah dilakukannya *split data* dengan pembagian 80% data latih dan 20% data uji. Perolehan data dari pembagian data tersebut menghasilkan data latih sebesar 5.563 data dan data uji sebesar 1.391 data.

Pada pembobotan TF-IDF ini dipergunakan untuk perhitungan algoritma *machine learning* SVM dan evaluasi model. Pemodelan SVM mendapatkan akurasi sebesar 91%. Setelah pemodelan berhasil langkah selanjutnya adalah evaluasi dimana penelitian ini menggunakan *confussion matrix* dan memperoleh nilai *accuracy* sebesar 90%, *precision* sebesar 82%, *recall* sebesar 87% dan *f-1 score* sebesar 85%.