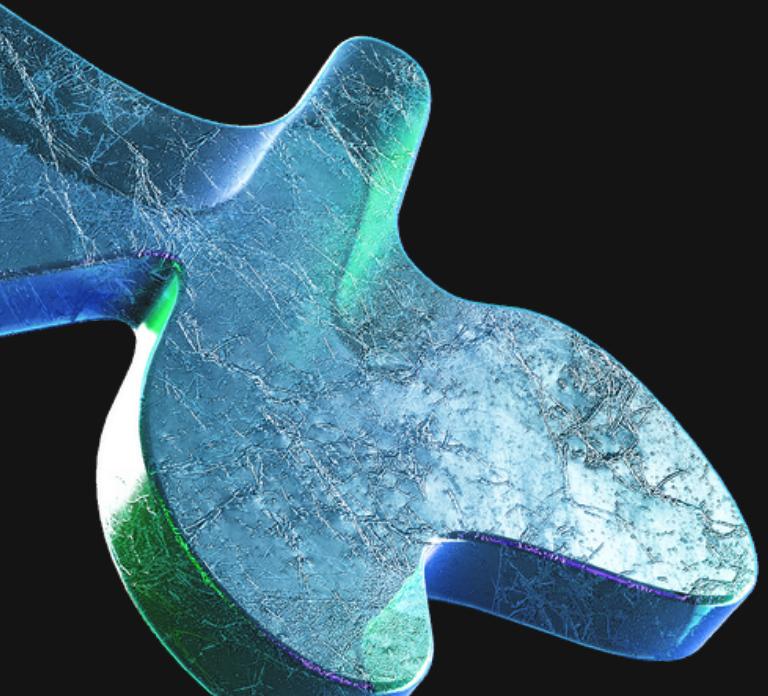


# Data Sentiment Analysis

Kelompok 3  
Bagus Ramadhani  
Rafi Arribaath Alfaresy  
Delifaysa Dyah Prameshti



# Latar Belakang

Dalam era digital saat ini, media sosial telah menjadi platform utama bagi masyarakat untuk berbagi pendapat, menyampaikan informasi, dan berinteraksi dengan sesama pengguna. Salah satu platform media sosial yang paling populer adalah Twitter. Platfrom Twitter digunakan untuk menjadi salah satu media sosial yang digunakan untuk menyampaikan opini. Dengan banyak nya opini tersebut perlu dilakukan analisis sentimen guna memahami suatu topik tersebut memiliki opini positif, negatif atau netral.

Analisis Sentiment adalah teknik yang digunakan untuk mengklasifikasikan teks menjadi beberapa kategori yang mana teknik ini digunakan untuk mengetahui kecendrungan suatu opini atau pendapat masyarakat. Kami melakukan eksperimen dengan 11000 data dari twitter yang mana eksperiment ini guna untuk mengetahui sentimen dari opini masyarakat dan untuk menerapkan teknik model algoritma yang sesuai agar dapat memprediksi sentimen dari opini masyarakat tersebut.



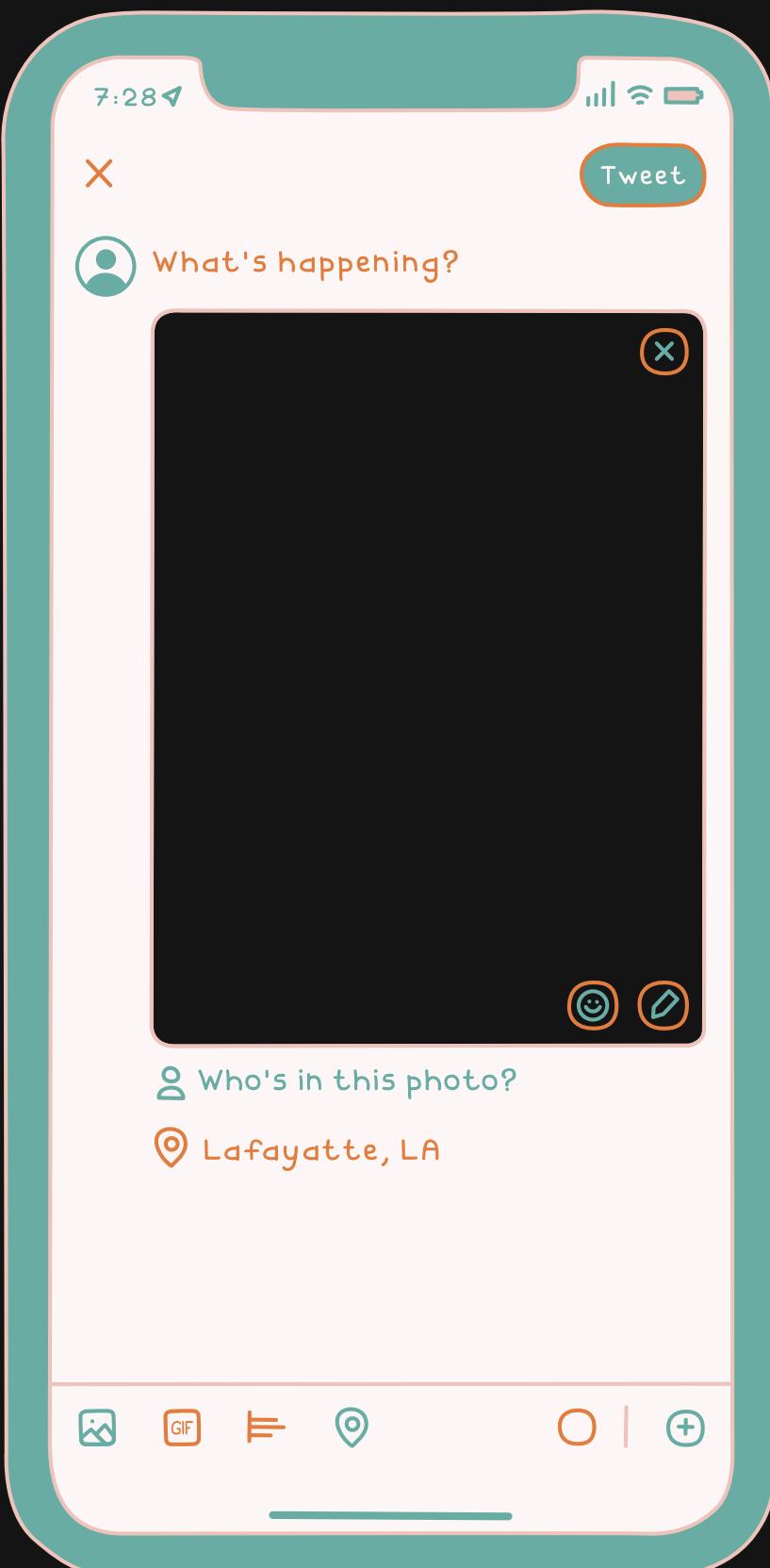
# Rumusan Masalah

---

- Mengetahui Opini pada suatu topik apakah lebih cendrung positif, negatif atau netral
- Melakukan analisis sentiment dengan model Neural Network dan LSTM sehingga dapat menentukan algoritma yang terbaik
- Seperti apa bentuk dari hasil analisis sentiment pada model LSTM dengan menggunakan Tensorflow?
- Membuat suatu model agar dapat bisa dikonsumsi oleh masyarakat umum menggunakan API

# Tujuan Penelitian

---



Tujuan diadakannya penelitian ini untuk memprediksi hasil sentiment yang dominan di gunakan pada pengguna twitter. Dan dapat menentukan model yang terbaik.

Tujuan lainnya untuk memperoleh API yang menghasilkan output dari kategori sentiment, seperti; positif, netral dan negatif.



# Sumber Data

Data yang di gunakan adalah data Binar Platinum Challenge 'Indonesian Abusive and Hate Speech Twitter Text'.

Data tersebut memuat komentar dan opini berupa tweet yang mengandung abusive dan hate speech yang tersebar di internet. Total data yang digunakan sebanyak 11.000 data

train\_preprocess.tsv.txt  
warung ini dimiliki oleh pengusaha pabrik tahu yang sudah puluhan tahun terkenal membuat tahu putih di bandung . tahu berkualitas , dipadu keahlian memasak , dipadu kreativitas , jadilah warung yang menyajikan menu utama berbahan tahu , ditambah menu umum lain seperti ayam . semuanya selera indonesia . harga cukup terjangkau . jangan lewatkan tahu bletoka nya , tidak kalah dengan yang asli dari tegal ! positive  
mohon ulama lurus dan k212 mmbri hujjah partai apa yang harus diwlh agar suara islam tidak pecah-pecah neutral  
lokasi strategis di jalan sumatera bandung . tempat nya nyaman terutama sofa di lantai 2 . paella nya enak , sangat pas dimakan dengan minum bir dingin . appetiser nya juga enak-enak . positive  
betapa bahagia nya diri ini saat unboxing paket dan barang nya bagus ! menetapkan beli lagi ! positive  
duh . jadi mahasiswa jangan sompong dong . kasih kartu kuning segala . belajar dulu yang baik , tidak usahlah ikut-ikut politik . nanti sudah selesai kuliah nya mau ikut politik juga tidak telat . dasar mahasiswa . negative  
makanan beragam , harga makanan di food stall akan ditambahkan 10 % lagi di kasir , suasana ramai dan perlu perhatian untuk mendapatkan parkir dan tempat duduk . positive  
pakai kartu kredit bca tidak untung malah rugi besar negative  
tempat unik , bagus buat foto , makanan enak , pegawai ramah , bersih dan luas , wifi kencang . harga standar dan sesuai dengan tempat nya . ada menu masakan makanan barat dan indonesia . menu favorit lychee mojito dan spaghetti positive  
saya bersama keluarga baru saja menikmati pengalaman kuliner yang menyenangkan di rm sari sunda di jalan . setiabudhi , bandung . tapi , karena tidak ada hubungan untuk rm sari sunda di jalan . setiabudhi , maka saya tuliskan di sini . jadi , maaf jika rekan-rekan pembaca ada yang bingung . beberapa hal yang saya suka dari restoran ini adalah bahwa pelayanan restoran ini terbilang cepat . positive  
bersyukur positive  
simcard indosat inaktiv gara-gara lupa isi pulsa dan kabar nya aktif jika pindah ke pasca bayar , ribet banget negative  
sifat iri sering muncul pada orang-orang yang tidak punya tujuan hidup negative  
sekadar menceritakan pengalaman saya pesan steak grilled beef 200 tidak disajikanlah steak 200 tidak

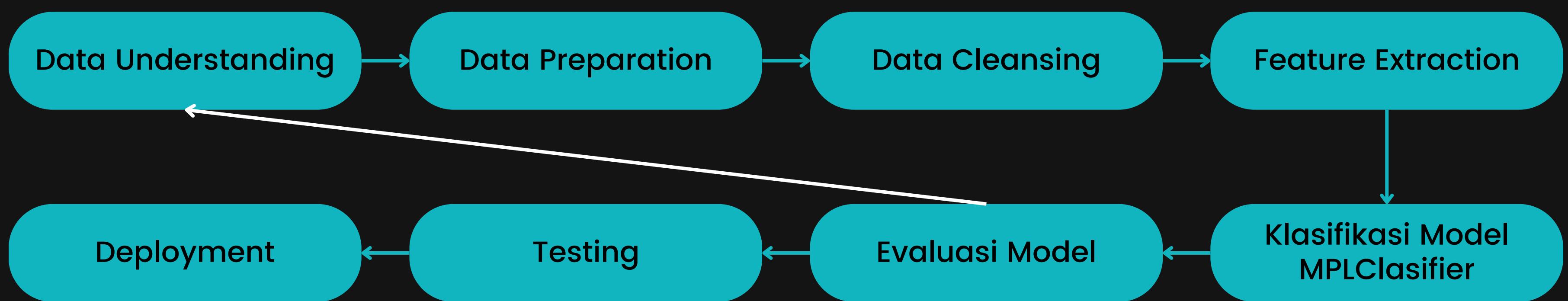
# Metodologi

---

Metode analisis yang dipilih untuk melaksanakan penelitian ini adalah analisis deskriptif. Peneliti memutuskan untuk menggunakan analisis deskriptif karena bertujuan untuk menghasilkan hasil sentimen dari data tweet yang telah melalui proses pengolahan menjadi kategori 'positif', 'netral', dan 'negatif'. Analisis deskriptif dipilih karena fokusnya terutama pada pemahaman mendalam tentang kondisi data yang ada serta pengidentifikasi pola-pola yang mungkin terdapat di dalamnya. Dengan demikian, metode ini diharapkan mampu memberikan wawasan yang lebih mendalam dan terperinci terkait dengan sentimen yang terungkap dalam data tweet yang dianalisis.

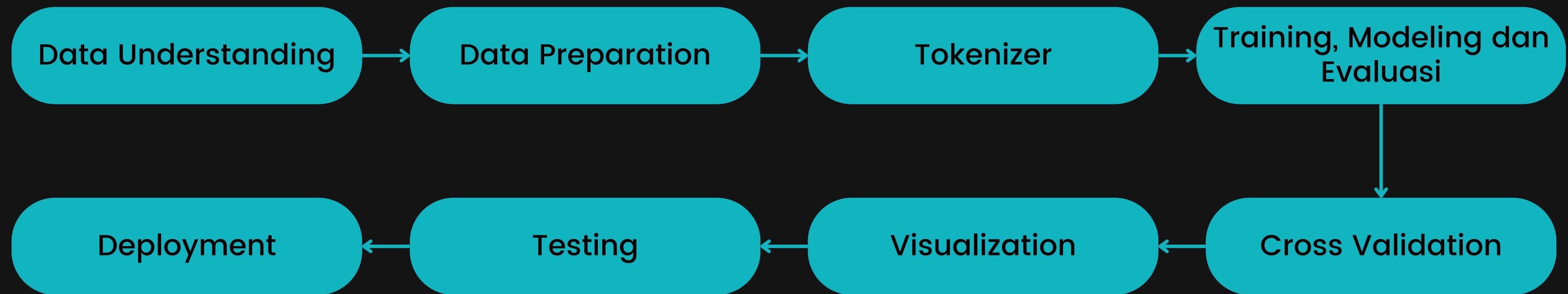
# Langkah Permodelan

## Neural Network



# Langkah Permodelan

LSTM



# Data Understanding

```
#cek data duplicate
df.duplicated().sum()

67

#menghapus data duplikat
df.drop_duplicates(inplace=True)

df.duplicated().sum()

0

df.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 10933 entries, 0 to 10999
Data columns (total 1 columns):
 #   Column   Non-Null Count   Dtype  
 ---  --       --           --      
 0   content   10933 non-null   object 
dtypes: object(1)
memory usage: 170.8+ KB

#menampilkan data teratas
df.head()
```

```
#memberi nama kolom
df.rename(columns={0: "content"}, inplace=True)
df

          content
0    warung ini dimiliki oleh pengusaha pabrik tahu...
1    mohon ulama lurus dan k212 mmbri hujah partai...
2    lokasi strategis di jalan sumatera bandung . t...
3    betapa bahagia nya diri ini saat unboxing pake...
4    duh . jadi mahasiswa jangan sompong dong . kas...
...
10995      tidak kecewa\tpositive\n
10996  enak rasa masakan nya apalagi kepiting yang me...
10997      hormati partai-partai yang telah berkoalisi\ntn...
10998      pagi pagi di tol pasteur sudah macet parah , b...
10999      meskipun sering belanja ke yogya di riau junct...

11000 rows × 1 columns
```

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11000 entries, 0 to 10999
Data columns (total 1 columns):
 #   Column   Non-Null Count   Dtype  
 ---  --       --           --      
 0   content   11000 non-null   object 
dtypes: object(1)
memory usage: 86.1+ KB
```

- Memeriksa apakah terdapat duplikasi data, dan menghapus duplikasi data
- Data memiliki 10933 baris dan juga 2 kolom

# Cleansing dan Feature Extraction

## Neural Network & LSTM

```
#Cleaning
import re
import string
import nltk
#remove URL
def remove_URL(tweet):
    url = re.compile(r'https?://\S+|www\.\S+')
    return url.sub(r'', tweet)

#Remove Emoji
def remove_emoji(tweet):
    emoji_pattern = re.compile("["
        u"\U0001F600-\U0001F64F"
        u"\U0001F300-\U0001F5FF"
        u"\U0001F1E0-\U0001F1FF"
    "| | | | | | | | "+", flags=re.UNICODE)
    return emoji_pattern.sub(r'', tweet)

#remove html
def remove_html(tweet):
    html = re.compile(r'<.*?>')
    return html.sub(r'', tweet)

# remove numbers
def remove_numbers(tweet):
    numbers = re.sub(r'\d+', '', tweet)
    return numbers

#menghapus symbol
def remove_symbols(tweet):
    symbols = re.sub(r'^[a-zA-Z0-9\s]', '', tweet) # Menghapus semua karakter yang bukan huruf, angka, atau spasi
    return symbols

#menghapus kata kata random
def remove_random_words(tweet):
    # Menghapus kata-kata yang terlihat seperti urutan karakter acak
    random_words = re.sub(r'\b\w{10,}\b', '', tweet)
    return random_words

data['cleansing'] = data['content'].apply(lambda x: remove_URL(x))
data['cleansing'] = data['cleansing'].apply(lambda x: remove_emoji(x))
data['cleansing'] = data['cleansing'].apply(lambda x: remove_html(x))
data['cleansing'] = data['cleansing'].apply(lambda x: remove_numbers(x))
data['cleansing'] = data['cleansing'].apply(lambda x: remove_symbols(x))
data['cleansing'] = data['cleansing'].apply(lambda x: remove_random_words(x))
```

```
| neg = df.loc[df['sentiment'] == 'negative'].content_clean.tolist()
neu = df.loc[df['sentiment'] == 'neutral'].content_clean.tolist()
pos = df.loc[df['sentiment'] == 'positive'].content_clean.tolist()

neg_sentiment = df.loc[df['sentiment'] == 'negative'].sentiment.tolist()
neu_sentiment = df.loc[df['sentiment'] == 'neutral'].sentiment.tolist()
pos_sentiment = df.loc[df['sentiment'] == 'positive'].sentiment.tolist()

total_data = pos + neu + neg
sentiments = pos_sentiment + neu_sentiment + neg_sentiment

print("Pos: %s, Neu: %s, Neg: %s" % (len(pos), len(neu), len(neg)))
print("Total data: %s" % len(total_data))
```

```
Pos: 6416, Neu: 1148, Neg: 3436
Total data: 11000
```

```
from sklearn.model_selection import train_test_split

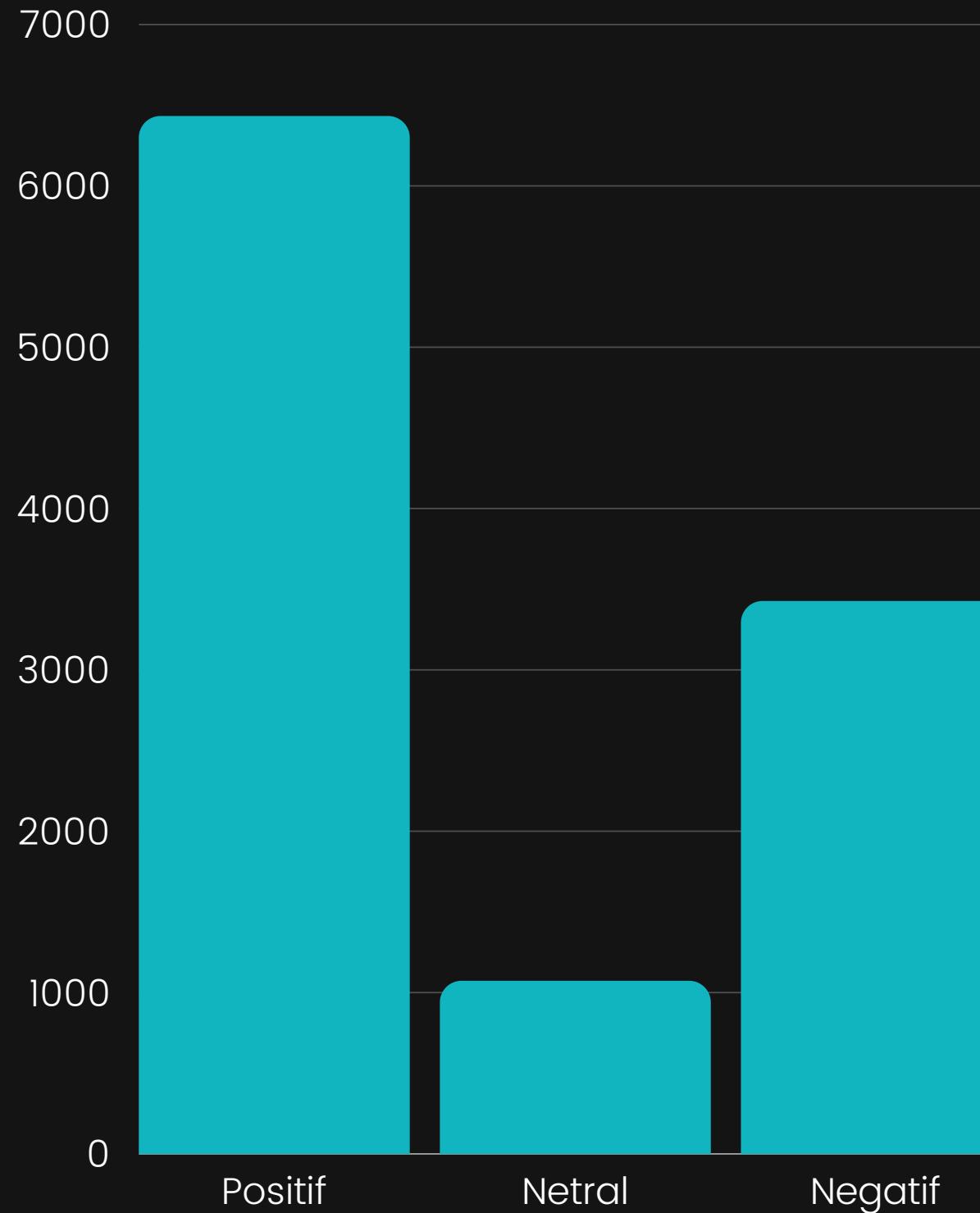
classes = df.label
classes
```

0	positif
1	netral
2	positif
3	positif
4	negatif
	...
10928	netral
10929	positif
10930	positif
10931	negatif
10932	positif

Name: label, Length: 10933, dtype: object

# Jumlah Data Berdasarkan Sentiment

## Neural Network



```
[36] #jumlah sentimen positif, netral dan negatif  
  
count_label = df['sentiment'].value_counts()  
count_label  
  
positif      6433  
negatif      3427  
netral       1073  
Name: sentiment, dtype: int64
```

Berdasarkan hasil di samping dapat dilihat bahwa sentimen yang paling banyak, terdapat di sentimen positif sebesar 6433, diikuti dengan sentimen negatif dengan jumlah 3427 dan di urutan terakhir adalah sentimen netral dengan angka 1073.

# Hasil Training Model

## Neural Network

Training ke- 1				
	precision	recall	f1-score	support
negatif	0.98	0.97	0.97	697
netral	0.91	0.97	0.94	200
positif	0.99	0.98	0.98	1290
accuracy			0.98	2187
macro avg	0.96	0.97	0.97	2187
weighted avg	0.98	0.98	0.98	2187
=====				
Training ke- 2				
	precision	recall	f1-score	support
negatif	0.98	0.97	0.98	670
netral	0.94	0.97	0.95	210
positif	0.99	0.99	0.99	1307
accuracy			0.98	2187
macro avg	0.97	0.98	0.97	2187
weighted avg	0.98	0.98	0.98	2187
=====				
Training ke- 3				
	precision	recall	f1-score	support
negatif	0.99	0.98	0.98	700
netral	0.95	0.97	0.96	202
positif	0.99	0.99	0.99	1285
accuracy			0.98	2187
macro avg	0.98	0.98	0.98	2187
weighted avg	0.98	0.98	0.98	2187

Training ke- 4				
	precision	recall	f1-score	support
negatif	0.99	0.98	0.98	701
netral	0.96	0.97	0.96	227
positif	0.99	0.99	0.99	1258
accuracy				2186
macro avg	0.98	0.98	0.98	2186
weighted avg	0.98	0.98	0.98	2186
=====				
Training ke- 5				
	precision	recall	f1-score	support
negatif	0.98	0.97	0.98	659
netral	0.92	0.97	0.94	234
positif	0.99	0.99	0.99	1293
accuracy				2186
macro avg	0.96	0.98	0.97	2186
weighted avg	0.98	0.98	0.98	2186
=====				
Rata-rata Accuracy: 0.981889782884892				

# Hasil Classification Report

## Neural Network

Akurasi: 0.98

Classification Report Naive Bayes

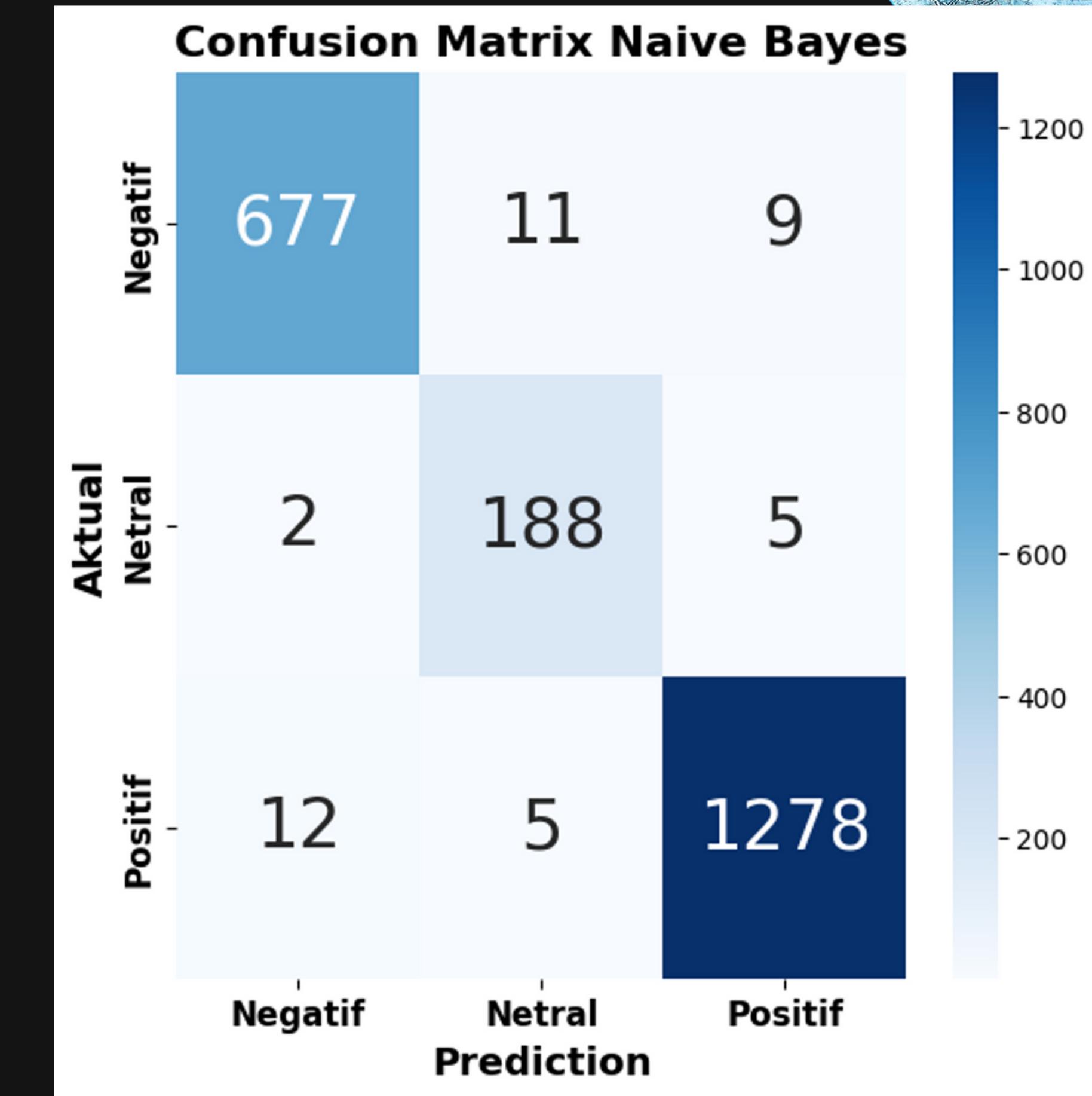
	precision	recall	f1-score	support
negatif	0.98	0.97	0.98	675
netral	0.93	0.97	0.95	219
positif	0.99	0.98	0.99	1293
accuracy			0.98	2187
macro avg	0.97	0.98	0.97	2187
weighted avg	0.98	0.98	0.98	2187

Evaluasi Confusion Matrix Naive Bayes

```
[[ 657     8    10]
 [   2   213     4]
 [ 11      9 1273]]
```

# Hasil Confusin Matrix

Neural Network



# Model LSTM

Long Short Term Memory

# Pembagian Data Training dan Testing Long Short Term Memory (LSTM)

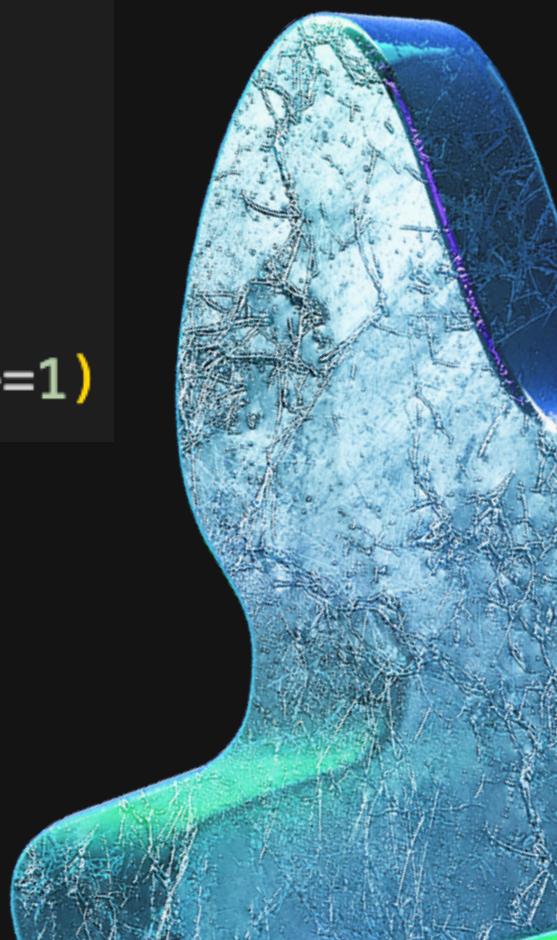
```
from sklearn.model_selection import train_test_split

file = open("x_pad_sequences.pickle",'rb')
X = pickle.load(file)
file.close()

file = open("y_labels.pickle",'rb')
Y = pickle.load(file)
file.close()

X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2, random_state=1)
```

Data dibagi menjadi 20% Data Testing dan 80% Data Training



# Training and Modelling

## Long Short Term Memory (LSTM)

```
Model: "sequential"
-----
Layer (type)          Output Shape       Param #
=====
embedding (Embedding) (None, 96, 100)    100000000
lstm (LSTM)           (None, 64)         42240
dense (Dense)         (None, 3)          195
=====
Total params: 10042435 (38.31 MB)
Trainable params: 10042435 (38.31 MB)
Non-trainable params: 0 (0.00 Byte)

WARNING:absl:`lr` is deprecated in Keras optimizer, please use `learning_rate` or use the legacy optimizer, e.g.,tf.keras.optimizers.legacy.Adam.
None
Epoch 1/10
880/880 [=====] - 283s 315ms/step - loss: 0.4501 - accuracy: 0.8215 - val_loss: 0.3270 - val_accuracy: 0.8773
Epoch 2/10
880/880 [=====] - 259s 294ms/step - loss: 0.1947 - accuracy: 0.9292 - val_loss: 0.3309 - val_accuracy: 0.8782
Epoch 3/10
880/880 [=====] - 259s 294ms/step - loss: 0.1096 - accuracy: 0.9563 - val_loss: 0.3897 - val_accuracy: 0.8682
Epoch 3: early stopping
```

Ketika model diterapkan dengan uji coba 10 epoch ternyata  
LSTM berhenti lebih awal pada epoch 3

# Classification Report

## Long Short Term Memory (LSTM)

69/69 [=====] - 2s 17ms/step				
Testing selesai				
	precision	recall	f1-score	support
0	0.81	0.83	0.82	685
1	0.87	0.69	0.77	233
2	0.90	0.92	0.91	1282
accuracy			0.87	2200
macro avg	0.86	0.81	0.83	2200
weighted avg	0.87	0.87	0.87	2200

Akurasi dari model LSTM setelah diterapkan sebesar 87%

# Hasil Training Model

## LSTM

Training ke- 1				
	precision	recall	f1-score	support
0	0.82	0.83	0.83	685
1	0.82	0.80	0.81	233
2	0.91	0.91	0.91	1282
accuracy			0.88	2200
macro avg	0.85	0.85	0.85	2200
weighted avg	0.88	0.88	0.88	2200

Training ke- 3				
	precision	recall	f1-score	support
0	0.82	0.83	0.82	685
1	0.83	0.76	0.80	233
2	0.91	0.91	0.91	1282
accuracy			0.87	2200
macro avg	0.85	0.84	0.84	2200
weighted avg	0.87	0.87	0.87	2200

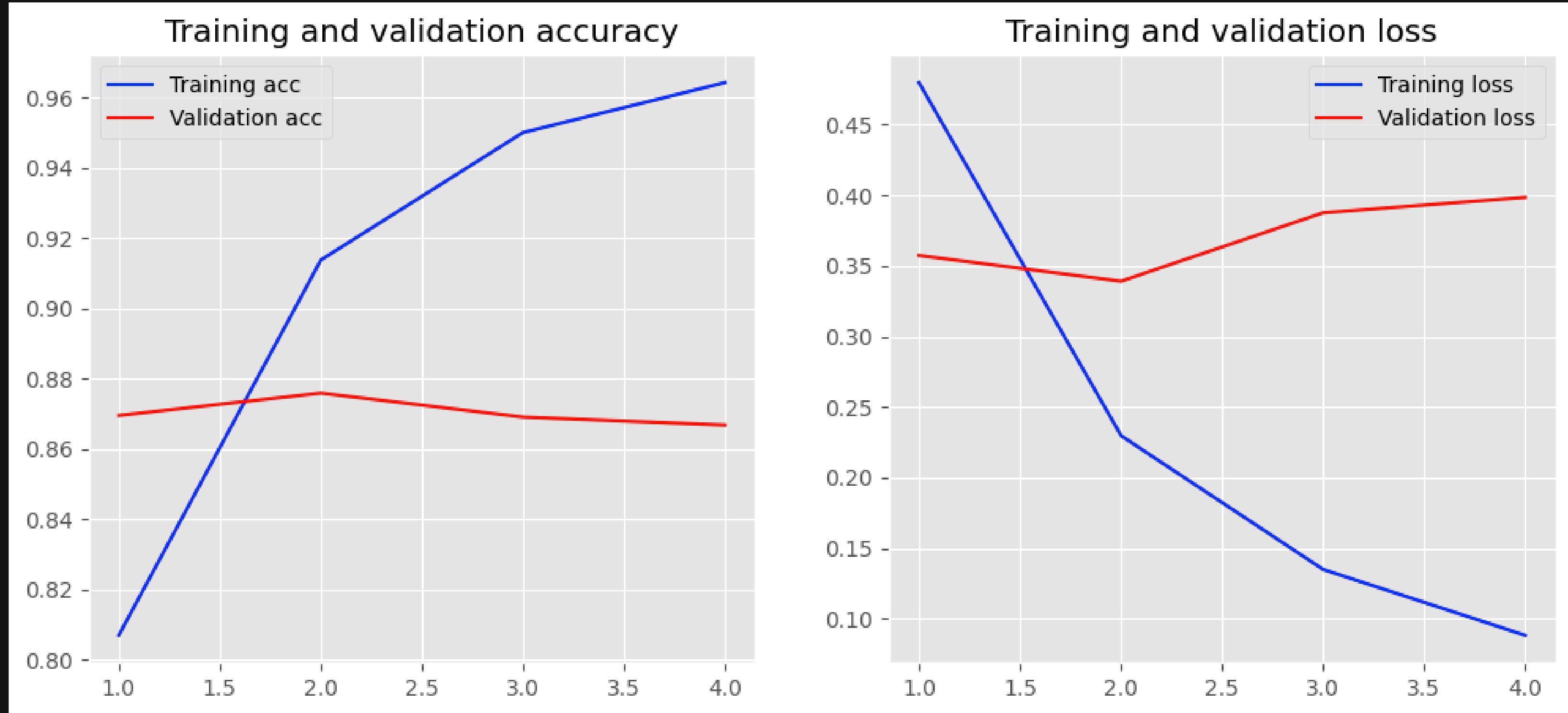
Training ke- 2				
	precision	recall	f1-score	support
0	0.78	0.82	0.80	685
1	0.72	0.82	0.76	233
2	0.91	0.87	0.89	1282
accuracy			0.85	2200
macro avg	0.80	0.83	0.82	2200
weighted avg	0.85	0.85	0.85	2200

Training ke- 4				
	precision	recall	f1-score	support
0	0.83	0.82	0.82	685
1	0.82	0.72	0.77	233
2	0.89	0.92	0.91	1282
accuracy			0.87	2200
macro avg	0.85	0.82	0.83	2200
weighted avg	0.87	0.87	0.87	2200

Rata-rata akurasi 0.86 ini menunjukan bahwa model yang digunakan berada di angka yang cukup stabil

# Visualization Model

LSTM



# Predict Hasil Sentiment

## LSTM Menggunakan Data Tweet

```
1/1 [=====] - 0s 499ms/step
```

```
Text: susilo bambang yudhoyono presiden ri ke 6 meminta khofifah emil untuk dekat dengan masyarakat
```

```
Sentiment: neutral
```

```
1/1 [=====] - 0s 377ms/step
```

```
Text: disaat semua cowok berusaha melacak perhatian gue loe lantas remehkan perhatian yg gue kasih khusus ke elo basic elo cowok bego
```

```
Sentiment: negative
```

```
1/1 [=====] - 1s 529ms/step
```

```
Text: syukurlah
```

```
Sentiment: positive
```



# Hasil API

The screenshot shows a web browser window displaying the Swagger UI for a sentiment prediction API. The URL in the address bar is `127.0.0.1:5000/docs/`. The page title is "API For Sentiment Prediction, By Group 3 <sup>1.0.0</sup>". Below the title, it says "[ Base URL: 127.0.0.1:5000 ] /docs.json". A sub-header states "API untuk Prediksi Sentimen, Oleh Kelompok 3".

The API documentation is organized into sections:

- Sentiment Analysis using Neural Network**:
  - POST /Neural Network** `post_Neural_Network`
- Upload File Data Tweet/X Menggunakan LSTM**:
  - POST /Upload File LSTM** `post_Upload_File_LSTM`
- Upload File Data Tweet/X Menggunakan Neural Network**:
  - POST /Upload File Neural Network** `post_Upload_File_Neural_Network`
- Sentiment Analysis Using LSTM**:
  - POST /lstm** `post_lstm`

At the bottom right of the page, it says "[Powered by [Flasgger](#) 0.9.7.1]

# Hasil API

Sentiment analysis using LSTM & Neural Network

Flagger

127.0.0.1:5000/docs/#/OrderedMap%20%7B%20"name"%3A%20"Sentiment%20Analysis%20using%20Neural%20Network"%20%7D/post\_Neural\_Network

### Sentiment Analysis using Neural Network

POST /Neural\_Network

Parameters

Name	Description
text * required	Apaan si bodo banget (formdata)

Execute Clear

Responses

Curl

```
curl -X POST "http://127.0.0.1:5000/Neural Network" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "text=Apaan si bodo banget"
```

Request URL

http://127.0.0.1:5000/Neural Network

Server response

Code Details

200

Response body

```
{ "data": { "sentiment": "negative", "text": "Apaan si bodo banget" }, "description": "Hasil Analisis Sentimen Menggunakan Neural Network", "status_code": 200 }
```

Download

Response headers

```
connection: close
content-length: 124
content-type: application/json
date: Sun, 24 Mar 2024 07:51:07 GMT
server: Werkzeug/3.0.1 Python/3.8.5
```

Responses

Flagger

127.0.0.1:5000/docs/#/OrderedMap%20%7B%20"name"%3A%20"Sentiment%20Analysis%20Using%20LSTM"%20%7D/post\_lstm

### Sentiment Analysis Using LSTM

POST /lstm

Parameters

Name	Description
text * required	Keren (formdata)

Execute Clear

Responses

Curl

```
curl -X POST "http://127.0.0.1:5000/lstm" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "text=Keren"
```

Request URL

http://127.0.0.1:5000/lstm

Server response

Code Details

200

Response body

```
{ "data": { "sentiment": "positive", "text": "Keren" }, "description": "Hasil Analisis Sentimen Menggunakan LSTM", "status_code": 200 }
```

Download

Response headers

```
connection: close
content-length: 124
content-type: application/json
date: Sun, 24 Mar 2024 07:50:48 GMT
server: Werkzeug/3.0.1 Python/3.8.5
```

Responses

# Hasil API

Upload data/files using LSTM & Neural Network

This screenshot shows the Flasgger API documentation for the 'Upload File Data Tweet/X Menggunakan Neural Network' endpoint. The interface includes:

- POST /Upload File Neural Network**
- Parameters**:
  - upload\_file \* required file (formDate)
- Responses**:
  - Curl: curl -X POST "http://127.0.0.1:5000/Upload File Neural Network" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "upload\_file=@data.csv;type=text/csv"
  - Request URL: http://127.0.0.1:5000/Upload File Neural Network
- Server response**:
  - Code: 200
  - Details: Response body (JSON array of tweet objects)

The response body contains several tweet objects, each with sentiment and clean text fields. For example:

```
[{"Sentiment": "positif", "Tweet_Clean": "kading aks berfikir kenapa aks teta percaya pada tuhan padahal aks selalu jatus berkali2 kading aks merasa tuhan itu ninggalan aks sendirian ketika orangtua berencana berpisah ketika kakaku lebih memilih jadi kristen ketika aks ter"}, {"Sentiment": "negatif", "Tweet_Clean": "user user aks itu akunku tau matane sigit tapi dililit dari mana its aks"}, {"Sentiment": "positif", "Tweet_Clean": "user user kau cehong kapur udah kelitan dongkeny dari awal tambah dong lagi hubukut"}, {"Sentiment": "negatif", "Tweet_Clean": "user ya basi taplak dkk xtxfxxfxfxfxfxfxf"}, {"Sentiment": "negatif", "Tweet_Clean": "deklarasi pilkada aman dan anti hoax warga dukuh sari jebon"}, {"Sentiment": "negatif", "Tweet_Clean": "gue bare ajia keler rewatch aldoak zero paling kampret emang endengnya karakter utama cowonya kena friendzone bray ad ur"}, {"Sentiment": "positif", "Tweet_Clean": "maka deklarasi pilkada aman dan anti hoax warga dukuh sari jebon"}]
```

This screenshot shows the Flasgger API documentation for the 'Upload File Data Tweet/X Menggunakan LSTM' endpoint. The interface includes:

- POST /Upload File LSTM**
- Parameters**:
  - upload\_file \* required file (formDate)
- Responses**:
  - Curl: curl -X POST "http://127.0.0.1:5000/Upload File LSTM" -H "accept: application/json" -H "Content-Type: multipart/form-data" -F "upload\_file=@Data Test\_API\_LSTM"
  - Request URL: http://127.0.0.1:5000/Upload File LSTM
- Server response**:
  - Code: 200
  - Details: Response body (JSON array of tweet objects)

The response body contains several tweet objects, each with sentiment and clean text fields. For example:

```
[{"Sentiment": "positive", "Tweet_Clean": "waring iai diajiki oleh pengusaha pabrik tahu yang sudah puluhan tahun terkenal membuat tahu putih di bandung tahu berkualitas dipadu kekualitas manisnya dengan saus kacang yang enak dan pas dengan nasi goreng"}, {"Sentiment": "positive", "Tweet_Clean": "mohon ulama lurus dan k mabri hujah portal apa yang harus diulah agar suara islam tidak pecahpecah"}, {"Sentiment": "positive", "Tweet_Clean": "lokasi strategis di jalan sumatera bandung tempat nya nyaman terutama sofa di lantai paella nya enak sangat pas dimakan dengan minuman dingin appetiser nya juga enak-enak"}, {"Sentiment": "positive", "Tweet_Clean": "betapa bahagia nya diri ini saat unboxing paket dan barang nya bagus menetapkan beli lagi"}, {"Sentiment": "positive", "Tweet_Clean": "nah judi mahasiswa jangan sombang dong kartu kartu kuning segala belajar doa yang batik tidak usahlah ikutik politik nanti sudah selesai kulin nya mau ikut politik juga tidak telat dasar mahasiswa"}, {"Sentiment": "positive", "Tweet_Clean": "maka deklarasi pilkada aman dan anti hoax warga dukuh sari jebon"}]
```

# Kesimpulan dan Saran

Hasil akurasi dari setiap model algoritma sudah cukup baik, dimana algoritma Neural Network menghasilkan akurasi sebesar 98% dan Model Algoritma LSTM menghasilkan akurasi sebesar 86%. Akan tetapi meskipun nilai akurasi terbilang cukup tinggi, eksperiment ini belum bisa diterapkan kepada user dikarnakan diperlukan pengecekan lebih lanjut untuk mengoptimalkan model tersebut, model tersebut harus dilakukan penanganan dalam mengatasi overfitting dan underfitting atau data yang tidak optimal sehingga hasil prediksi nya akan jauh lebih baik.