

# CHALLENGE PLATINUM : SENTIMENT ANALYSIS DENGAN LSTM DAN CNN

DISUSUN OLEH :  
FAJRI NOVITASARI  
MIFTAH ACHMANDA  
(-) FIONAL

# PENDAHULUAN

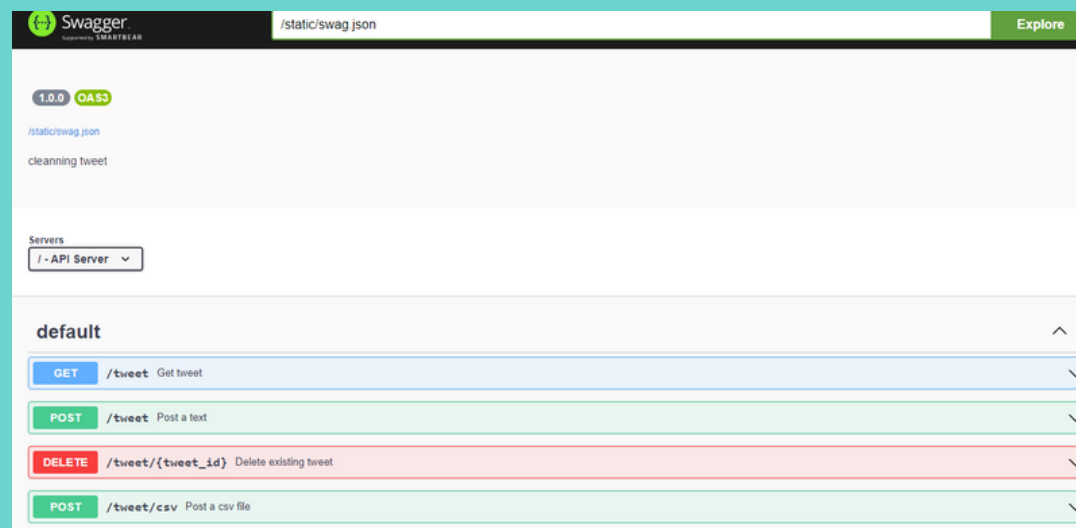
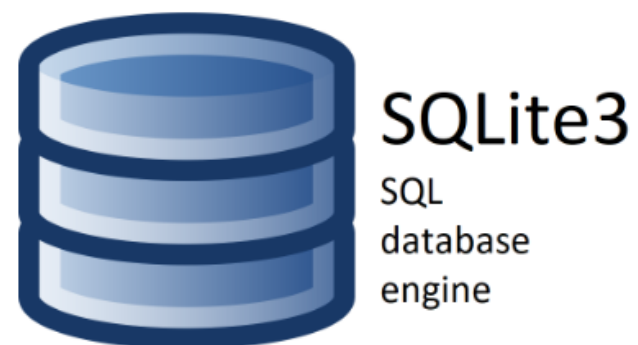
## API untuk melakukan penerapan sentiment analisis

pada challenge chapter platinum diminta untuk melakukan cleansing data dengan menggunakan API/restful API dimana API RESTful adalah antarmuka yang digunakan oleh dua sistem komputer untuk bertukar informasi secara aman melalui internet .

tools yang digunakan :

- sqlite3 --> untuk membuat database
- python --> bahasa pemrograman yang digunakan
- flask --> library framework untuk membuat api
- swagger ui --> untuk membuat tampilan ap
- library padas
- library regex
- metode LSTM
- metode CNN

LSTM dan CNN digunakan dalam pembuatan model mesin learning untuk mengenal/mengetahui positif, negatif, dan netral dari suatu kalimat



# TUJUAN

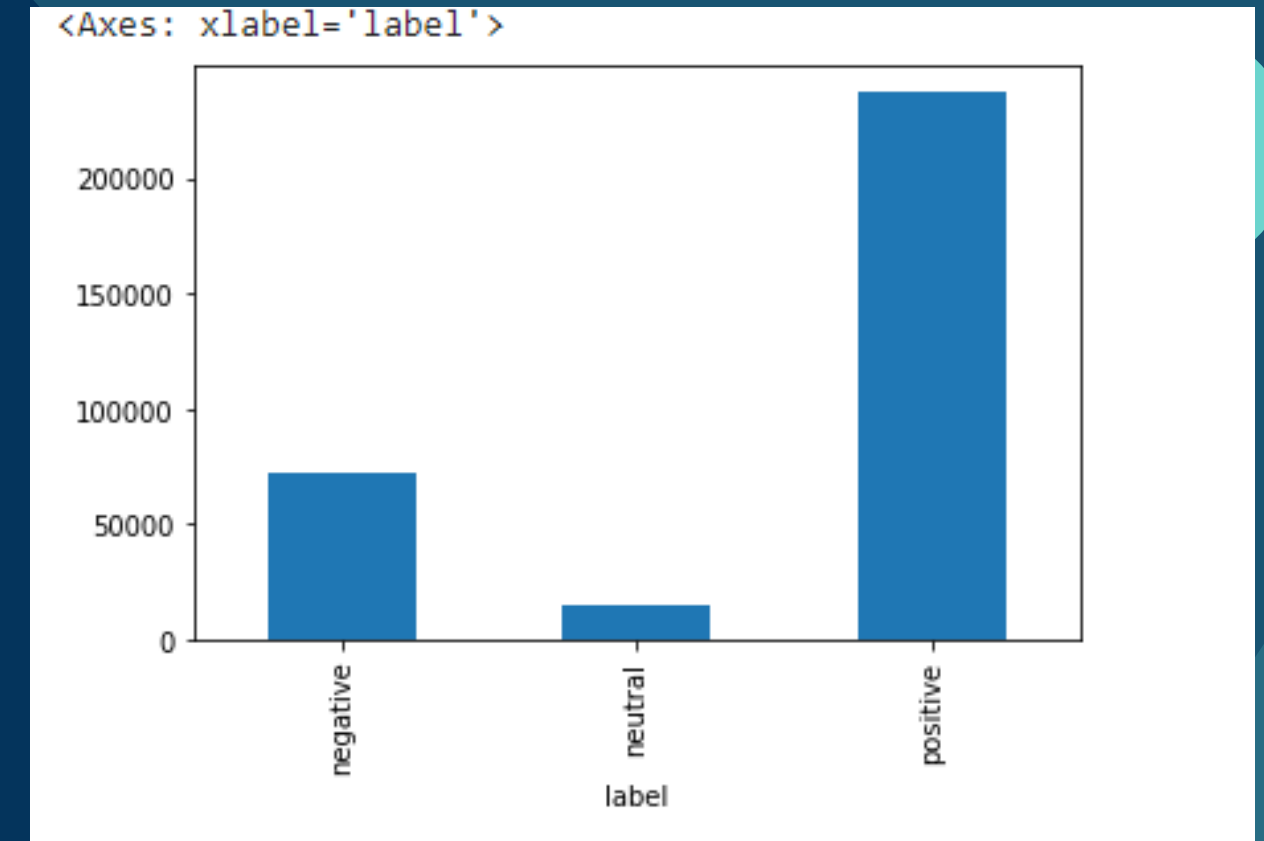
Tujuan dari penelitian ini adalah untuk menghasilkan sebuah API yang mampu menghasilkan output berupa sentimen yang dikategorikan sebagai 'positif', 'negatif', dan 'netral' berupa masukan data text dan file

# MELAKUKAN PREPROSESING DATA

## Analisis terhadap data

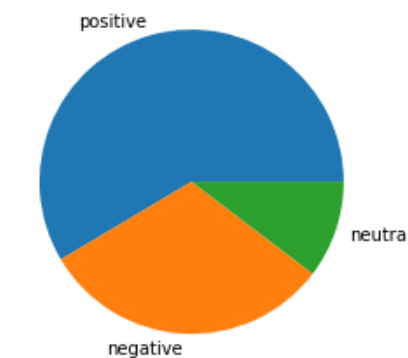
Pada data yang di berikan bahwa:

- Data training berjumlah 9900 baris dengan 2 kolom berupa tweet dan label sentimen
- kata positive lebih banyak dibandingkan data kata negative dan neutral pada data train



```
plt.pie(df['label'].value_counts(), labels=df['label'].value_counts().index)
```

```
([<matplotlib.patches.Wedge at 0x7f18e0ac6490>,  
<matplotlib.patches.Wedge at 0x7f18e0a65370>,  
<matplotlib.patches.Wedge at 0x7f18e09a8490>],  
[Text(-0.29159537105240846, 1.0606470381709499, 'positive'),  
Text(-0.06215344486204681, -1.098242664119265, 'negative'),  
Text(1.042476544718992, -0.3510593307558585, 'neutral')])
```



# METODE DATA CLEANSING

## 1. Membuat database dengan import sqlite

Melakukan insert pada database dengan membuat database untuk setiap model yaitu sentiment\_cnn dan sentiment\_lstm dengan perintah CREATE TABLE

```
sqlite.py  X
sqlite.py > ...
1
2  import sqlite3
3  db = sqlite3.connect('database.db', check_same_thread=False)
4  db.text_factory = bytes
5  mycursor = db.cursor()
6  table1 = db.execute('''CREATE TABLE IF NOT EXISTS sentiment_cnn (cnn_id INTEGER PRIMARY KEY AUTOINCREMENT
7  table2 = db.execute('''CREATE TABLE IF NOT EXISTS sentiment_lstm (lstm_id INTEGER PRIMARY KEY AUTOINCREMENT
8
9
10
```

## 2. Melakukan cleansing data

Pada data kami melakukan cleansing data dengan regex dimana menghilangkan angka dan tanda baca serta mengganti semua huruf besar menjadi kecil

```
datacleansing.py X flask_app.py
datacleansing.py > ...
10
11 def remove_tweet(text):
12     text = text.lower() #proses merubah huruf menjadi kecil
13     text = re.sub('\n', ' ', text)
14     text = re.sub('rt', ' ', text)
15     text = re.sub(r'[0-9]+', '', text)
16     text = re.sub('user', ' ', text)
17     text = re.sub('((www\.[^\s]+)|(https?://[^\s]+)|(http?://[^\s]+))', ' ', text)
18     text = re.sub(r'#', '', text)
19     text = re.sub(r',', '', text)
20     text = re.sub(r'[0-9]+', '', text)
21     text = re.sub(' +', ' ', text)
22     text = re.sub('[^0-9a-zA-Z]+', '', text)
23     text = re.sub(' +', ' ', text)
24
25     return text
26
```

# MEMBUAT MODEL LSTM

## 1. Mengecek tingkat akurasi model

pada model didapatkan nilai akurasinya yaitu 86% dimana nilai ini merupakan nilai yang cukup tinggi, selain itu juga kami melakukan pengecekan akurasi dengan menggunakan cross validation untuk meyakinkan hasilnya

```
69/69 [=====] - 2s 18ms/step
Training ke- 5
          precision    recall  f1-score   support

         0           0.83       0.83       0.83         685
         1           0.84       0.69       0.76         233
         2           0.91       0.93       0.92        1282

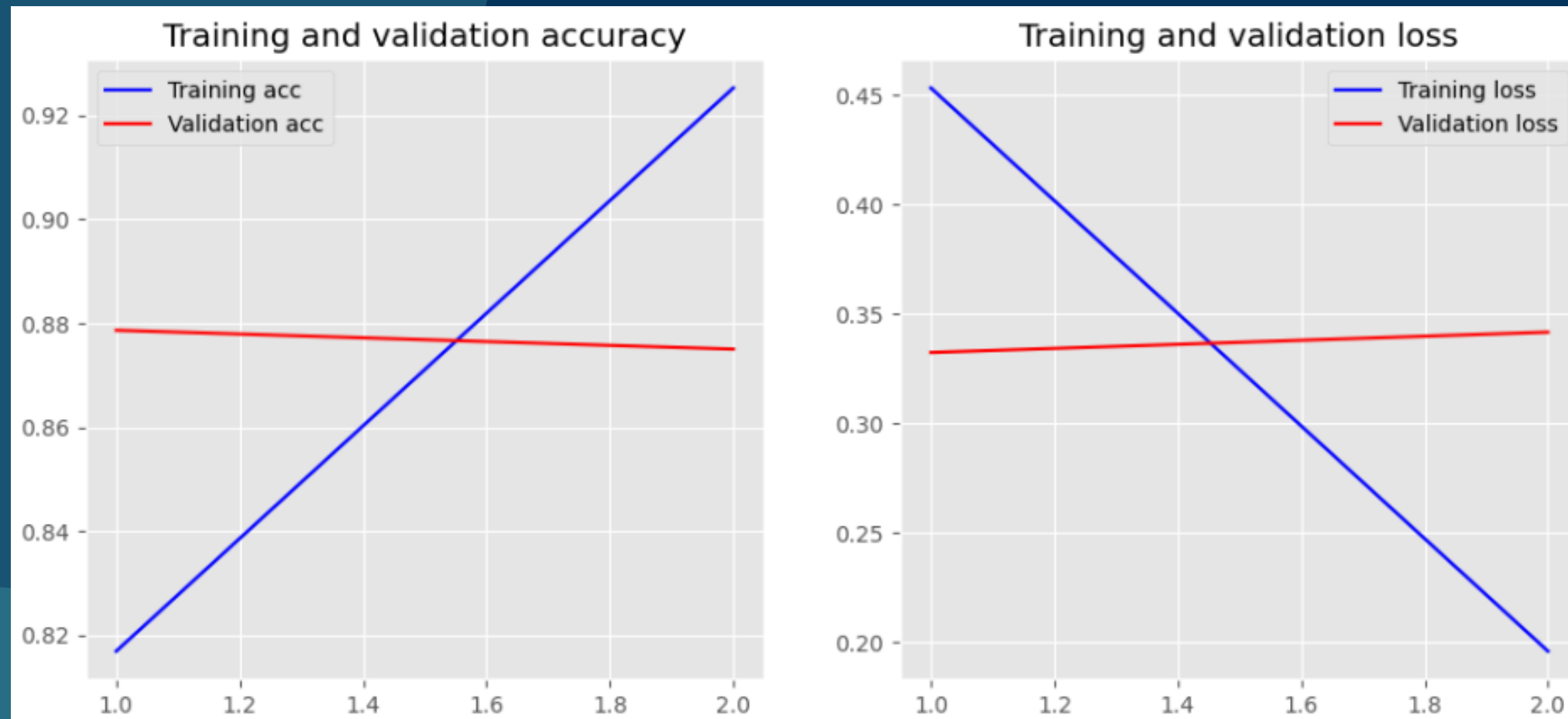
 accuracy              0.88         2200
 macro avg           0.86       0.82       0.83         2200
weighted avg           0.87       0.88       0.87         2200

=====

Rata-rata Accuracy: 0.863181818181818
```

# MODEL LSTM

## 2. Visualisasi Training and Validation Accuracy & Lost pada Model LSTM



Hasilnya jika kita melihat visualisasinya model LSTM tergolong Underfitting.



# MODEL CNN

## 1.Mengecek tingkat akurasi model

pada model didapatkan nilai akurasi yaitu 88% dimana nilai ini merupakan nilai yang cukup tinggi, selain itu juga kami melakukan pengecekan akurasi dengan menggunakan cross validation untuk meyakinkan hasilnya

```
=====
69/69 [=====] - 1s 11ms/step
Training ke- 5

```

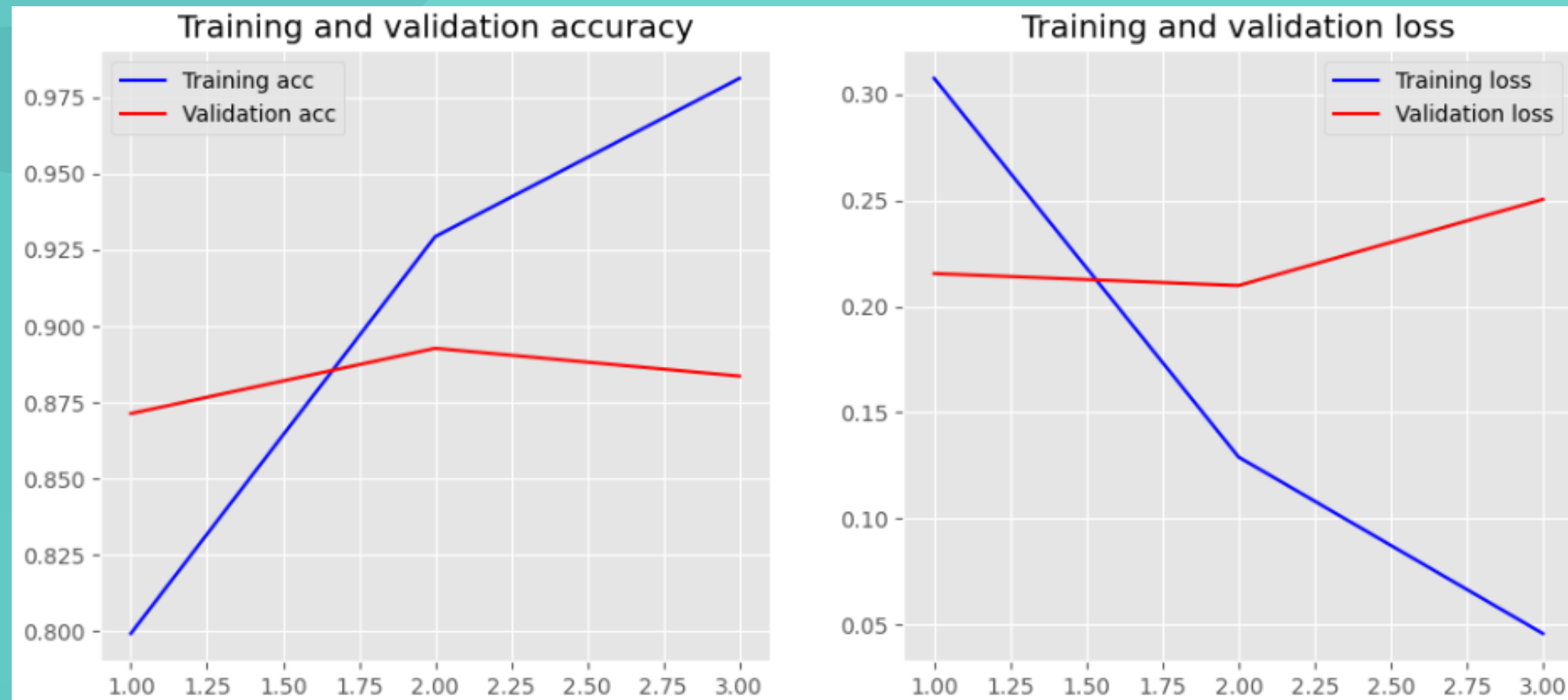
	precision	recall	f1-score	support
0	0.81	0.87	0.84	685
1	0.81	0.73	0.77	233
2	0.94	0.92	0.93	1282
accuracy			0.88	2200
macro avg	0.85	0.84	0.85	2200
weighted avg	0.89	0.88	0.88	2200

```
=====

Rata-rata Accuracy: 0.8840909090909091
```

# MODEL CNN

## 2. Visualisasi Training and Validation Accuracy & Lost pada Model CNN



Hasilnya jika kita melihat visualisasinya model CNN tergolong Underfitting.

# KONFIGURASI SWAGGER DAN FLASK

```
flask_app.py X
flask_app.py > ...

14
15 # instantiate flask object
16 app = Flask(__name__)
17 # set app configs
18 app.config['JSON_SORT_KEYS'] = False
19
20 # flask swagger configs
21 SWAGGER_URL = '/swagger'
22 API_URL = '/static/swag.json'
23 SWAGGERUI_BLUEPRINT = get_swaggerui_blueprint(
24     SWAGGER_URL,
25     API_URL,
26     config={
27         'app_name': "Data Analyst Sentiment!"
28     }
29 )
30 app.register_blueprint(SWAGGERUI_BLUEPRINT, url_prefix=SWAGGER_URL)
31
32 # Database
33 db = sqlite3.connect('database.db', check_same_thread=False)
34 db.row_factory = sqlite3.Row
35 mycursor = db.cursor()
36

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL SQL CONSOLE
```

# Konfigurasi Rooting CNN

```
flask_app.py > ...
44  ### ===== ANALYST SENTIMENT =====
45
46  #CNN
47  file = open("model/cnn/x_pad_sequences.pickle", 'rb')
48  feature_file_from_cnn = pickle.load(file)
49  file.close()
50
51  tokenizer = joblib.load('model/cnn/tokenizer.pickle')
52  model_file_from_cnn = load_model("model/cnn/modelcnn.h5")
53
54  def predict_sentiment_cnn(text):
55      sentiment = ['negative', 'neutral', 'positive']
56      text = [cleansing_text(text)]
57      feature = tokenizer.texts_to_sequences(text)
58      feature = pad_sequences(feature, maxlen=feature_file_from_cnn.shape[1])
59      prediction = model_file_from_cnn.predict(feature)
60      get_sentiment = np.argmax(prediction[0])
61      return sentiment[get_sentiment]
62
63  def sentiment_cnn_csv(input_file):
64      column = input_file.iloc[:, 0]
65      print(column)
66
```

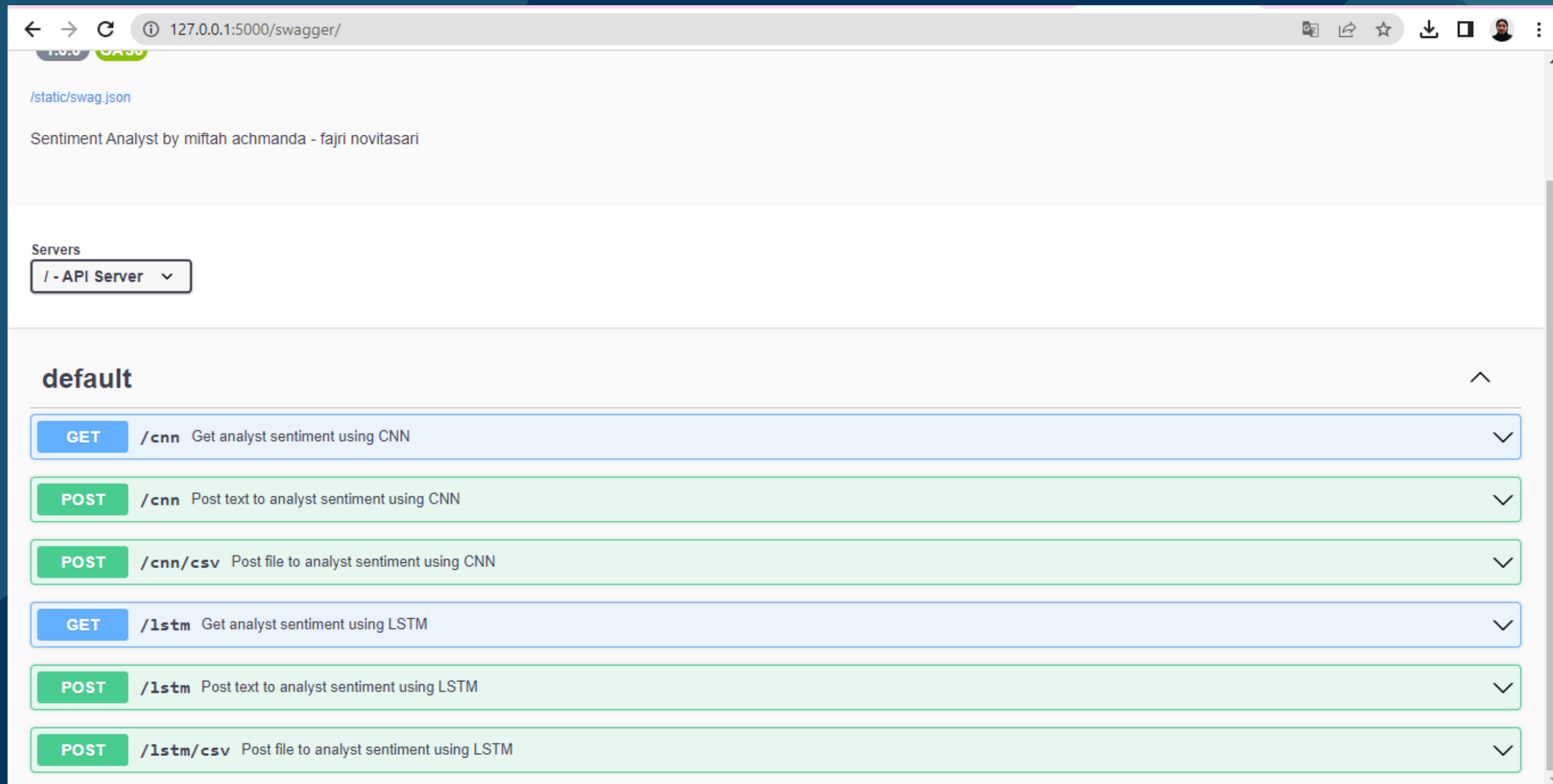
```
flask_app.py > ...
76  ### INPUT TEXT ###
77  @app.route("/cnn", methods=['POST'])
78  def cnn():
79      original_text = str(request.form["text"]) #get text from user
80      text = cleansing_text(original_text) #cleaning text
81      text_sentiment = predict_sentiment_cnn(text)
82
83      query = "insert into sentiment_cnn (original_text, clean_text, analyst_sentiment) values (?, ?, ?)"
84      variable = (original_text, text, str(text_sentiment))
85      mycursor.execute(query, variable)
86      db.commit()
87
88      # Define API response
89      json_response = {
90          'description': "Analysis Sentiment Success!",
91          'original_text' : original_text,
92          'text' : text,
93          'sentiment' : text_sentiment
94      }
95      response_data = jsonify(json_response)
96      return response_data
97
98  @app.route("/cnn", methods = ["GET"])
```

# Konfigurasi Rooting LSTM

```
flask_app.py X
flask_app.py > ...

132 #-----LSTM-----#
133 file = open("model/lstm/x_pad_sequences.pickle", 'rb')
134 feature_file_from_lstm = pickle.load(file)
135 file.close()
136
137 tokenizer = joblib.load('model/lstm/tokenizer.pickle')
138 model_file_from_lstm = load_model("model/lstm/model_lstm.h5")
139
140 def predict_sentiment_lstm(text):
141     sentiment = ['negative', 'neutral', 'positive']
142     text = [cleansing_text(text)]
143     feature = tokenizer.texts_to_sequences(text)
144     feature = pad_sequences(feature, maxlen=feature_file_from_lstm.shape[1])
145     prediction = model_file_from_lstm.predict(feature)
146     get_sentiment = np.argmax(prediction[0])
147     return sentiment[get_sentiment]
148
149
150 def sentiment_lstm_csv(input_file):
151     column = input_file.iloc[:, 0]
152     print(column)
153
154     for data_file in column: # Define and execute query for insert cleaned text and sentiment to sqlite database
```

# TAMPILAN SWAGGER UI DAN HASIL



# Hasil Melakukan Sentiment Analisis dengan CNN

```
[
  {
    "cnn_id": 1,
    "original_text": "- disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego ! ! !'",
    "clean_text": " disaat semua cowok berusaha melacak perhatian gue loe lantas remehkan perhatian yg gue kasih khusus ke elo basic elo cowok bego ",
    "analyst_sentiment": "negative"
  },
  {
    "cnn_id": 2,
    "original_text": "RT USER: USER siapa yang telat ngasih tau elu?edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga'",
    "clean_text": " siapa yang telat ngasih tau elu edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga ",
    "analyst_sentiment": "negative"
  },
  {
    "cnn_id": 3,
    "original_text": "41. Kadang aku berfikir, kenapa aku tetap percaya pada Tuhan padahal aku selalu jatuh berkali-kali. Kadang aku merasa Tuhan itu meninggalkan aku sendirian. Ketika orangtuaku berencana berpisah, ketika kakakku lebih memilih jadi Kristen. Ketika aku anak ter",
    "clean_text": " kadang aku berfikir kenapa aku tetap percaya pada tuhan padahal aku selalu jatuh berkali kali kadang aku merasa tuhan itu meninggalkan aku sendirian ketika orangtuaku berencana berpisah ketika kakakku lebih memilih jadi kristen ketika aku anak ter",
    "analyst_sentiment": "positive"
  },
  {
    "cnn_id": 4,
    "original_text": "USER USER AKU ITU AKU\\n\\nKU TAU MATAMU SIPIT TAPI DILIAT DARI MANA ITU AKU'",
    "clean_text": " aku itu aku n nku tau matamu sipit tapi diliat dari mana itu aku ",
    "analyst_sentiment": "negative"
  },
]
```



Download

Response headers

# Hasil Melakukan Sentiment Analisis dengan LSTM

```
[
  {
    "lstm_id": 1,
    "original_text": "tempat makan yang cukup nyaman , namun sangat minim tempat parkir untuk mobil . ramen nya adalah ramen yang paling enak yang pernah saya makan",
    "clean_text": "tempat makan yang cukup nyaman namun sangat minim tempat parkir untuk mobil ramen nya adalah ramen yang paling enak yang pernah saya makan",
    "analyst_sentiment": "positive"
  },
  {
    "lstm_id": 2,
    "original_text": "bukalapak sedang membuka promo besar-besaran",
    "clean_text": "bukalapak sedang membuka promo besar besaran",
    "analyst_sentiment": "neutral"
  },
  {
    "lstm_id": 3,
    "original_text": "- disaat semua cowok berusaha melacak perhatian gue. loe lantas remehkan perhatian yg gue kasih khusus ke elo. basic elo cowok bego ! ! !'",
    "clean_text": " disaat semua cowok berusaha melacak perhatian gue loe lantas remehkan perhatian yg gue kasih khusus ke elo basic elo cowok bego ",
    "analyst_sentiment": "neutral"
  },
  {
    "lstm_id": 4,
    "original_text": "RT USER: USER siapa yang telat ngasih tau elu?edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga'",
    "clean_text": " siapa yang telat ngasih tau elu edan sarap gue bergaul dengan cigax jifla calis sama siapa noh licew juga ",
    "analyst_sentiment": "negative"
  },
  {
    "lstm_id": 5,
    "original_text": "41. Kadang aku berfikir, kenapa aku tetap percaya pada Tuhan padahal aku selalu jatuh berkali-kali. Kadang aku merasa Tuhan itu ninggalkan aku sendirian"
```



Download



## KESIMPULAN

pada hasil pemodelan LSTM memiliki tingkat akurasi 86% dan CNN memiliki tingkat akurasi 88% dimana hasil tersebut merupakan tingkat yang relative tinggi. LSTM dan CNN juga merupakan salah satu pemodelan mesin learning neural network dimana pada model tersebut menggunakan data train dan test, data train disini sebagai data latih yang di uji untuk setiap algoritma pemodelan sedangkan data testing digunakan sebagai pengukur tingkat keakuratan data tesebut.

## **SARAN**

- Penambahan data training untuk meningkatkan performa model learning
- Penambahan proses cleansing data untuk meningkatkan hasil lebih efektif