

The background features a dark blue gradient with three glowing, translucent 3D torus shapes. One ring is positioned in the upper left, another in the lower right, and a third smaller one is at the bottom left corner.

Sentiment Analysis

By : Kelompok 5

Langkah Pengeraaan

Menyiapkan Dataset

- Mencari Dataset
- Membersihkan Dataset
- Menyeimbangkan Dataset

Membuat Model

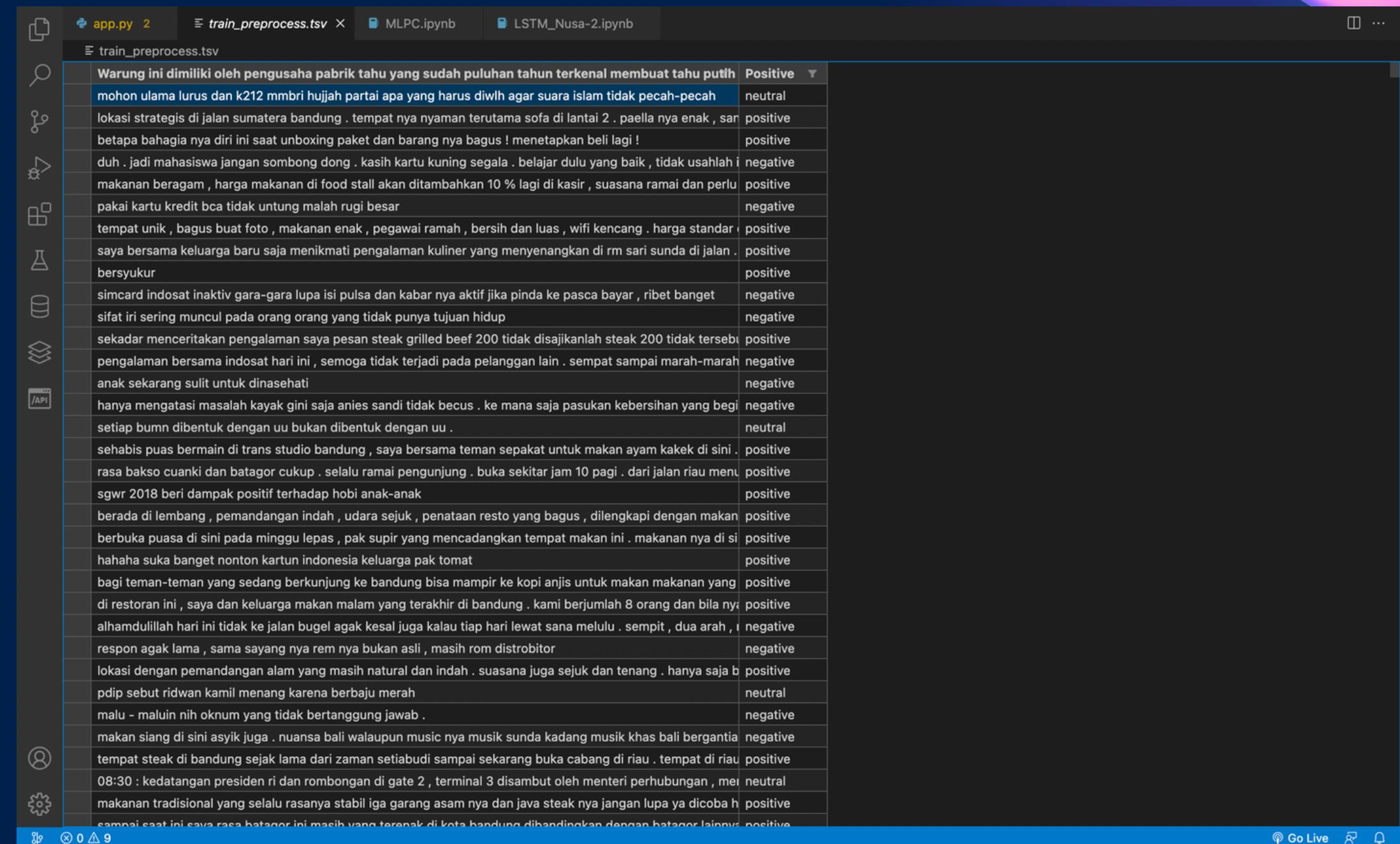
- Melakukan Feature Extraction
- Membuat Model untuk Training
- Validasi Model

Membuat API

- Membuat API dengan Flask dan Swagger
- Menggunakan sqlite Sebagai Database untuk Menampilkan Data.

Menyiapkan Dataset

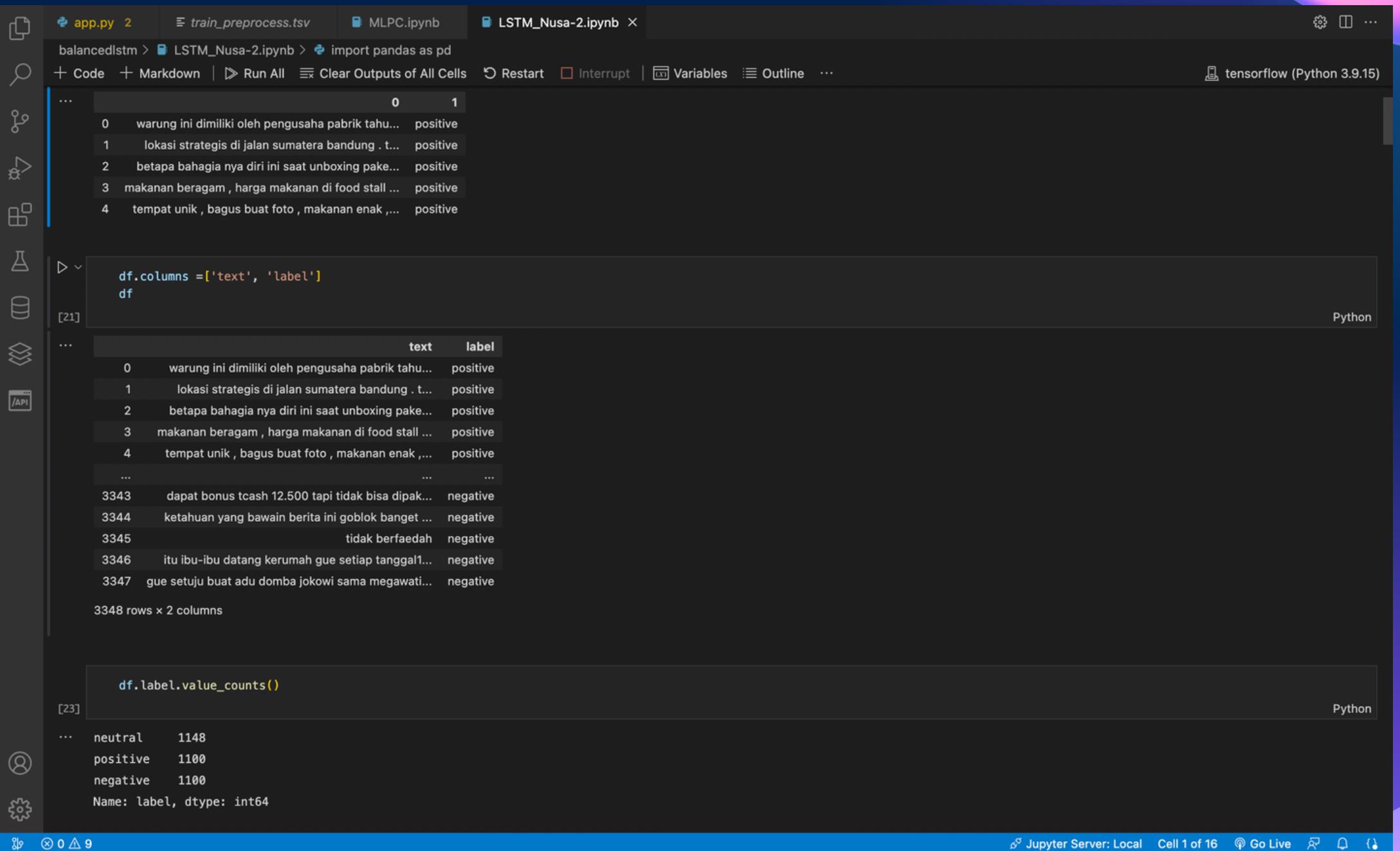
Dataset yang digunakan merupakan tweet yang sudah dikumpulkan



Text	Label
Warung ini dimiliki oleh pengusaha pabrik tahu yang sudah puluhan tahun terkenal membuat tahu putih mohon ulama lurus dan k212 mmbri hujah 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 , san betapa bahagia nya diri ini saat unboxing paket dan barang nya bagus ! menetapkan beli lagi !	positive
duh . jadi mahasiswa jangan sombong dong . kasih kartu kuning segala . belajar dulu yang baik , tidak usahlah i	negative
makanan beragam , harga makanan di food stall akan ditambahkan 10 % lagi di kasir , suasana ramai dan perlu	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	positive
saya bersama keluarga baru saja menikmati pengalaman kuliner yang menyenangkan di rm sari surda di jalan . bersyukur	positive
simcard indosat inaktiv gara-gara lupa isi pulsa dan kabar nya aktif jika pinda 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 tersebut pengalaman bersama indosat hari ini , semoga tidak terjadi pada pelanggan lain . sempat sampai marah-marah	positive
anak sekarang sulit untuk dinasehati	negative
hanya mengatasi masalah kayak gini saja anies sandi tidak becus . ke mana saja pasukan kebersihan yang begi	negative
setiap bunn dibentuk dengan uu bukan dibentuk dengan uu .	neutral
sehabis puas bermain di trans studio bandung , saya bersama teman sepakat untuk makan ayam kakek di sini .	positive
rasa bakso cuanki dan batagor cukup . selalu ramai pengunjung . buka sekitar jam 10 pagi . dari jalan riau menu	positive
sgwr 2018 beri dampak positif terhadap hobi anak-anak	positive
berada di lembang , pemandangan indah , udara sejuk , penataan resto yang bagus , dilengkapi dengan makan	positive
berbuka puasa di sini pada minggu lepas , pak supir yang mencadangkan tempat makan ini . makanan nya di si	positive
hahaha suka banget nonton kartun indonesia keluarga pak tomat	positive
bagi teman-teman yang sedang berkunjung ke bandung bisa mampir ke kopi anjis untuk makan makanan yang	positive
di restoran ini , saya dan keluarga makan malam yang terakhir di bandung . kami berjumlah 8 orang dan bila ny	positive
alhamdulillah hari ini tidak ke jalan bugel agak kesal juga kalau tiap hari lewat sana melulu . sempit , dua arah , i	negative
respon agak lama , sama sayang nya rem nya bukan asli , masih rom distributor	negative
lokasi dengan pemandangan alam yang masih natural dan indah . suasana juga sejuk dan tenang . hanya saja b	positive
pdip sebut ridwan kamil menang karena berbaju merah	neutral
malu - maluin nih oknum yang tidak bertanggung jawab .	negative
makan siang di sini asyik juga . nuansa bali walaupun music nya musik surda kadang musik khas bali bergantian	negative
tempat steak di bandung sejak lama dari zaman setiabudi sampai sekarang buka cabang di riau . tempat di riau	positive
08:30 : kedatangan presiden ri dan rombongan di gate 2 , terminal 3 disambut oleh menteri perhubungan , mei	neutral
makanan tradisional yang selalu rasanya stabil iga garang asam nya dan java steak nya jangan lupa ya dicoba h	positive
sampai saat ini saya rasa batagor ini masih yang terenak di kota bandung dibandingkan dengan batagor lainnya . positive	positive

Menyiapkan Dataset

Menyeimbangkan data pada dataset agar jumlah sentimen yang digunakan untuk training berada pada nilai yang cukup seimbang



The screenshot shows a Jupyter Notebook interface with several tabs at the top: app.py, train_preprocess.tsv, MLPC.ipynb, and LSTM_Nusa-2.ipynb. The main area displays Python code and its output.

Code and Output (Cell 21):

```
df.columns =['text', 'label']
df
```

[21]

	text	label
0	warung ini dimiliki oleh pengusaha pabrik tahu...	positive
1	lokasi strategis di jalan sumatera bandung . t...	positive
2	betapa bahagia nya diri ini saat unboxing pake...	positive
3	makanan beragam , harga makanan di food stall ...	positive
4	tempat unik , bagus buat foto , makanan enak ,...	positive
...
3343	dapat bonus tcash 12.500 tapi tidak bisa dipak...	negative
3344	ketahuan yang bawain berita ini goblok banget ...	negative
3345	tidak berfaedah	negative
3346	itu ibu-ibu datang kerumah gue setiap tanggal1...	negative
3347	gue setuju buat adu domba jokowi sama megawati...	negative

3348 rows x 2 columns

Code and Output (Cell 23):

```
df.label.value_counts()
```

[23]

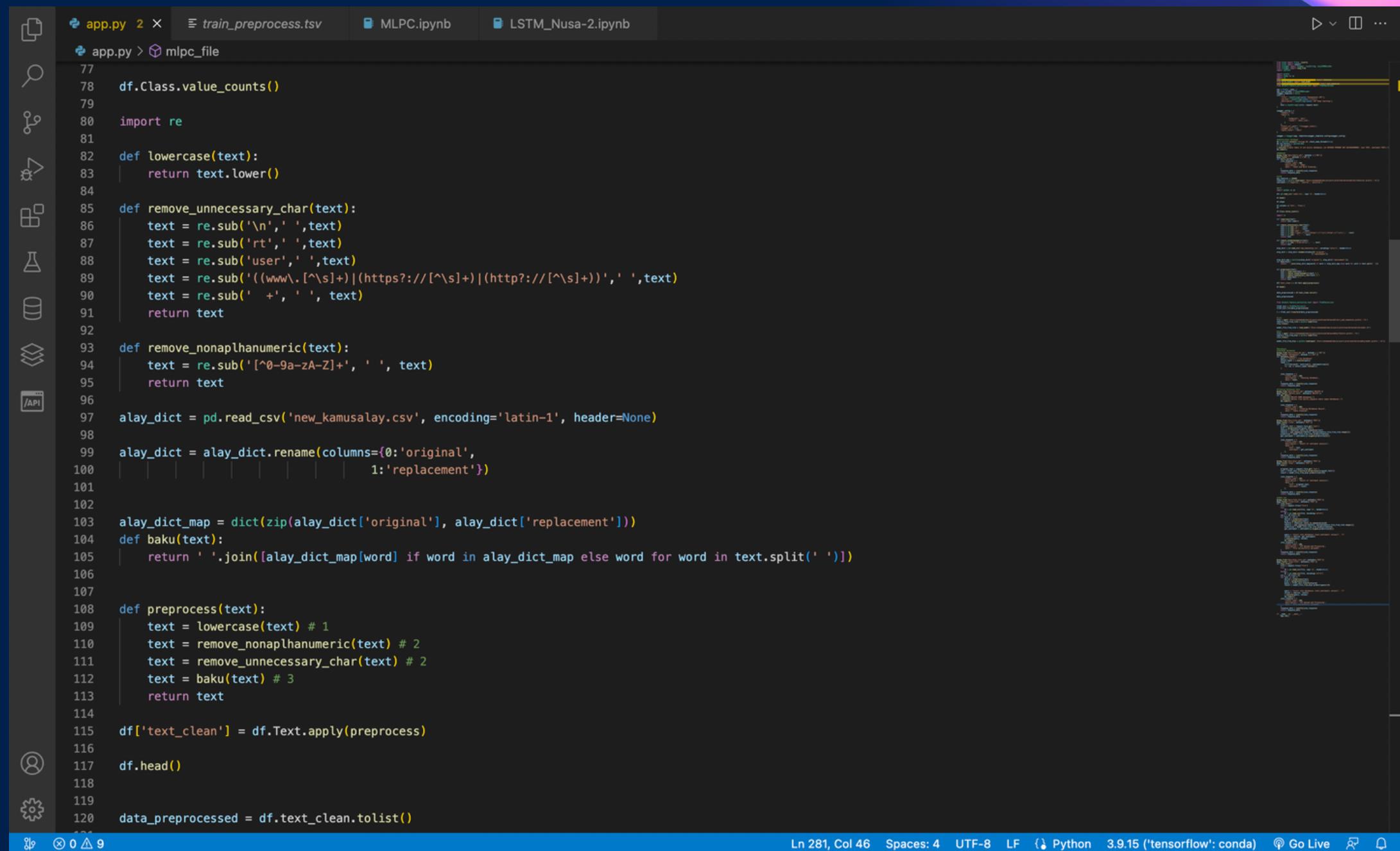
label	count
neutral	1148
positive	1100
negative	1100

Name: label, dtype: int64

Jupyter Server: Local Cell 1 of 16 Go Live

Menyiapkan Dataset

Menormalisasi text pada
Dataset menggunakan
library Regex dan
new_kamusalay.csv agar
tokenization lebih optimal



The screenshot shows a code editor window with a dark theme. The main pane displays Python code for preprocessing a dataset. The code includes imports for pandas and re, defines helper functions for lowercase conversion, character removal, and non-alphanumeric filtering, reads a CSV file for Alay-to-Baku mapping, and applies a multi-step preprocessing pipeline to a DataFrame. The status bar at the bottom indicates the code has 281 lines, 46 columns, 4 spaces, and is in UTF-8 encoding, using Python 3.9.15 ('tensorflow': conda). Other tabs visible in the header include 'train_preprocess.tsv', 'MLPC.ipynb', and 'LSTM_Nusa-2.ipynb'.

```
app.py 2 X  train_preprocess.tsv  MLPC.ipynb  LSTM_Nusa-2.ipynb
app.py > mlpc_file
77
78     df.Class.value_counts()
79
80     import re
81
82     def lowercase(text):
83         return text.lower()
84
85     def remove_unnecessary_char(text):
86         text = re.sub('\n', ' ', text)
87         text = re.sub('rt', ' ', text)
88         text = re.sub('user', ' ', text)
89         text = re.sub('((www\.[^s]+)|(https?://[^s]+)|(http?://[^s]+))', ' ', text)
90         text = re.sub(' +', ' ', text)
91
92         return text
93
94     def remove_nonaplhanumeric(text):
95         text = re.sub('[^0-9a-zA-Z]+', ' ', text)
96
97     return text
98
99     alay_dict = pd.read_csv('new_kamusalay.csv', encoding='latin-1', header=None)
100
101    alay_dict = alay_dict.rename(columns={0:'original',
102                                         1:'replacement'})
103
104    alay_dict_map = dict(zip(alay_dict['original'], alay_dict['replacement']))
105    def baku(text):
106        return ' '.join([alay_dict_map[word] if word in alay_dict_map else word for word in text.split(' ')])
107
108    def preprocess(text):
109        text = lowercase(text) # 1
110        text = remove_nonaplhanumeric(text) # 2
111        text = remove_unnecessary_char(text) # 2
112        text = baku(text) # 3
113
114        return text
115
116    df['text_clean'] = df.Text.apply(preprocess)
117
118    df.head()
119
120    data_preprocessed = df.text_clean.tolist()
```

Ln 281, Col 46 Spaces: 4 UTF-8 LF Python 3.9.15 ('tensorflow': conda) Go Live

Feature Extraction

Pada MLPClassifier, feature extraction dilakukan menggunakan metode TF-IDF

```
from sklearn.feature_extraction.text import TfidfVectorizer  
  
tfidf_vect = TfidfVectorizer()  
tfidf_vect.fit(data_preprocessed)  
  
X = tfidf_vect.transform(data_preprocessed)  
print ("Feature Extraction selesai")
```

Python

Feature Extraction

Pada model LSTM, feature extraction dilakukan menggunakan Tokenizer

```
import pickle
from keras.preprocessing.text import Tokenizer
from keras_preprocessing.sequence import pad_sequences
from collections import defaultdict

max_features = 100000
tokenizer = Tokenizer(num_words=max_features, split=' ', lower=True)
tokenizer.fit_on_texts(total_data)
with open ('tokenizer.pickle', 'wb') as handle:
    pickle.dump(tokenizer, handle, protocol=pickle.HIGHEST_PROTOCOL)
print("tokenizer.pickle has been created!")

X = tokenizer.texts_to_sequences(total_data)

vocab_size = len(tokenizer.word_index)
maxlen = max(len(x) for x in X)

X = pad_sequences(X)
print(X)
with open ('x_pad_sequences.pickle', 'wb') as handle:
    pickle.dump(X, handle, protocol=pickle.HIGHEST_PROTOCOL)
print('x_pad_sequences.pickle has been created!')
```

Python

Training Model

Metode neural network yang digunakan adalah MLPClassifier

The screenshot shows a Jupyter Notebook interface with a dark theme. The top bar displays file tabs for 'app.py' (2), 'train_preprocess.tsv', 'MLPC.ipynb' (current), and 'LSTM_Nusa-2.ipynb'. The toolbar includes icons for file operations, search, and cell execution. The main area contains three code cells:

```
X_train, X_test, y_train, y_test = train_test_split(X, classes, test_size = 0.2)

[16]

from sklearn.neural_network import MLPClassifier

model = MLPClassifier()
model.fit(X_train, y_train)

print ("Training selesai")

[17]

... Training selesai

[18]

pickle.dump(model, open("model.pickle", "wb"))

[19]

from sklearn.metrics import classification_report

test = model.predict(X_test)

print ("Testing selesai")

print(classification_report(y_test, test))

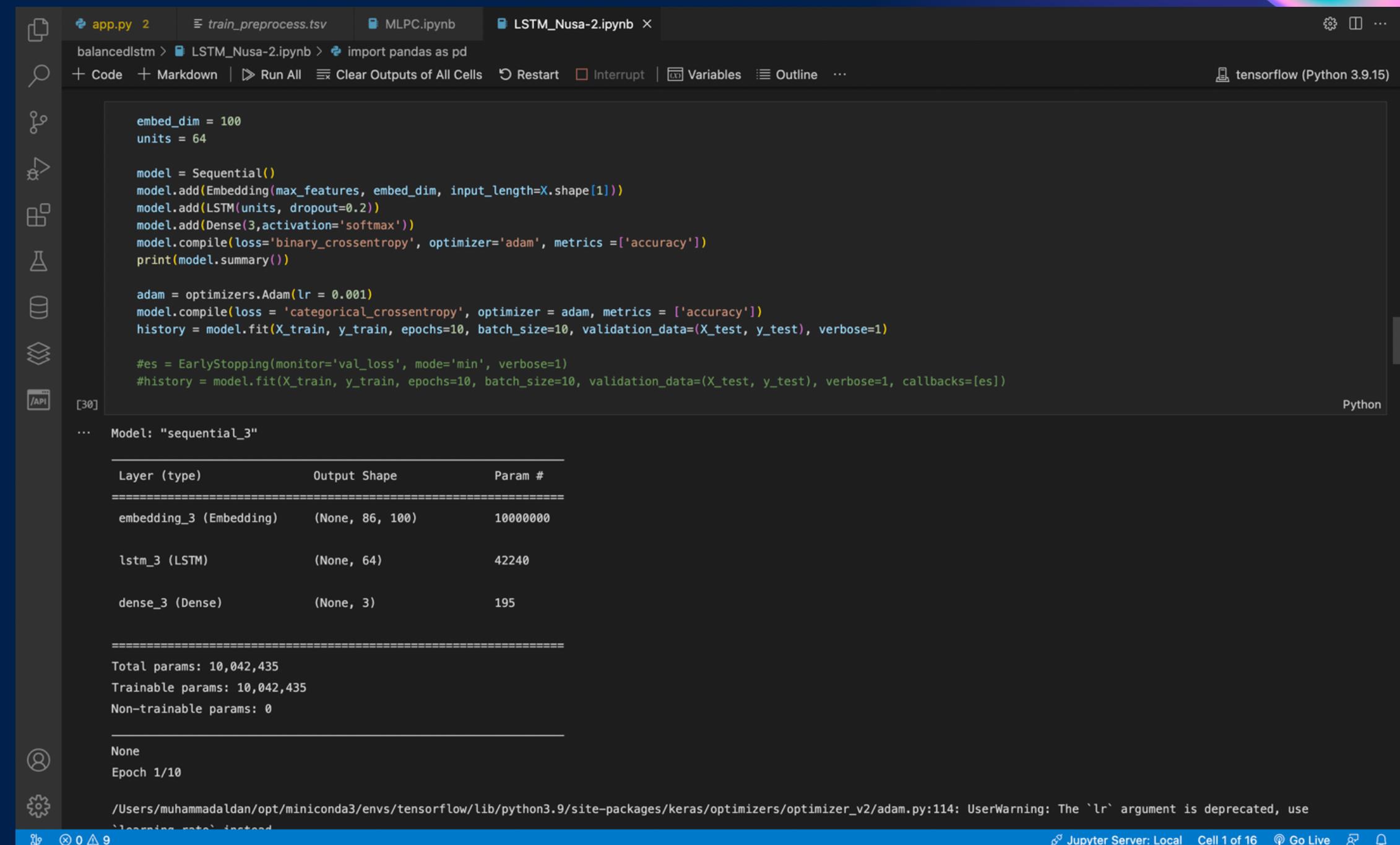
[19]

... Testing selesai
```

Cell [16] contains the command to split the dataset. Cell [17] imports the MLPClassifier from sklearn and fits it to the training data, printing a success message. Cell [18] dumps the trained model to a pickle file. Cell [19] imports classification_report and prints the classification report for the test data, showing metrics for negative, neutral, and positive classes, as well as overall accuracy and weighted avg.

Training Model

Training Model LSTM
dilakukan dengan 3 layer.
Layer embed
(embed_dim=100),LSTM(6
4 units), dan Dense.



The screenshot shows a Jupyter Notebook interface with a dark theme. The top bar displays several open files: app.py, train_preprocess.tsv, MLPC.ipynb, and LSTM_Nusa-2.ipynb. The toolbar includes icons for file operations, search, and various notebook functions. The main area contains Python code for defining and training a sequential model. The code includes imports for pandas, tensorflow, and keras, defines embedding, LSTM, and Dense layers, compiles the model with Adam optimizer and binary_crossentropy loss, and fits it to training data. It also includes a summary of the model's architecture and parameters. A status bar at the bottom shows the number of cells (0), the current cell index (1/16), and other system information.

```
embed_dim = 100
units = 64

model = Sequential()
model.add(Embedding(max_features, embed_dim, input_length=X.shape[1]))
model.add(LSTM(units, dropout=0.2))
model.add(Dense(3, activation='softmax'))
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
print(model.summary())

adam = optimizers.Adam(lr = 0.001)
model.compile(loss = 'categorical_crossentropy', optimizer = adam, metrics = ['accuracy'])
history = model.fit(X_train, y_train, epochs=10, batch_size=10, validation_data=(X_test, y_test), verbose=1)

#es = EarlyStopping(monitor='val_loss', mode='min', verbose=1)
#history = model.fit(X_train, y_train, epochs=10, batch_size=10, validation_data=(X_test, y_test), verbose=1, callbacks=[es])
```

[30]

... Model: "sequential_3"

Layer (type)	Output Shape	Param #
embedding_3 (Embedding)	(None, 86, 100)	10000000
lstm_3 (LSTM)	(None, 64)	42240
dense_3 (Dense)	(None, 3)	195

Total params: 10,042,435
Trainable params: 10,042,435
Non-trainable params: 0

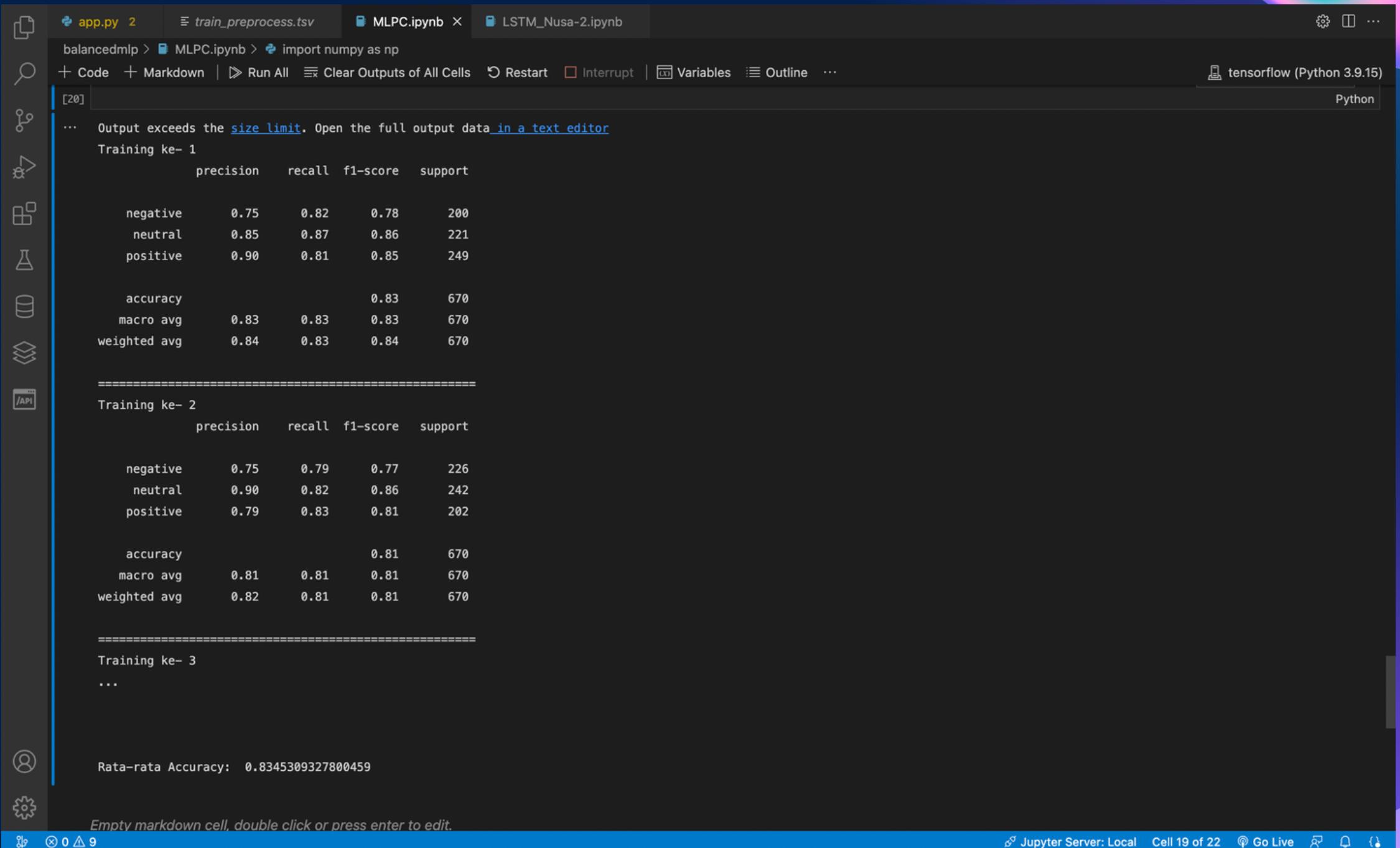
None
Epoch 1/10

/Users/muhammadaldan/opt/miniconda3/envs/tensorflow/lib/python3.9/site-packages/keras/optimizers/optimizer_v2/adam.py:114: UserWarning: The `lr` argument is deprecated, use
`learning_rate` instead.

Jupyter Server: Local Cell 1 of 16 Go Live

Cross Validation

Cross validation dilakukan untuk mengetahui efektivitas model MLP, sehingga dapat diketahui rata-rata akurasi model saat digunakan pada dataset lain.



The screenshot shows a Jupyter Notebook interface with several open files: app.py, train_preprocess.tsv, MLP.ipynb, and LSTM_Nusa-2.ipynb. The current cell (Cell 19 of 22) displays the results of a cross-validation loop. The output shows performance metrics for three training iterations (ke-1, ke-2, ke-3) and a final average accuracy.

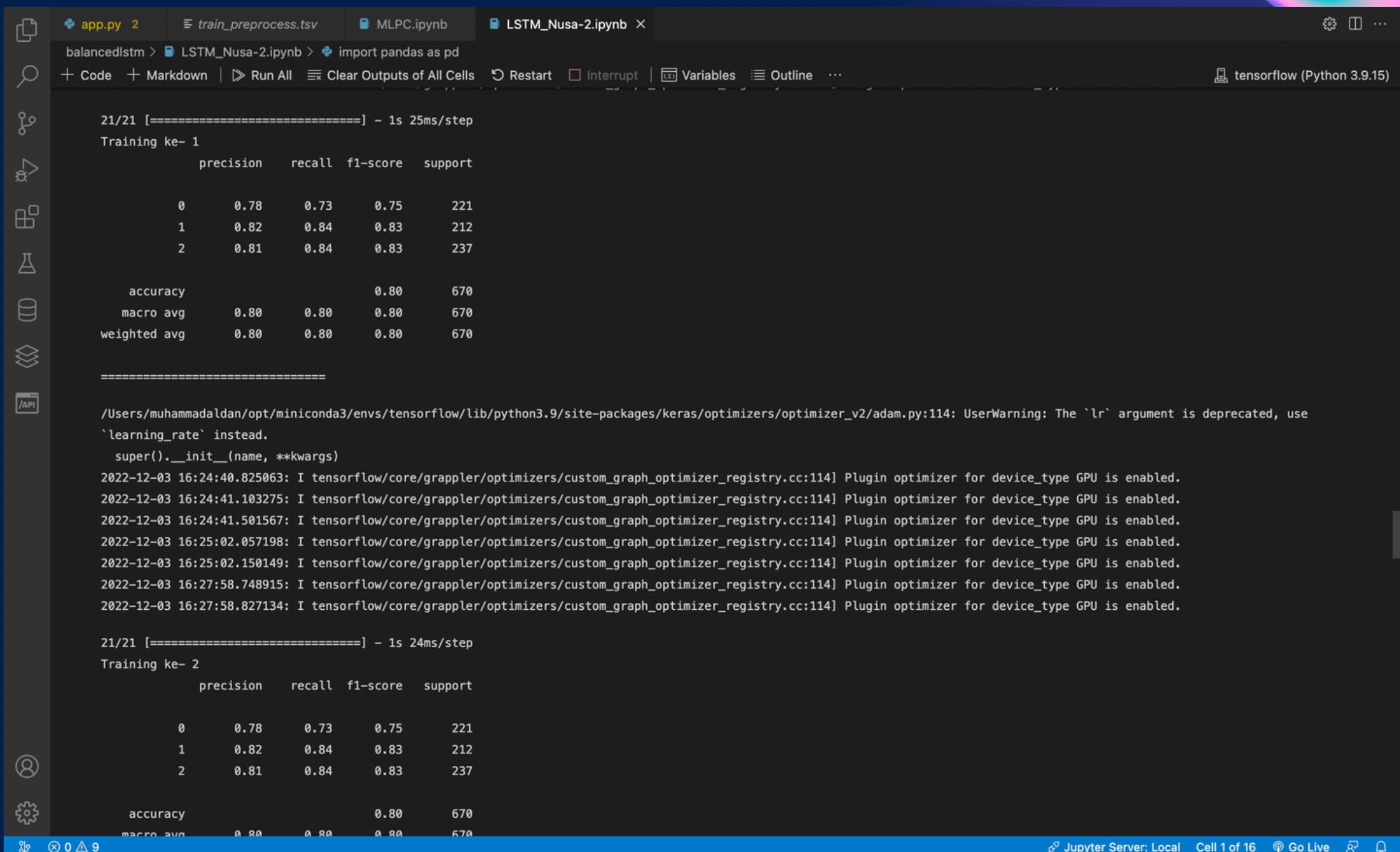
```
... Output exceeds the size limit. Open the full output data in a text editor
Training ke- 1
precision    recall   f1-score  support
negative      0.75     0.82      0.78     200
neutral       0.85     0.87      0.86     221
positive      0.90     0.81      0.85     249
accuracy          -         -        0.83     670
macro avg      0.83     0.83      0.83     670
weighted avg   0.84     0.83      0.84     670
=====
Training ke- 2
precision    recall   f1-score  support
negative      0.75     0.79      0.77     226
neutral       0.90     0.82      0.86     242
positive      0.79     0.83      0.81     202
accuracy          -         -        0.81     670
macro avg      0.81     0.81      0.81     670
weighted avg   0.82     0.81      0.81     670
=====
Training ke- 3
...
Rata-rata Accuracy:  0.8345309327800459
```

Empty markdown cell, double click or press enter to edit.

Jupyter Server: Local Cell 19 of 22 Go Live

Cross Validation

Cross validation dilakukan untuk mengetahui efektivitas model LSTM, sehingga dapat diketahui rata-rata akurasi model saat digunakan pada dataset lain.



The screenshot shows a Jupyter Notebook interface with several open files: `app.py`, `train_preprocess.tsv`, `MLPC.ipynb`, and `LSTM_Nusa-2.ipynb`. The notebook is running on a TensorFlow kernel (Python 3.9.15). The current cell displays the results of a cross-validation loop for an LSTM model. The output shows two training sessions (ke-1 and ke-2) with detailed classification reports and GPU plugin optimizer logs.

Training ke- 1

	precision	recall	f1-score	support
0	0.78	0.73	0.75	221
1	0.82	0.84	0.83	212
2	0.81	0.84	0.83	237
accuracy			0.80	670
macro avg	0.80	0.80	0.80	670
weighted avg	0.80	0.80	0.80	670

=====

```
/Users/muhammadaldan/opt/miniconda3/envs/tensorflow/lib/python3.9/site-packages/keras/optimizers/optimizer_v2/adam.py:114: UserWarning: The `lr` argument is deprecated, use `learning_rate` instead.  
super().__init__(name, **kwargs)  
2022-12-03 16:24:40.825063: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.  
2022-12-03 16:24:41.103275: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.  
2022-12-03 16:24:41.501567: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.  
2022-12-03 16:25:02.057198: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.  
2022-12-03 16:25:02.150149: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.  
2022-12-03 16:27:58.748915: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.  
2022-12-03 16:27:58.827134: I tensorflow/core/grappler/optimizers/custom_graph_optimizer_registry.cc:114] Plugin optimizer for device_type GPU is enabled.
```

Training ke- 2

	precision	recall	f1-score	support
0	0.78	0.73	0.75	221
1	0.82	0.84	0.83	212
2	0.81	0.84	0.83	237
accuracy			0.80	670
macro avg	0.80	0.80	0.80	670
weighted avg	0.80	0.80	0.80	670

Membuat API

Terdapat 4 endpoint utama dalam API. 2 endpoint yang dapat mendeteksi sentimen dalam teks menggunakan model LSTM dan juga NN. Serta terdapat 2 endpoint yang dapat mendeteksi sentimen menggunakan input file .csv. Sedangkan endpoint database digunakan untuk menampilkan data yang sudah diproses pada File Sentiment Analysis

The screenshot shows a web browser window displaying the API documentation for 'API Deep learning'. The title bar indicates the URL is 127.0.0.1. The main content area is titled 'Dokumentasi API 1.0.0' and includes the following sections:

- Hello World**:
 - GET /
- Database Checks**:
 - GET /databases
 - DELETE /delete_data
- Text sentiment analysis**:
 - POST /lstm
 - POST /mlpc
- File sentiment analysis**:
 - POST /lstm-file
 - POST /mlpc-file

At the bottom right of the page, there is a small note: '[Powered by Flasgger 0.9.5]'

Thank You

@aldan_ha

muhammadaldan@gmail.com

<https://github.com/aldanha>

Full project documentation can be accessed on : <https://github.com/aldanha/binarplatinum.git>