

Analisis Sentiment

Analysist Data & Building API

By : Reni Kartika Sari & Putri Ratna Sari



Pendahuluan



Berdasarkan laporan We Are Social mencatat, jumlah pengguna internet di Indonesia telah mencapai 212,9 juta pada Januari 2023. Ini berarti sekitar 77% dari populasi Indonesia telah menggunakan internet. Melihat trennya, jumlah pengguna internet di Indonesia terus tumbuh setiap tahun. Adapun, lonjakan pengguna internet di dalam negeri terjadi pada 2017.

Dengan tingginya tren tersebut, memungkinkan para pengguna internet mengakses berbagai platform sosial media. Sebagian besar orang Indonesia banyak menggunakan sosial media dalam berbagai hal baik hanya sekedar *Scrolling* melihat gambar, video ataupun berkomentar. Contoh social media yang banyak digunakan di kalangan ada Facebook, Instagram, Youtube, Twitter, Tiktok, Whatsapp dan lain-lain

Analisis sentimen adalah proses menganalisis teks digital untuk menentukan apakah nada emosional pesan tersebut positif, negatif, atau netral. Saat ini, perusahaan memiliki data teks dalam volume besar seperti email, transkrip obrolan dukungan pelanggan, komentar media sosial, dan ulasan. Alat analisis sentimen dapat memindai teks ini untuk secara otomatis menentukan sikap penulis terhadap suatu topik. Perusahaan menggunakan wawasan dari analisis sentimen untuk meningkatkan mutu layanan pelanggan dan meningkatkan reputasi merek.

Seberapa Penting?

1. Memberikan Wawasan yang Objective
2. Membangun layanan yang baik
3. Menganalisis dalam skala yg besar
4. Hasil waktu yang nyata

Tantangan

1. Sarkasme
2. Negasi adalah penggunaan kata negative untuk menyampaikan pembalikan makna
3. Multipolaritas terjadi ketika sebuah kalimat mengandung lebih dari satu sentimen

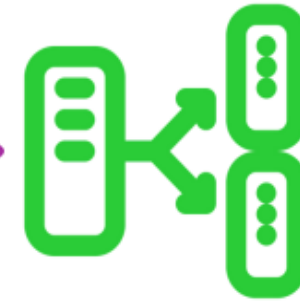
Metode Penelitian



Feature-Label
Classification



Feature
Extraction



Train-Test Data
Split

```
# Oversampling neutral label to be equal with negative label

data_pos = data[data['Label']=='positive']
data_neu = data[data['Label']=='neutral']
data_neg = data[data['Label']=='negative']

df_neu = data_neu.sample(data_neg.shape[0], replace = True)

data = pd.concat([data_pos, df_neu, data_neg])

# Group data tweet

pos_tweet = data.loc[data['Label']=='positive'].Tweet.tolist()
neg_tweet = data.loc[data['Label']=='negative'].Tweet.tolist()
neu_tweet = data.loc[data['Label']=='neutral'].Tweet.tolist()

# Group data label

pos_label = data.loc[data['Label']=='positive'].Label.tolist()
neg_label = data.loc[data['Label']=='negative'].Label.tolist()
neu_label = data.loc[data['Label']=='neutral'].Label.tolist()
```

```
# Tokenizer and pad sequence

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

```
# Split train-test data with proportion 80:20

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

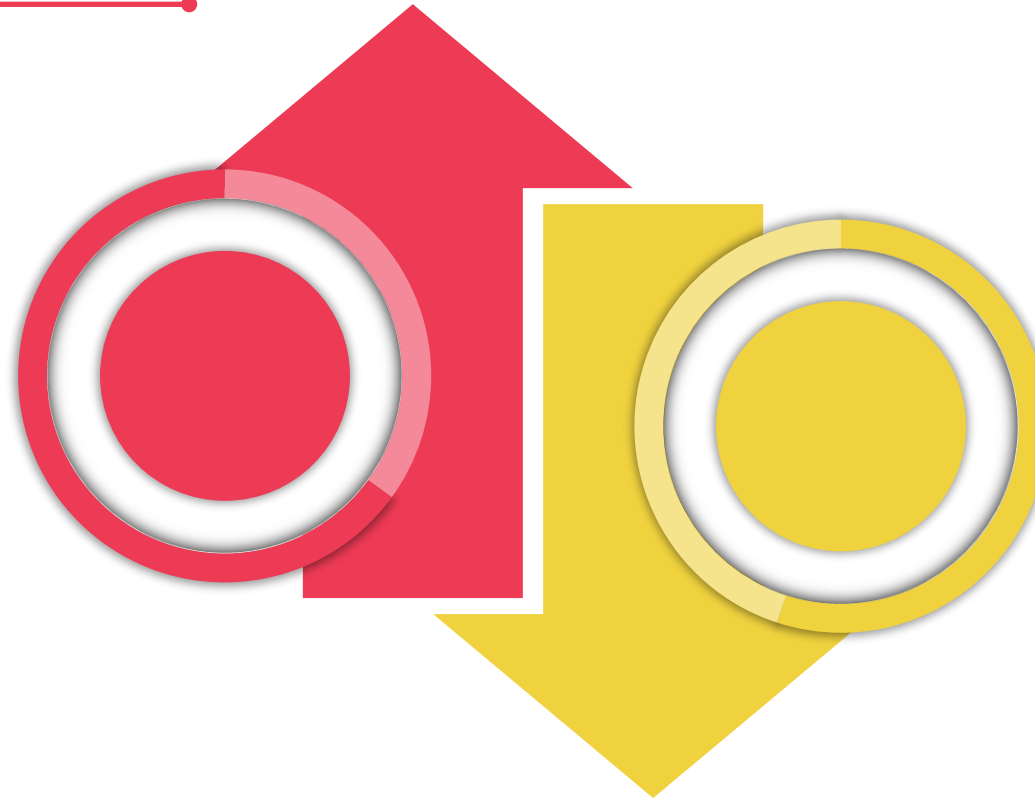
file = open("y_labels.pickle", 'rb')
Y = pickle.load(file)
file.close()

X_train, X_test, y_train, y_test = train_test_split(
    X,
    Y,
    test_size=0.2,
    random_state=42)
```

Analysis Sentiment

Neural Network

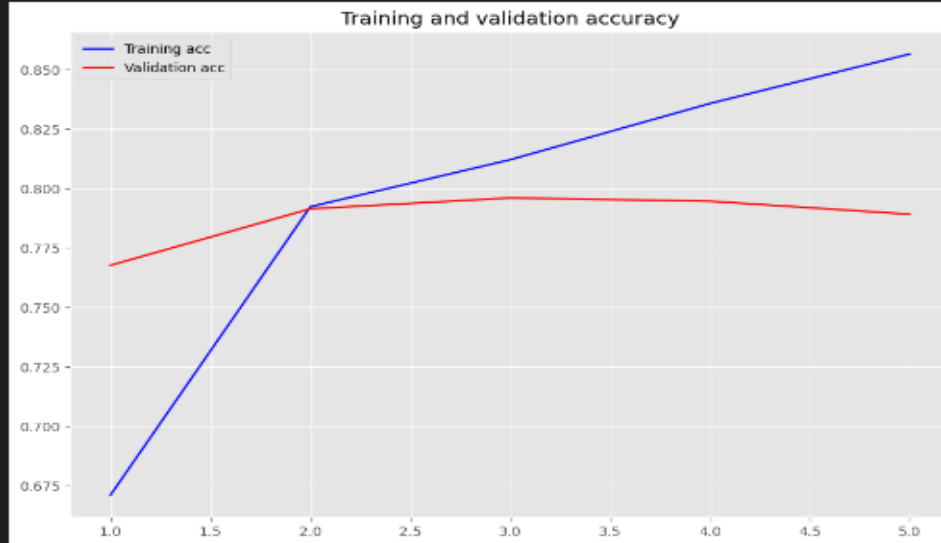
Dirancang untuk memetakan data input ke data output tanpa mempertimbangkan urutan atau urutan titik data apa pun



Model LSTM

Sistem penyimpanan data yang dapat memproses, memprediksi, dan mengklasifikasikan informasi yang telah disimpan dalam jangka waktu lama sekali pun. LSTM hadir sebagai bentuk modifikasi dari *recurrent neural network* atau RNN dan termasuk sebagai salah satu yang populer..

Result Model Neural Network



Setelah 5 kali training dengan epoch mayoritas berhenti di epoch ke 5 dan 6, didapatkan rata-rata akurasi model senilai 79.1%

```
1/1 [=====] - 0s 124ms/step  
Text:  rasanya enak  
Sentiment: positive
```

```
1/1 [=====] - 0s 148ms/step  
Text:  rasanya aneh  
Sentiment: negative
```

Result API

Sentiment Analysis using RNN

POST /rnn

Parameters

Cancel

Name	Description
text * required string (formData)	<input type="text" value="MAKANAN ENAK2"/>

Execute

Clear

Responses

Response content type application/json

Curl

```
curl -X POST "http://127.0.0.1:5000/rnn" -H "accept: application/json" -H "Content-Type: application/x-www-form-urlencoded" -d "text=MAKANAN ENAK2"
```

Request URL

http://127.0.0.1:5000/rnn

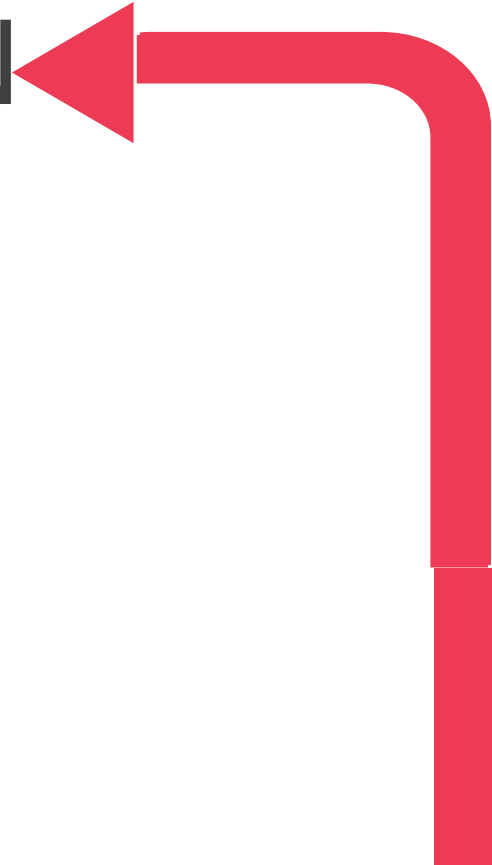
Server response

Code	Details
200	<div><p>Response body</p><pre>{ "data": { "cleaned_text": "makan enak2", "sentiment": "positive", "text": "MAKANAN ENAK2" }, "description": "Results of RNN model", "status_code": 200 }</pre><p>Download</p></div>

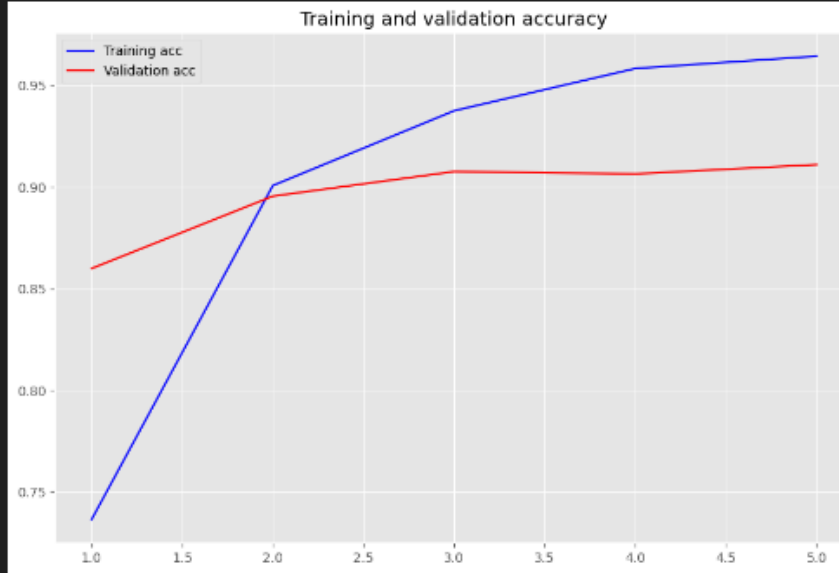
Response headers

```
connection: close  
content-length: 141  
content-type: application/json  
date: Fri, 18 Aug 2023 11:32:16 GMT  
server: Werkzeug/2.2.2 Python/3.10.7
```

RNN



Result Model LSTM



Setelah 5 kali training dengan epoch masing-masing berhenti di epoch ke 5, didapatkan rata-rata akurasi model senilai 96.1%

```
1/1 [=====] - 0s 266ms/step
Text:  rasanya enak
Sentiment: positive
```

```
1/1 [=====] - 0s 253ms/step
Text:  standarisasi makananan
Sentiment: neutral
```

Result API

LSTM

Sentiment Analysis using LSTM

POST

/lstm

Parameters

Cancel

Name	Description
text * required string (formData)	<input type="text" value="MAKANAN ENAK2"/>

Execute

Clear

Responses

Response content type application/json

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=MAKANAN ENAK2"

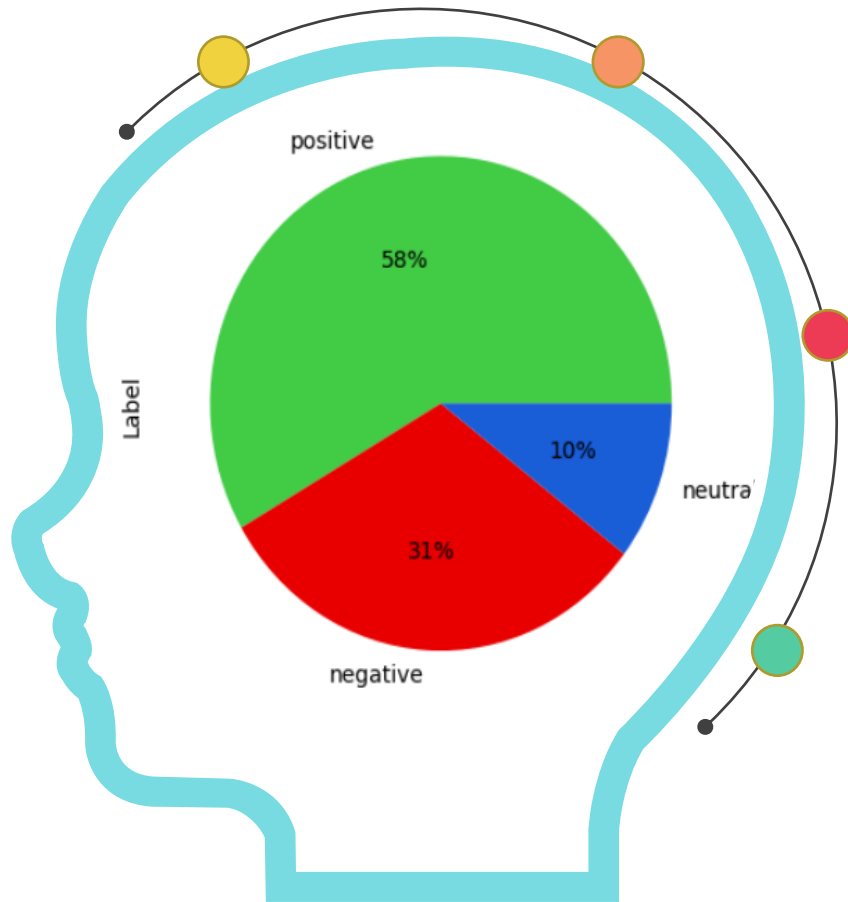
Request URL

http://127.0.0.1:5000/lstm

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "data": { "cleaned_text": "makan enak2", "sentiment": "positive", "text": "MAKANAN ENAK2" }, "description": "Results of LSTM model", "status_code": 200 }</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>connection: close content-length: 142 content-type: application/json date: Fri, 18 Aug 2023 11:31:31 GMT server: Werkzeug/2.2.2 Python/3.10.7</pre></div></div>

Hasil



Data Set ini terdiri dari 58% Positif Tweet, 31% Negative Tweet dan 10% Neutral Tweet.



Model LSTM dapat mengklasifikasi Neutral dan Positive Tweet namun tidak untuk Negative Tweet



Model NN dapat mengklasifikasi Negative dan Positive Tweet. Namun tidak untuk Neutral



Untuk API hanya dapat mengklasifikasikan Positive Tweet

Kesimpulan

Berdasarkan hasil pengujian dataset yang diberikan, dapat diambil kesimpulan bahwa mayoritas isi tweet berisikan kalimat yang cenderung positif.

Pengujian menggunakan metode model LSTM dan Neural Network cukup efektif dalam mengklasifikasikan isi tweet. Namun penting untuk dipahami bahwa hasil dari klasifikasi tweet dapat menghasilkan nilai yang salah karena perbedaan yang cukup signifikan dari jumlah label (positif, negative, netral) yang ada pada dataset.

Untuk mengetahui apakah model sudah berjalan dengan baik atau tidaknya dapat dilihat dari hasil visualisasinya. Visualisasi untuk Training and Validation Accuracy yang baik adalah yang selalu naik di tiap epoch nya. Sedangkan untuk Training and Validation Loss selalu turun di tiap epochnya.





THANK YOU