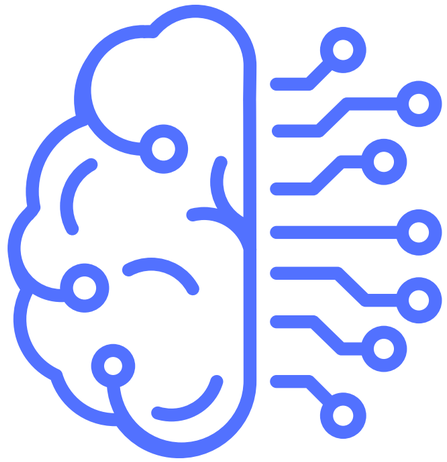


PROJECT 1: SENTIMENT ANALYSIS

NLP GROUP A - KANG AI

NLP GROUP A -
KANG AI

TEAM MEMBER

Ade

Dede

Daffa

NLP GROUP A -
KANG AI

BACKGROUND

As the political season approaches, public opinions toward presidential and vice-presidential candidates have become a major discussion topic on social media.

People express their views through comments, tweets, and posts reflecting support, criticism, or neutrality.
Due to the large volume and diversity of social media texts, manual analysis is inefficient.

Therefore, Natural Language Processing (NLP) can be leveraged to perform sentiment analysis automatically.

This project aims to develop a sentiment analysis model to assess public sentiment toward presidential candidates, while experimenting with both traditional and deep learning algorithms to identify the best-performing approach for Indonesian political text data.

PROBLEM STATEMENT

The main problem addressed is how to build an NLP model capable of:

1. Automatically classifying public sentiment toward presidential candidates (positive, negative, neutral).
2. Handling informal, mixed-language text commonly used on social media.
3. Determining which algorithm performs best across evaluation metrics.

NLP GROUP A -
KANG AI

OBJECTIVES

- Develop a sentiment analysis model for Indonesian political text.
- Compare traditional ML models (Naïve Bayes) with deep learning models (LSTM, BERT).
- Evaluate model performance using accuracy, precision, recall, and F1-score.
- Identify the best algorithm for this task.
- Publish project results and code on GitHub.

SCOPE

Focus: Sentiment classification of public opinion toward presidential candidates.

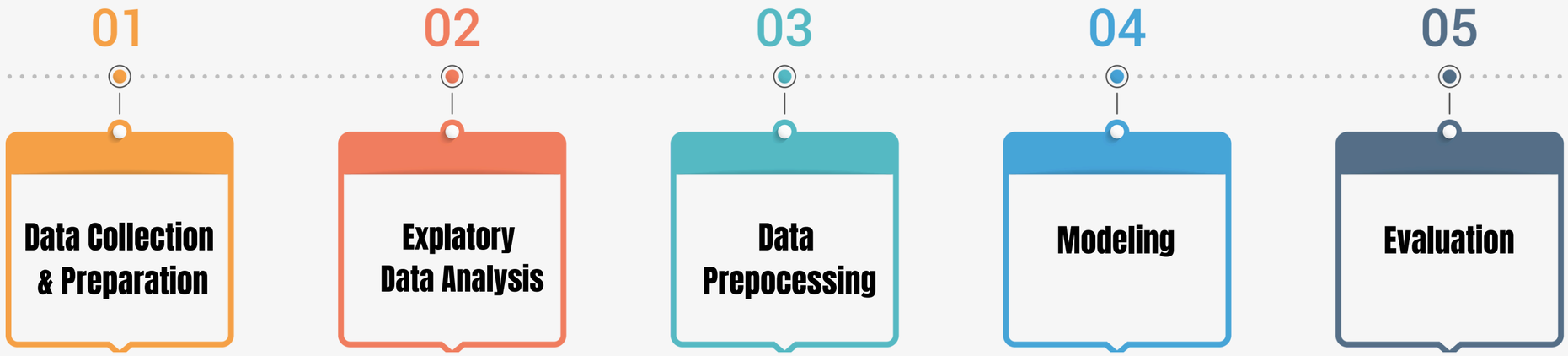
Project Steps: Data collection, preprocessing, EDA, model experimentation, evaluation..

Start Date: 26 September 2025
Finish Date: 10 October 2025

Limitations:
Only sentiment classification (positive, negative, neutral).
Excludes topic analysis or fake news detection.

NLP GROUP A
- KANG AI

PROJECT CYCLE



NLP GROUP A -
KANG AI

DATA COLLECTION & PREPARATION

Data Source:
Dataset “tweet.csv” disediakan oleh Indonesia AI. Data ini adalah hasil web scraping dari platform x.com terkait dengan Pilpres tahun 2019.

Deskripsi variabel utama:

- Tweet→ kalimat tweet yang dipublikasi
- Sentimen → kategori sentimen dari tweet

Penelitian berfokus pada prediksi **sentimen** dari sebuah tweet yang dipublikasi oleh user.

Import Library and Data Set

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.feature_extraction.text import CountVectorizer


df = pd.read_csv
 ("drive/MyDrive/Project NLP/Sentiment Analysis/Dataset/tweet.csv")
```

NLP GROUP A - KANG AI

EXPLORATORY DATA ANALYSIS

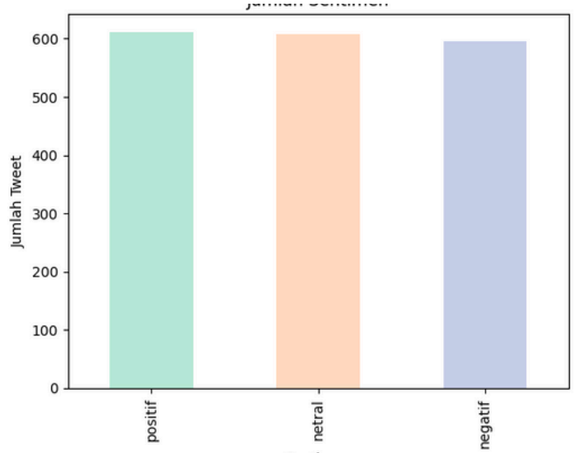
1 DATASET INFORMATION

DF.INFO()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1815 entries, 0 to 1814
Data columns (total 3 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Unnamed: 0  1815 non-null   int64 
 1   sentimen    1815 non-null   object
 2   tweet       1815 non-null   object
dtypes: int64(1), object(2)
memory usage: 42.7+ KB
```

2. JENIS SENTIMEN

```
DF["SENTIMEN"].VALUE_COUNTS().PLOT(KIND="BAR",
COLOR=["#B5EAD7", "#FFDAC1", "#C7CEEA"])
PLT.TITLE("JUMLAH SENTIMEN")
PLT.XLABEL("SENTIMEN")
PLT.YLABEL("JUMLAH TWEET")
PLT.SHOW()
```



3. MISSING AND DUPLICATE VALUE

```
df.isnull().sum()
```

	0
Unnamed: 0	0
sentimen	0
tweet	0

```
df[df.duplicated()]
```

	Unnamed: 0	sentimen	tweet

NLP GROUP A -
KANG AI

DATA PREPROCESSING

TAHAP PREPOCESSING TERHADAP DATASET YANG DIAMBIL DARI PLATFORM MEDIA SOSIAL (TERUTAMA TWITTER) DILAKUKAN SECARA HATI-HATI DAN MENDETAIL. DIKARENAKAN:

1. PLATFORM MEDIA SOSIAL MEMBUAT USER BISA LEBIH BANYAK **BEREKSPRESI SECARA BEBAS**.

2. PENGGUNAAN GAYA BAHASA YANG DIDALAMNYA BANYAK **SLANG**, BAHASA **CAMPURAN** INDONESIA-INGGRIS (ATAU CAMPUR BAHASA DAERAH), SPAM PENGGUNAAN **EMOJI**, **TANDA BACA** YANG TIDAK FUNGSIONAL YANG MEREPRESENTASIKAN DARI GAYA MENGETIK INDIVIDUAL MASING-MASING USER.

3. **EMOTICON**/ **EMOJI MEREPRESENTASIKAN** ISI HATI USER SAAT BERINTERAKSI DENGAN USER LAINNYA DI MEDIA SOSIAL.
TERLEBIH SAAT MENGANALISIS MODEL NLP UNTUK 'SENTIMENT ANALYSIS', EMOSI USER SANGAT MENENTUKAN KATEGORI SENTIMEN, APAKAH POSITIF/ NETRAL/ NEGATIF.

4. TERDAPAT BEBERAPA **NOISE LAINNYA** YANG TERDETEKSI DI DALAM DATASET, SEPERTI PENGGUNAAN TANDA **MENTION** DAN **HASHTAG**, **LINK** YANG HARUS DIBERSIHKAN.
TIDAK KALAH PENTINGNYA, DALAM DATASET JUGA TERDAPAT EMOJI YANG **MIS-DECODING** (MOJIBAKE/ ENCODING MIX-UPS) SEPERTI "Ã°Å¸Â€˜Â‍Ã°Å¸Â€˜Â‍Ã°Å¸Â€˜Â‍" YANG PERLU DIPERBAIKI MENGGUNAKAN LIBRARY PYTHON [FTFY] (HTTPS://PYPI.ORG/PROJECT/FTFY/).
TIDAK SAMPAI DI SITU, MOJIBAKE KEMUDIAN DIKONVERSI KE KATA MENGGUNAKAN LIBRARY PYTHON [DEMOJI] (HTTPS://PYPI.ORG/PROJECT/DEMOJI/) UNTUK MENORMALISASIKAN TEKS DAN MEMPERTAHANKAN SEMANTIK KOMUNIKASI USER SAAT PROSES TOKENISASI.

5. **PENINJAUAN ULANG** SAAT PROSES MENGHILANGKAN **STOPWORDS**. KARENA DALAM NLP BAHASA INDONESIA, **KATA PENYANGKALAN** SEPERTI 'TIDAK' ATAU 'BUKAN' TERMASUK KE DALAM KATEGORI STOPWORDS NAMUN FATAL APABILA SECARA TIDAK SADAR TEREKSEKUSI SAAT PROSES CLEANING. DIBUTUHKAN CUSTOM STOPWORDS UNTUK MENGECUALIKAN KATA-KATA PENYANGKALAN TSB (TERMASUK GAYA PENULISAN SLANG-NYA).

NLP GROUP A -
KANG AI

DATA PREPROCESSING

DATA CLEANSING

LOWERCASING
TOKENIZING
REMOVE STOPWORD
REMOVE NOISE
(PUNCTUATION,
SPECIAL CHAR)
ENCODING EMOJI
VECTORIZER

```python
#text cleaning
def clean_text(text):
  text = ftfy.fix_text(text)
  text = emoji.demojize(text)
  text = text.lower()
  text = re.sub(r"http\S+", " URL ", text)
  text = re.sub(r"^RT[\s]+", "", text)
  text = re.sub(r"^(@\w+\s*)+", "", text)
  text = re.sub(r"#(\w+)", r"\1", text)
  text = re.sub(r"@\w+", " USER ", text)
  text = re.sub(r"[^\w\s]+", " ", text)
  text = re.sub(r"[_]+", " ", text)
  text = re.sub(r"\b\d+\b", " ", text)
  text = re.sub(r"\s+", " ", text).strip()
  return text

vectorizer = CountVectorizer()

df["cleaned"]=df["tweet"].apply(clean_text)
vectorizer.fit(df["cleaned"])
vectorizer.get_feature_names_out()[:30]
```

```python
#hapus stopwords
stop_words = set(stopwords.words("indonesian"))
custom_words = list
  (stop_words - {"tidak", "bukan", "belum", "gak", "gk", "blm", "tdk", "g"})

def remove_stopwords(text):
  tokens = word_tokenize(text)
  filtered_words = [word for word in tokens if word not in custom_words]
  return " ".join(filtered_words)
```

NLP GROUP A - KANG AI

DATA PREPROCESSING

STEMMING

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

factory = StemmerFactory()
stemmer = factory.create_stemmer()

df['stemmed'] = df['cleaned'].apply(lambda x: stemmer.stem(x))
```

```
df[["tweet", "stemmed"]].head(10)
```

	tweet	stemmed
0	Kata @prabowo Indonesia tidak dihargai bangsa ...	kata user indonesia tidak harga bangsa asing b...
1	Batuan Langka, Tasbih Jokowi Hadiah dari Habib...	batu langka tasbih jokowi hadiah dari habib lu...
2	Di era Jokowi, ekonomi Indonesia semakin baik....	di era jokowi ekonomi indonesia makin baik 01i...
3	Bagi Sumatera Selatan, Asian Games berdampak p...	bagi sumatera selatan asi games dampak pd ekon...
4	Negara kita ngutang buat bngun infrastruktur y...	negara kita ngutang buat bngun infrastruktur y...
5	Yg bisikin pak jokowi, cm mikirin perputaran d...	yg bisikin pak jokowi cm mikirin putar duit di...
6	Masa tenang msih ngoceh aja..ttp jokowi harga ...	masa tenang msih ngoceh aja ttp jokowi harga mati
7	#UASdifitnahKejiBalasDiTPS kerjasa ekonomi b...	uasdifitnahkejibalasditps kerjasa ekonomi bila...
8	Iya bener Aa, kita MANTAP kan pilihan ke Pemim...	iya bener aa kita mantap kan pilih ke pimpin y...
9	Prabowo-Sandi Sepakat Tak Ambil Gaji karena Ne...	prabowo sandi sepakat tak ambil gaji karena ne...

```
before = len(set(" ".join(df['cleaned']).split()))
after = len(set(" ".join(df['stemmed']).split()))
print(f"Kata unik sebelum: {before}, sesudah stemming: {after}")
```

```
Kata unik sebelum: 7486, sesudah stemming: 5719
```

NLP GROUP A - KANG AI

DATA PREPROCESSING

COMPARISON OF RAW, CLEANSED AND STEMMED DATA

Perbandingan data raw, cleaned, stemmed

```
1 df.sample(5)[["tweet", "cleaned", "stemmed"]]
```

	tweet	cleaned	stemmed
792	Trimakasih @KemenPU Trimakasih @tni_ad Trima...	trimakasih USER trimakasih USER trimakasih USE...	trimakasih user trimakasih user trimakasih use...
1258	Pemimpin yang sudah selesai dengan dirinya, su...	pemimpin yang sudah selesai dengan dirinya sud...	pimpin yang sudah selesai dengan diri sudah ka...
1375	Prabowo Sebut Ekonomi Indonesia Salah Arah Kes...	prabowo sebut ekonomi indonesia salah arah kes...	prabowo sebut ekonomi indonesia salah arah sal...
199	Di bawah perintah pak @jokowi bangunkan SPBU p...	di bawah perintah pak USER bangunkan spbu pert...	di bawah perintah pak user bangun spbu pertama...
1767	Wow ga ngambil gaji sedikitpun kalo jadi presi...	wow ga ngambil gaji sedikitpun kalo jadi presi...	wow ga ngambil gaji sedikit kalo jadi presiden...

NLP GROUP A - KANG AI

MODELING

TRADITIONAL ML ALGORITHM : NAIVE BAYES

Dua Teknik Representasi teks:

- Count Vectorizer → Ambil kata & frekuensinya
- TF-IDF Vectorizer → Ambil kata & bobotnya

	word	count
4754	ekonomi	818
15326	prabowo	578
20449	user	559
8465	jokowi	538
21273	yg	528
5678	gaji	444
20338	url	381
6745	harga	357
16814	sandi	328
15599	presiden	309

	Kata	Bobot_TFIDF
454	ekonomi	0.044825
1518	prabowo	0.038401
858	jokowi	0.035490
2085	user	0.034761
569	gaji	0.033444
2183	yg	0.032077
2072	url	0.031578
682	harga	0.028645
1700	sandi	0.027204
1554	presiden	0.023632

NLP GROUP A - KANG AI

MODELING

TRADITIONAL ML ALGORITHM : NAIVE BAYES

TRAIN

-- CountVectorizer --
Accuracy: 0.9511

	precision	recall	f1-score	support
negatif	0.97	0.95	0.96	481
netral	0.93	0.95	0.94	489
positif	0.95	0.95	0.95	482
accuracy			0.95	1452
macro avg	0.95	0.95	0.95	1452
weighted avg	0.95	0.95	0.95	1452

TEST

-- CountVectorizer --
Accuracy: 0.6446

	precision	recall	f1-score	support
negatif	0.63	0.68	0.65	115
netral	0.70	0.62	0.65	118
positif	0.62	0.64	0.63	130
accuracy			0.64	363
macro avg	0.65	0.65	0.65	363
weighted avg	0.65	0.64	0.64	363

NLP GROUP A - KANG AI

MODELING

TRADITIONAL ML ALGORITHM : NAIVE BAYES

TRAIN

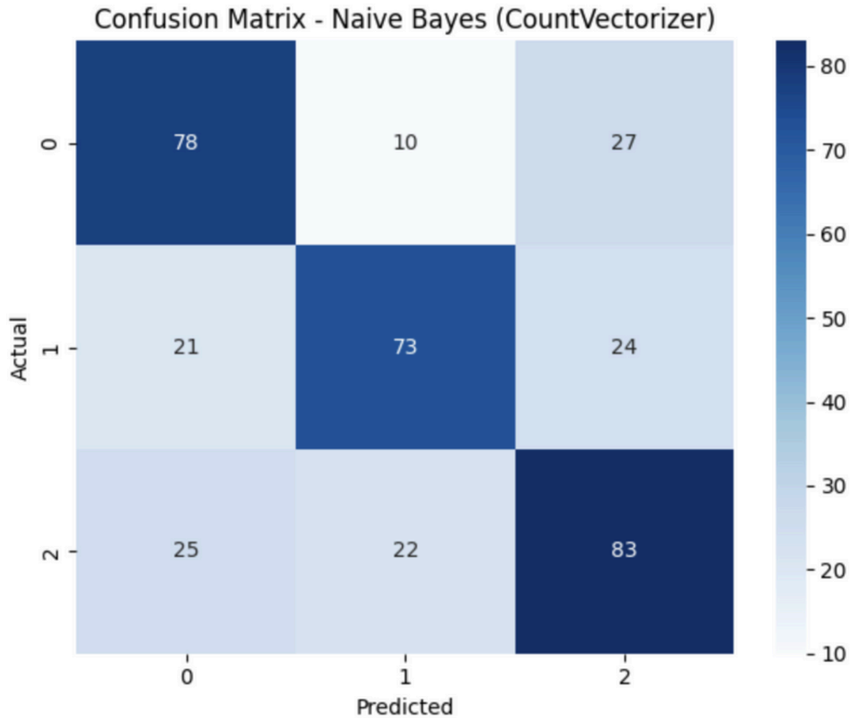
```
-- TF-IDF --
Accuracy: 0.7982
              precision    recall  f1-score   support

     negatif       0.75      0.89      0.82       481
      netral       0.83      0.75      0.79       489
     positif       0.83      0.75      0.79       482

    accuracy                           0.80      1452
   macro avg       0.80      0.80      0.80      1452
weighted avg       0.80      0.80      0.80      1452
```

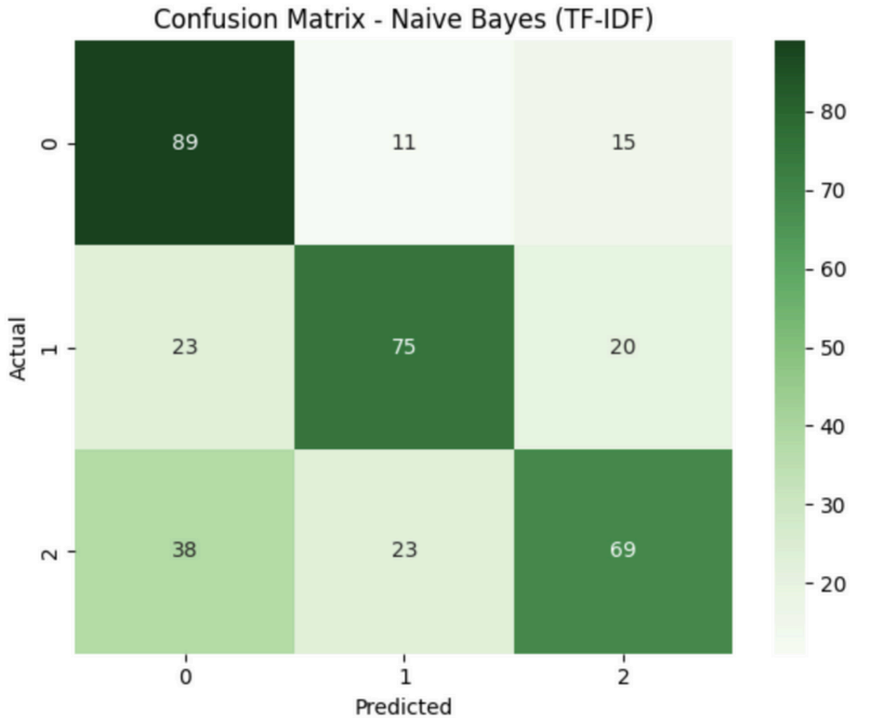
TEST

```
-- TF-IDF --
Accuracy: 0.6419
              precision    recall  f1-score   support

     negatif       0.59      0.77      0.67       115
      netral       0.69      0.64      0.66       118
     positif       0.66      0.53      0.59       130

    accuracy                           0.64       363
   macro avg       0.65      0.65      0.64       363
weighted avg       0.65      0.64      0.64       363
```

NLP GROUP A
- KANG AI

HEATMAP CONFUSION MATRIX





NLP GROUP A -
KANG AI

MODELING

DEEP LEARNING ALGORITHM : LSTM

1) DATA SPLIT - TRAINING AND TEST

```python
from keras.models import Sequential
from keras.layers import Dense, Embedding, LSTM, Bidirectional, Dropout
from keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.preprocessing.text import Tokenizer
from sklearn.model_selection import train_test_split
from keras.callbacks import EarlyStopping
from keras.regularizers import l2, Regularizer
from sklearn.metrics import accuracy_score, classification_report
from keras.layers import BatchNormalization

labels = df['sentimen'].map({'negatif': 0, 'netral': 1, 'positif': 2}).values

#split data - use the 'labels' variable which contains integer mappings
stemmed_text = df['stemmed'].astype(str).values

tokenizer = Tokenizer(num_words=5000)
tokenizer.fit_on_texts(stemmed_text)

#ubah teks jadi sequences
sequences = tokenizer.texts_to_sequences(stemmed_text)

#padding
padded = pad_sequences(sequences, maxlen=100)

x_raw_train, x_raw_test, y_train, y_test = train_test_split(padded, labels, test_size=0.2, random_state=42)
```

✓ 7.0s Python

NLP GROUP A -
KANG AI

MODELING

2) MODEL SUMMARY

DEEP LEARNING ALGORITHM : LSTM

Model: "sequential"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 100, 32)	160,000
bidirectional (Bidirectional)	(None, 64)	16,640
dense (Dense)	(None, 64)	4,160
dropout (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 3)	195

Total params: 180,995 (707.01 KB)

Trainable params: 180,995 (707.01 KB)

Non-trainable params: 0 (0.00 B)

NLP GROUP
A - KANG AI

MODELING

DEEP LEARNING ALGORITHM : LSTM

2) MODEL SUMMARY WITH WORD2VEC

Model: "sequential_1"

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 100, 100)	572,000
bidirectional_1 (Bidirectional)	(None, 64)	34,048
dense_2 (Dense)	(None, 64)	4,160
dropout_1 (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 3)	195

Total params: 610,403 (2.33 MB)

Trainable params: 610,403 (2.33 MB)

Non-trainable params: 0 (0.00 B)

NLP GROUP A - KANG AI

MODELING

DEEP LEARNING ALGORITHM : LSTM

3) MODEL EVALUATION

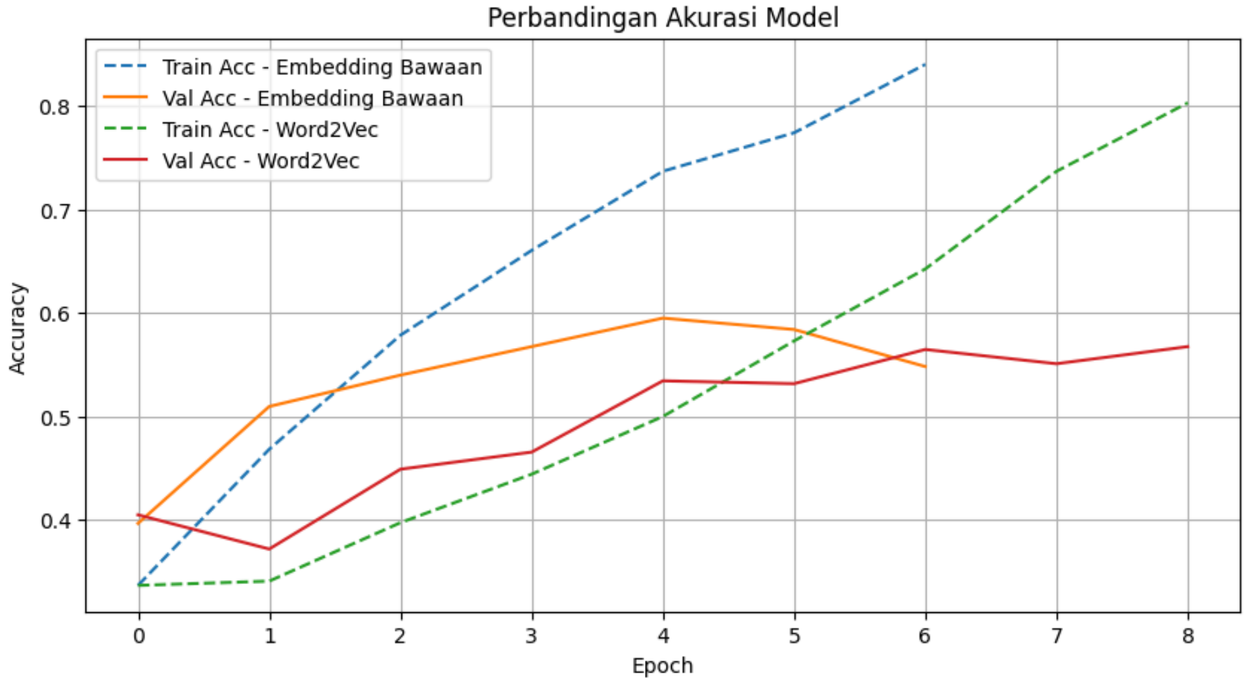
Training Parameters

Loss Function: Sparse categorical crossentropy

Optimizer: Adam

Batch Size: 16

Epochs: 15

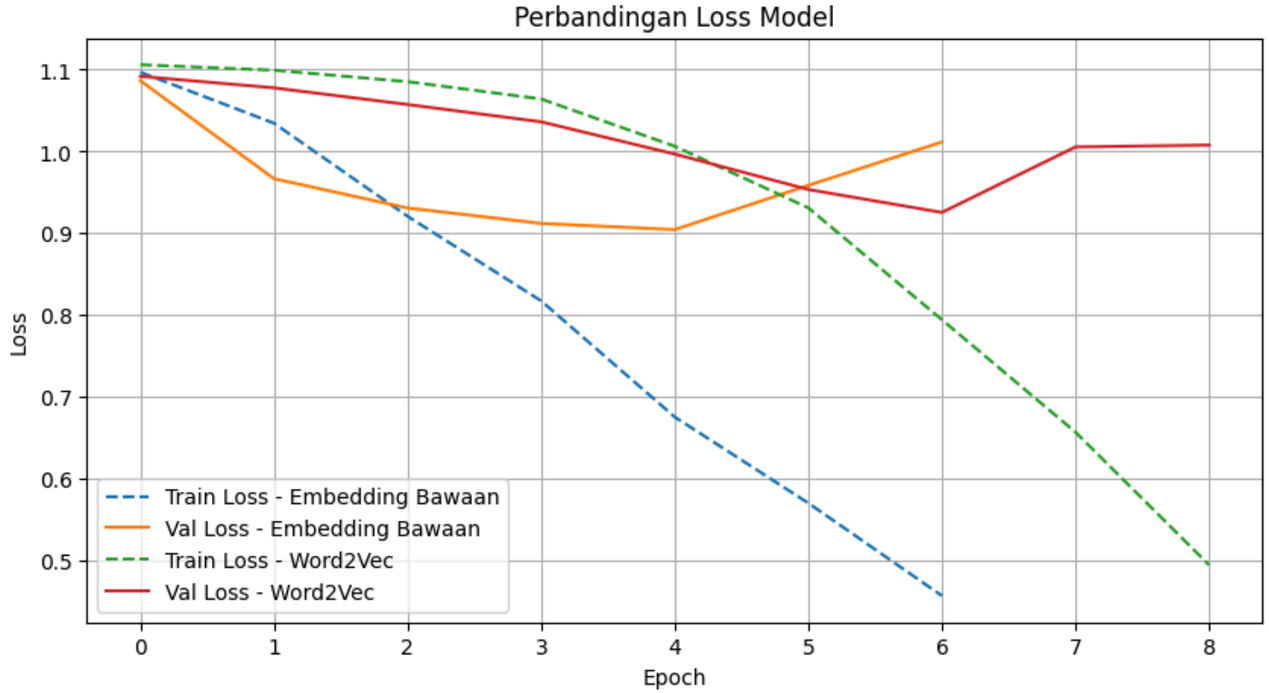


NLP GROUP A - KANG AI

MODELING

DEEP LEARNING ALGORITHM : LSTM

3) MODEL EVALUATION



NLP GROUP A - KANG AI

TESTING RESULT

```
Text from user: Masih banyak kinerja prabowo sebagai presiden yang harus dikritisi
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
Predicted Sentiment: Sentimen Negatif, Hasil Prediksi: 0.13847339153289795
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step
Predicted Sentiment: Sentimen Negatif, Hasil Prediksi: 0.08926136791706085
Text from user: Pertumbuhan ekonomi menjadi salah satu titik utama penilaian kinerja kabinet baru prabowo
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
Predicted Sentiment: Sentimen Negatif, Hasil Prediksi: 0.09263844043016434
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
Predicted Sentiment: Sentimen Negatif, Hasil Prediksi: 0.08870456367731094
Text from user: Prabowo tuh keren banget, benar-benar bekerja sebagai presiden #BismillahRumahdiMenteng
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
Predicted Sentiment: Sentimen Negatif, Hasil Prediksi: 0.43508556485176086
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 22ms/step
Predicted Sentiment: Sentimen Negatif, Hasil Prediksi: 0.19846287369728088
```

NLP GROUP A - KANG AI

CONCLUSION

Dari dua pendekatan algoritma yang digunakan untuk melatih performa model NLP, bisa disimpulkan bahwa:

- Algoritma ML, terutama dengan menggunakan pendekatan TF-IDF memiliki keunggulan dalam efisiensi. Namun, bias masih bisa terjadi dalam sebuah konteks kalimat (ada keterbatasan).
- Algoritma NN, yang memanfaatkan iterasi hyperparameter tuning, embedding bawaan dan kemudian melakukan proses embedding kembali menggunakan Word2Vec yang memiliki kelebihan bisa lebih menangkap makna semantik dalam kalimat-kalimat pada sebuah dataset.
- Algoritma ML lebih stabil dilakukan, terutama terhadap dataset dengan ukuran yang kecil-sedang (dalam projek ini, jumlah cuitan berjumlah -+ 1.800 cuitan.
- Algoritma NN, walaupun tidak secepat model ML (terlihat dari grafik), namun dapat menghasilkan yang lebih general, tidak menghasilkan overfitting seperti model ML.
- Kombinasi dua algoritma ini ideal untuk dilakukan, terutama menggunakan TF-IDF di stage filtering pertama, kemudian lanjut ke algoritma NN untuk melakukan pemahaman yang lebih mendalam di proses selanjutnya.

NLP GROUP A -
KANG AI

THANK YOU