LAMPIRAN

Proses yang dijelaskan dalam skripsi dibagi menjadi tahapan utama berdasarkan deskripsi pada BAB 3 (Metode Penelitian) dan BAB 4 (Implementasi dan Pembahasan).

1. Pengumpulan Data: Data diperoleh dari platform X (Twitter) menggunakan tools tweet-harvest, dengan kata kunci terkait "Danantara", rentang waktu 24 Februari hingga 14 Maret 2025, menghasilkan 1.601 baris data awal dalam format CSV.

```
filename = 'Danantara.csv'
search_keyword = 'Danantara BUMN since:2025-02-24
until:2025-03-14 lang:id'
limit = 300
!npx -y tweet-harvest@2.6.1 -o "{filename}" -s
"{search_keyword}" --tab "LATEST" -l {limit} --token
{twitter_auth_token}
```

.

2. Pembersihan Data dan Seleksi Fitur: Data dibersihkan dengan menghapus nilai kosong dan duplikat, menghasilkan 1.219 baris. Fitur utama yang dipilih adalah kolom "full_text" sebagai representasi opini pengguna.

```
# Hapus duplikat dan Nilai Kosong
df.dropna(subset=['full_text'], inplace=True)
print(f"[2] Jumlah data setelah menghapus nilai kosong:
{len(df)}")
# Hapus data duplikat
df.drop_duplicates(subset=['full_text'], inplace=True)
print(f"[3] Jumlah data setelah menghapus duplikat:
{len(df)}")
#Seleksi Atribut (full_text)
if 'full_text' not in df.columns:
    raise ValueError("Kolom 'full_text' tidak ditemukan
dalam dataset!")
df = df[['full_text']]
```

3. Pra-Pemrosesan Teks: Teks dinormalisasi melalui tahapan: case folding (ubah ke huruf kecil), cleansing (hapus URL, mention, hashtag, angka, tanda baca), tokenisasi, stopword removal menggunakan NLTK, stemming dengan Sastrawi, joining, dan filtering baris kosong, menghasilkan 1.215 baris data bersih.

```
# --- 2. Pra-pemrosesan Teks ---
factory = StemmerFactory()
stemmer = factory.create_stemmer()
stop_words = set(stopwords.words('indonesian'))
def clean_text(text):
    text = str(text).lower()
    text = re.sub(r'http\S+|@\w+|#\w+', '', text)
    text = re.sub(r'\d+', '', text)
    text = text.translate(str.maketrans('', '',
string.punctuation))
    tokens = text.split()
    tokens = [word for word in tokens if word not in
stop_words]
    stemmed = [stemmer.stem(word) for word in tokens]
    return ' '.join(stemmed)

df['cleaned_text'] = df['full_text'].apply(clean_text)
```

4. Pelabelan Sentimen Berbasis Lexicon: Pelabelan dilakukan menggunakan kamus InSet Lexicon untuk menghitung skor polaritas: >0 positif, <0 negatif, =0 netral. Data netral dipisahkan, meninggalkan data positif dan negatif untuk analisis selanjutnya.

```
# --- 3. Load Lexicon dengan Skor Polaritas ---
positive_words = {}
with open('positive.tsv', 'r') as f:
    next(f) # Skip header row if present
    for line in f:
        try:
            word, score = line.strip().split('\t')
            positive_words[word] = int(score)
        except ValueError:
            # Skip lines where score is not a valid integer
            continue
negative_words = {}
with open('negative.tsv', 'r') as f:
    next(f) # Skip header row if present
    for line in f:
        try:
            word, score = line.strip().split('\t')
            negative_words[word] = int(score)
        except ValueError:
            # Skip lines where score is not a valid integer
            Continue
# --- Labeling ---
df_labeled = df
df_labeled['sentiment'] = ''
for index, row in df_labeled.iterrows():
    words = row['cleaned_text'].split()
    total_score = 0
```

```
    for word in words:
        if word in positive_words:
            total_score += positive_words[word]
        elif word in negative_words:
            total_score += negative_words[word]
    if total_score > 0:
        df_labeled.loc[index, 'sentiment'] = 'positive'
    elif total_score < 0:
        df_labeled.loc[index, 'sentiment'] = 'negative'
    else:
        df_labeled.loc[index, 'sentiment'] = 'neutral'
df = df_labeled
```

5. Pembagian Data: Label netral dipisahkan kemudian data final dibagi menjadi 80% data latih (901 baris) dan 20% data uji (226 baris) menggunakan rasio 80:20 dengan stratify untuk menjaga keseimbangan kelas.

```
#Seleksi netral
f_netral = df[df['sentiment'] == 'neutral']
df_final = df[df['sentiment'] != 'neutral']
df_netral.to_csv('data_netral.csv', index=False)
df_final.to_csv('data_final.csv', index=False)

X = df_final['cleaned_text']
y = df_final['sentiment']

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

6. Ekstraksi Fitur dengan TF-IDF: Data teks diubah menjadi representasi numerik menggunakan TF-IDF, menghasilkan 2.641 fitur unik dari data latih.

```
vectorizer = TfidfVectorizer()
X_train_tfidf = vectorizer.fit_transform(X_train)
X_test_tfidf = vectorizer.transform(X_test)
```

7. Penyeimbangan Data dengan SMOTE: Teknik oversampling SMOTE diterapkan hanya pada data latih untuk menyeimbangkan kelas (649 sampel per kelas setelah penyeimbangan), mengatasi ketidakseimbangan antara positif dan negatif.

```
from imblearn.over_sampling import SMOTE

smote = SMOTE(random_state=42)
X_train_resampled, y_train_resampled =
smote.fit_resample(X_train_tfidf, y_train)
```

8. Pelatihan Model SVM: Model SVM dengan kernel linear dilatih menggunakan data latih (sebelum dan sesudah SMOTE) untuk membandingkan performa.

```
# Pelatihan Sesudah SMOTE pakai data hasil SMOTE
(resampled)
svm_model = SVC(kernel='linear', random_state=42)
svm_model.fit(X_train_resampled, y_train_resampled)

# Pelatihan tanpa SMOTE
svm_model_before_smote = SVC(kernel='linear',
random_state=42)
svm_model_before_smote.fit(X_train_tfidf, y_train)
```

9. Pengujian dan Evaluasi: Model diuji dengan data uji, menghasilkan akurasi 81.86% setelah SMOTE. Evaluasi menggunakan metrik: akurasi, precision, recall, F1-score, dan confusion matrix. Hasil menunjukkan peningkatan pada kelas negatif setelah SMOTE.

```
# Pengujian dan Evaluasi Sesudah SMOTE pakai resampled data
y_pred_after_smote = svm_model.predict(X_test_tfidf)
y_pred_before_smote =
svm_model_before_smote.predict(X_test_tfidf) # Pengujian
dan Evaluasi tanpa SMOTE
```

PEMBERITAHUAN SEBELUM UJIAN :
Jika pengumpulan dokumen Tugas Akhir di perpustakaan melewati batas akhir semester berjalan, maka mahasiswa harus menyelesaikan registrasi dan KRS semester berikutnya.

KRITERIA KELULUSAN UJIAN TUGAS AKHIR

1. Lulus dengan memperhatikan catatan ujian tugas akhir, dan atau melakukan perbaikan atau penyempurnaan naskah dan atau produk dalam waktu maksimum dua bulan dari tanggal ujian tugas akhir, yaitu

mulai Kamis, 24 Juli 2025 sampai dengan Rabu, 24 September 2025

Jika dalam waktu yang ditentukan mahasiswa tersebut tidak dapat menyelesaikan, maka mahasiswa yang bersangkutan dianggap tidak lulus ujian.

2. Tidak lulus, disarankan oleh Ketua Tim Penguji untuk mempelajari ulang materi, merombak produk/naskah, atau mengganti judul.

Ketentuan bagi peserta yang tidak lulus ujian tugas akhir.

1) Mahasiswa wajib menempuh ujian tugas akhir ulang

2) Kesempatan ujian tugas akhir ulang hanya diberikan dalam rentang waktu maksimum 6 bulan, setelah ujian sidang/pendadaran

3) Jika sampai batas waktu maksimum 6 bulan tersebut belum dapat diajukan/diselesaikan, maka calon peserta ujian dinyatakan sebagai mahasiswa peserta Tugas Akhir baru, dengan segala ketentuan yang berlaku bagi peserta baru

4) Mahasiswa yang akan menempuh ujian tugas akhir ulang ini diwajibkan membayar biaya ujian sesuai tarif yang ditetapkan.

Yogyakarta, _____

Memahami dan bersedia

Mematuhi peraturan di atas,

Petra Aldevand Hosyo

YAYASAN PENDIDIKAN WIDYA BAKTI YOGYAKARTA
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
Jl. Raya Janti (Majapahit) No.143, Yogyakarta, 55198, Telp (0274) 486664,
Website: www.utdi.ac.id , E-mail: info@utdi.ac.id

Hari, tanggal	:	Kamis, 24 Juli 2025
Waktu	:	10.00
Nama	:	Petra Aldevand Hosyo
No. Mahasiswa / Prodi	:	215610042 / Sistem Informasi

No	Hal yang harus diperbaiki	Pemberi Catatan
1.	Pada Bab 3 tidak ada tahapan dari rancangan sistem, dan analysis svm. Dari mulai Tabel 4.1 sampai dengan Tabel 4.14 tidak sesuai, yang disajikan code tapi diberi nama Tabel.Tidak terlihat rancangan sistem dan Implementasi sistem yang dibangun. Sajikan proses dilengkapi dengan data dari setiap tahapan penelitian. Perbaiki spasi dari beberapa penulisan di setiap Bab.	Asyahri Hadi
2.	* Tata tulis naskah mohon diperhatikan lagi * bab 3 belum memumculkan rancangan terkait dengan sistem yang akan dibuat * halaman 52?? perlu dipertimbangkan	debbie
3.		
4.		

YAYASAN PENDIDIKAN WIDYA BAKTI YOGYAKARTA
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
Jl. Raya Janti (Majapahit) No.143, Yogyakarta, 55198, Telp (0274) 486664,
Website: www.utdi.ac.id , E-mail: info@utdi.ac.id

KEPUTUSAN HASIL UJIAN PENDADARAN

Sesuai dengan hasil sidang pendadaran pada tanggal 24 Juli 2025 maka

Nama Mahasiswa	Petra Aldevand Hosyo
NIM / Program Studi	215610042 / Sistem Informasi
Jenjang	S1

dinyatakan LULUS

Ketua Penguji Deborah Kurniawati, S.Kom., M.Cs.