

Lampiran 16 source code

➤ Source code instalisasi library dan Hasil Output

```
[16]
✓ 26s
!pip install Sastrawi
!pip install tqdm
!pip install scikit-learn

... Requirement already satisfied: Sastrawi in /usr/local/lib/python3.12/dist-packages (1.0.1)
Requirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-packages (4.67.3)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.12/dist-packages (1.6.1)
Requirement already satisfied: numpy>=1.19.5 in /usr/local/lib/python3.12/dist-packages (from scikit-learn) (2.0.2)
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.12/dist-packages (from scikit-learn) (1.16.3)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.12/dist-packages (from scikit-learn) (1.5.3)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.12/dist-packages (from scikit-learn) (3.6.0)
```

➤ Source code import library pada google colab

```
[2]
✓ 1s
import pandas as pd
import re
import regex
from google.colab import files
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from tqdm import tqdm
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score
import matplotlib.pyplot as plt
```

➤ Source code upload dataset excel pada google colab

```
print("Silahkan upload file Excel (.xlsx) komentar...")
uploaded = files.upload()

file_name = list(uploaded.keys())[0]
print("File berhasil diupload:", file_name)

... Silahkan upload file Excel (.xlsx) komentar...
Pilih File | komentar-dengan-label.xlsx
komentar-dengan-label.xlsx(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 99482 bytes, last modified: 22/2/2026 - 100% done
Saving komentar-dengan-label.xlsx to komentar-dengan-label.xlsx
File berhasil diupload: komentar-dengan-label.xlsx
```

➤ Source code membaca dan membersihkan nama kolom

```
[4]
✓ 0s
df = pd.read_excel(file_name)

df.columns = df.columns.str.strip()
print("Daftar Kolom:", df.columns)

Daftar Kolom: Index(['Unique ID', nan, 'Unnamed: 1',
                    'Date', 'Name',
                    'Comment', 'Likes',
                    '(view source comment)', 'Profile ID',
                    'Label', 'Trust'],
dtype='object')
```

➤ Source code deteksi kolom komentar pada dataset

```
[5]
✓ Os
comment_candidates = ["Comment", "comment", "Komentar", "komentar"]
COMMENT_COLUMN = None

for col in df.columns:
    if col in comment_candidates:
        COMMENT_COLUMN = col
        break

if COMMENT_COLUMN is None:
    raise ValueError("Kolom komentar tidak ditemukan")

df[COMMENT_COLUMN] = df[COMMENT_COLUMN].astype(str)
print("Kolom komentar:", COMMENT_COLUMN)
```

... Kolom komentar: Comment

➤ Source code pembuatan lexicon

```
[6]
✓ Os
LEXICON = {

    # Positif
    "percaya": 3, "terpercaya": 3, "valid": 2, "jujur": 2, "resmi": 2,
    "baik": 1, "bagus": 1, "senang": 2, "mantap": 2,

    # Negatif
    "ragu": -3, "meragukan": -3, "aneh": -2, "drama": -2,
    "lebay": -2, "settingan": -3,
    "bohong": -4, "hoax": -4, "scam": -4,
    "tipu": -4, "penipuan": -4, "fake": -4,

}
```

➤ Source code preprocessing teks

```
[7]
✓ Os
factory = StemmerFactory()
stemmer = factory.create_stemmer()

def clean_text(text):
    text = text.lower()
    text = regex.sub(r'^\w\s', ' ', text)
    text = re.sub(r'\s+', ' ', text).strip()
    return text

def tokenize(text):
    return clean_text(text).split()
```

- Source code proses perhitungan sentiment score

```
[8]
✓ 0s
def sentiment_score(text):
    words = tokenize(text)
    total, count = 0, 0

    for word in words:
        base = stemmer.stem(word)
        if base in LEXICON:
            total += LEXICON[base]
            count += 1

    if count == 0:
        return 0
    return total / count
```

- Source code analisis seluruh komentar

```
[10]
✓ 1m
scores = []
predictions = []

for comment in tqdm(df[COMMENT_COLUMN], desc="Analisis komentar"):
    s = sentiment_score(comment)
    scores.append(s)
    predictions.append(classify_label(s))

df["Sentiment Score"] = scores
df["Label Prediction"] = predictions

Analisis komentar: 100%|██████████| 999/999 [01:47<00:00, 9.33it/s]
```

- Source code proses evaluasi klasifikasi sentimen dan hasil output evaluasi

```
[11]
✓ 0s
df["Label"] = df["Label"].astype(str).str.strip().str.capitalize()

df = df.dropna(subset=["Label"])

true_labels = df["Label"]
pred_labels = df["Label Prediction"]

acc = accuracy_score(true_labels, pred_labels)
print("\nAkurasi Model:", round(acc*100,2), "%")

print("\nClassification Report:")
print(classification_report(true_labels, pred_labels))

cm = confusion_matrix(
    true_labels,
    pred_labels,
    labels=["Negatif", "Netral", "Positif"]
)

print("\nConfusion Matrix:")
print(cm)
```

...

Akurasi Model: 89.8 %

Classification Report:

	precision	recall	f1-score	support
Negatif	0.75	0.04	0.07	78
Netral	0.91	0.99	0.95	871
Positif	0.31	0.23	0.26	22
accuracy			0.90	971
macro avg	0.66	0.42	0.43	971
weighted avg	0.88	0.90	0.86	971

Confusion Matrix:

```
[[ 3  70  5]
 [ 1 864  6]
 [ 0  17  5]]
```

➤ Source code visualisasi confusion matrix sentimen

```
[13]
✓ Os  label_counts = df["Label Prediction"].value_counts()

plt.figure(figsize=(6,4))
plt.bar(label_counts.index, label_counts.values)
plt.title("Distribusi Prediksi Sentimen")
plt.xlabel("Kategori")
plt.ylabel("Jumlah Komentar")
plt.grid(axis='y', linestyle='--', alpha=0.4)
plt.show()
```

➤ Source code distribusi nilai sentiment score

```
[14]
✓ Os  plt.figure(figsize=(6,4))
plt.hist(df["Sentiment Score"], bins=20)
plt.axvline(-1, linestyle="--")
plt.axvline(1, linestyle="--")
plt.title("Distribusi Sentiment Score")
plt.xlabel("Nilai Score")
plt.ylabel("Jumlah Komentar")
plt.show()
```

```
[12]
✓ Os ▶ labels = ["Negatif", "Netral", "Positif"]

plt.figure(figsize=(5,5))
plt.imshow(cm)
plt.title("Confusion Matrix Sentimen")

plt.xticks(range(len(labels)), labels)
plt.yticks(range(len(labels)), labels)

for i in range(len(labels)):
    for j in range(len(labels)):
        plt.text(j,i,cm[i,j],ha="center",va="center")

plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```

➤ Source code heatmap confusion matrix trust issue

```
[15]
✓ Os ▶ plt.figure(figsize=(5,5))
plt.imshow(cm)
plt.title("Confusion Matrix Sentimen")
plt.colorbar()

plt.xticks(range(len(labels)), labels)
plt.yticks(range(len(labels)), labels)

for i in range(len(labels)):
    for j in range(len(labels)):
        plt.text(j,i,cm[i,j],ha="center",va="center")

plt.xlabel("Prediksi Model")
plt.ylabel("Label Manual")
plt.show()
```

➤ Source code proses penyimpanan hasil analisis ke file excel

```
[1]
df.to_excel("Hasil_Trust_Evaluasi.xlsx", index=False)
print("File hasil tersimpan!")
```

▼ File hasil tersimpan!

DAFTAR RIWAYA HIDUP



Penulis bernama Fikariny Bethony, lahir di kondongan' Toraja Utara pada tanggal 24 November 2001. Penulis merupakan anak kedua dari dua bersaudara, dari pasangan Bapak Nicholas Bethony dan Ibu Anastasia Padang'. Berstatus sebagai warga negara Indonesia dan menganut agama Kristen Protestan, saat ini penulis berdomosili di kelurahan Mentirotiku, Kecamatan Rantepao

Kabupaten Toraja Utara, Provinsi Sulawesi Selatan. Penulis telah menempuh beberapa pendidikan formal dari SDN 6 Rantepao tahun 2007-2013, kemudian melanjutkan pendidikan di SMPN 1 Tikala tahun 2013-2016. Setelah itu melanjutkan pendidikan di SMA PELITA RANTEPAO tahun 2016-2019. Pada tahun 2021 penulis melanjutkan pendidikan ke perguruan tinggi di Universitas Kristen Indonesia Toraja dan lulus pada tahun 2026.