LAMPIRAN

```
#import library
from google.colab import files

uploaded = files.upload()
```

```
Choose Files  komentar (3).xlsx
komentar (3).xlsx(application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 17853 bytes, last modified: 2/6/2026 - 100% done
Saving komentar (3).xlsx to komentar (3).xlsx
```

```
!pip install pandas openpyxl matplotlib
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.12/dist-packages (2.2.2)
Requirement already satisfied: openpyxl in /usr/local/lib/python3.12/dist-packages (3.1.5)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.12/dist-packages (3.10.0)
Requirement already satisfied: numpy>=1.26.0 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.3)
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.12/dist-packages (from openpyxl) (2.0.0)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (1.3.3)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (4.62.0)
Requirement already satisfied: kiwisolver>=1.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (1.5.0)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (26.0)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (11.3.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.12/dist-packages (from matplotlib) (3.3.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
```

```
#load dataset
import pandas as pd

file_path = "komentar.xlsx"
df = pd.read_excel(file_path)

# Cek data
df.head()
```

```python
#load dataset
import pandas as pd

file_path = "komentar.xlsx"
df = pd.read_excel(file_path)

# Cek data
df.head()
```

	No	Komentar	Tanggal	Label
0	1	Shalom kk admin, ijin bertanya, untuk CAT haru...	2025-01-12	positif
1	2	mau isi trucer study tapi ga bisa pake NIM ,gi...	2025-01-12	netral
2	3	Ijin tanya min. Bagi yang mendaftar jalur pres...	2025-01-13	positif
3	4	Oke ko cekong	2025-01-13	netral
4	5	Shalom kak admin, ijin bertanya. Kapan pendaft...	2025-01-13	netral

Next steps: Generate code with df | New interactive sheet

```python
#Case Folding
def case_folding(text):
    text = text.lower()
    return text

df["Komentar"] = df["Komentar"].astype(str)
df["Komentar_CF"] = df["Komentar"].apply(case_folding)
```

```python
import re

#Cleaning
def cleaning(text):
    text = re.sub(r"http\S+", "", text)
    text = re.sub(r"@\w+", "", text)
    text = re.sub(r"#\w+", "", text)
```

```
#Cleaning
def cleaning(text):
    text = re.sub(r"http\S+", "", text)
    text = re.sub(r"@\w+", "", text)
    text = re.sub(r"#\w+", "", text)
    text = re.sub(r"[0-9]+", "", text)
    text = text.strip()
    return text

df["Komentar_Clean"] = df["Komentar_CF"].apply(cleaning)
```

```
#Stopword Removal
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords

stop_words = set(stopwords.words("indonesian"))

def remove_stopwords(text):
    words = text.split()
    words = [word for word in words if word not in stop_words]
    return " ".join(words)

df["Komentar_Bersih"] = df["Komentar_Clean"].apply(remove_stopwords)
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

```
#kamus lexicon sentimen
positive_words = [
    "baik","bagus","hebat","mantap","keren","sukses","luar biasa","terbaik"
]

negative_words = [
    "buruk","jelek","parah","kecewa","benci","kurang","gagal"
]
```

```
#lexicon_based scoring
def lexicon_sentiment(text):

    score = 0

    words = text.split()

    for word in words:

        if word in positive_words:
            score += 1

        elif word in negative_words:
            score -= 1

    if score > 0:
        return "Positif"

    elif score < 0:
        return "Negatif"

    else:
        return "Netral"
```

```
#prediksi sentimen lexicon
df["Sentimen_Lexicon"] = df["Komentar_Bersih"].apply(lexicon_sentiment)

df.head()
```

```
#spilt dataset
from sklearn.model_selection import train_test_split

X = df["Komentar_Bersih"]
y = df["Label"]

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

```
from sklearn.feature_extraction.text import CountVectorizer

vectorizer = CountVectorizer()

X_train_vec = vectorizer.fit_transform(X_train)
X_test_vec = vectorizer.transform(X_test)
```

```
from sklearn.naive_bayes import MultinomialNB

#training model naive bayes
model_nb = MultinomialNB()

model_nb.fit(X_train_vec, y_train)
```

```
▾ MultinomialNB
MultinomialNB()
```

```
#prediksi data testing
y_pred = model_nb.predict(X_test_vec)
```

```python
#visualisasi cofusion matrix
from sklearn.metrics import confusion_matrix, accuracy_score
import matplotlib.pyplot as plt
import seaborn as sns

y_true = df["Label"]
y_pred = df["Sentimen_Prediksi"]

cm = confusion_matrix(
    y_true,
    y_pred,
    labels=["positif", "netral", "negatif"]
)

akurasi = accuracy_score(y_true, y_pred)

print("Confusion Matrix (Naive Bayes):")
print(cm)
print("Akurasi Naive Bayes (Full Dataset):", akurasi)

plt.figure(figsize=(6,5))
sns.heatmap(
    cm,
    annot=True,
    fmt="d",
    cmap="Blues",
    xticklabels=["positif","netral","negatif"],
    yticklabels=["positif","netral","negatif"]
)

plt.xlabel("Prediksi Naive Bayes")
plt.ylabel("Label Manual")
plt.title("Confusion Matrix Naive Bayes Analisis Sentimen komentar UKI Toraja")
plt.show()
```

```python
hasil_file = "hasil_analisis_sentimen.xlsx"
df.to_excel(hasil_file, index=False)

print(f"File hasil analisis berhasil disimpan: {hasil_file}")
```

DAFTAR RIWAYAT HIDUP

Penulis Bernama **Elni Ma'tang Bonggakarua,** yang akrab disapa Elni, lahir di pondingao' pada tanggal 23 mei 2003, merupakan anak ketiga dari empat bersaudara dari pasangan Daniel Bonggakarua dan Jeni Kombong. Berstatus sebagai warga negara indonesia dan menganut agama kristen protestan, saat ini penulis berdomisili di Kecamatan Bittuang, Kabupaten Tana Toraja, Provinsi Sulawesi Selatan. Saya memulai pendidikan formal di SDN 352 Sasak, dan lulus pada tahun 2015, selanjutnya melanjutkan pendidikan di SMPN 1 Bittuang, dan lulus pada tahun 2018, selanjutnya melanjutkan pendidikan di SMAN 11 Tana Toraja, dan lulus pada tahun 2021. Pada tahun 2021, saya terdaftar sebagai mahasiswa di Universitas Kristen Indonesia Toraja, Fakultas Teknik Informatika. Dan lulus pada tahun 2026.