# Sentiment Analysis Kaoutar lakdim

```
!pip install -q vaderSentiment
```

```
import pandas as pd
import re, string, nltk
from nltk.tokenize import TweetTokenizer
from nltk.corpus import stopwords
# --- VADER: choose ONE of the next two imports --------------------
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer      # (offline-friendly)
# from nltk.sentiment import SentimentIntensityAnalyzer                  # <- use this if you prefer NLTK's copy
# ------------------------------------------------------------------

nltk.download("stopwords", quiet=True)   # only ~30 KB; no "punkt" needed

# ------------------------------------------------------------------
# 2) Load your tweets CSV (change the filename/column names as needed)
# ------------------------------------------------------------------
df = pd.read_csv("tweets-data.csv")      # or the file you exported earlier
df = df.sample(n=500, random_state=42)   # matches “take a sample of 500 rows”

# If your tweets column is named something else (e.g. "text"), edit here:
TWEET_COL = "Tweets"

# ------------------------------------------------------------------
# 3) Cleaning helper
# ------------------------------------------------------------------
URL_RX  = re.compile(r'https?://\S+|www\.\S+')
MENT_RX = re.compile(r'@\w+')
HASH_RX = re.compile(r'#(\w+)')          # keeps the word, drops the #

tokenizer = TweetTokenizer()
STOP = set(stopwords.words("english")).union({"rt"})  # add extra stop-words if you like

def clean_tweet(text: str) -> str:
    text = URL_RX.sub("", text)
    text = MENT_RX.sub("", text)
    text = HASH_RX.sub(r"\1", text)
    text = text.translate(str.maketrans("", "", string.punctuation))
    tokens = [tok.lower() for tok in tokenizer.tokenize(text)
              if tok.isalpha() and tok.lower() not in STOP]
    return " ".join(tokens)              # <- single “sentence” required by the brief

# ------------------------------------------------------------------
# 4) VADER wrapper → returns (label, compound score)
# ------------------------------------------------------------------
analyzer = SentimentIntensityAnalyzer()

def vader_sentiment(sentence: str, neutral_threshold: float = 0.05):
    cscore = analyzer.polarity_scores(sentence)["compound"]
    if cscore >=  neutral_threshold:
        label = "Positive"
    elif cscore <= -neutral_threshold:
        label = "Negative"
    else:
        label = "Neutral"
    return pd.Series([label, cscore])

# ------------------------------------------------------------------
# 5) Apply the pipeline and create the two new columns
# ------------------------------------------------------------------
df["cleaned"] = df[TWEET_COL].astype(str).apply(clean_tweet)                     # step ②
df[["sentiment_label", "sentiment_score"]] = df["cleaned"].apply(vader_sentiment)  # step ③

# ------------------------------------------------------------------
# 6) Quick sanity-check
# ------------------------------------------------------------------
print(df[[TWEET_COL, "cleaned", "sentiment_label", "sentiment_score"]].head())
```

```
                                                Tweets  \
2899  Le #DessinDePresse de Sanaga : ls sont morts c...
594   #Russia #Wagner #RussiaCivilWar https://t.co/P...
2870  Exclusive content -https://t.co/oEiSIIB2Z1\n.\...
52    Auch heute geht die politische Nachricht des T...
1391  @crazyclipsonly Same type that would take a ho...
```

```
                                                cleaned sentiment_label  \
2899  le dessindepresse de sanaga ls sont morts comm...        Positive
594                        russia wagner russiacivilwar         Neutral
2870  exclusive content cosplay japan titan titanics...        Negative
52    auch heute geht die politische nachricht des t...        Negative
1391  type would take homemade playstationcontrolled...         Neutral

      sentiment_score
2899           0.4767
594            0.0000
2870          -0.4404
52            -0.5994
1391           0.0000
```

```
## transfomers
```

```
!pip install transformers torch pandas nltk
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.11/dist-packages (4.52.4)
Requirement already satisfied: torch in /usr/local/lib/python3.11/dist-packages (2.6.0+cu124)
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: nltk in /usr/local/lib/python3.11/dist-packages (3.9.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from transformers) (3.18.0)
Requirement already satisfied: huggingface-hub<1.0,>=0.30.0 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.33.0)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.11/dist-packages (from transformers) (2.0.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.11/dist-packages (from transformers) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.11/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.11/dist-packages (from transformers) (2024.11.6)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from transformers) (2.32.3)
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.21.2)
Requirement already satisfied: safetensors>=0.4.3 in /usr/local/lib/python3.11/dist-packages (from transformers) (0.5.3)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.11/dist-packages (from transformers) (4.67.1)
Requirement already satisfied: typing-extensions>=4.10.0 in /usr/local/lib/python3.11/dist-packages (from torch) (4.14.0)
Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-packages (from torch) (3.5)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.11/dist-packages (from torch) (3.1.6)
Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from torch) (2025.3.2)
Requirement already satisfied: nvidia-cuda-nvrtc-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch) (12.4.127)
Requirement already satisfied: nvidia-cuda-runtime-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch) (12.4.127
Requirement already satisfied: nvidia-cuda-cupti-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch) (12.4.127)
Requirement already satisfied: nvidia-cudnn-cu12==9.1.0.70 in /usr/local/lib/python3.11/dist-packages (from torch) (9.1.0.70)
Requirement already satisfied: nvidia-cublas-cu12==12.4.5.8 in /usr/local/lib/python3.11/dist-packages (from torch) (12.4.5.8)
Requirement already satisfied: nvidia-cufft-cu12==11.2.1.3 in /usr/local/lib/python3.11/dist-packages (from torch) (11.2.1.3)
Requirement already satisfied: nvidia-curand-cu12==10.3.5.147 in /usr/local/lib/python3.11/dist-packages (from torch) (10.3.5.147)
Requirement already satisfied: nvidia-cusolver-cu12==11.6.1.9 in /usr/local/lib/python3.11/dist-packages (from torch) (11.6.1.9)
Requirement already satisfied: nvidia-cusparse-cu12==12.3.1.170 in /usr/local/lib/python3.11/dist-packages (from torch) (12.3.1.170
Requirement already satisfied: nvidia-cusparselt-cu12==0.6.2 in /usr/local/lib/python3.11/dist-packages (from torch) (0.6.2)
Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/python3.11/dist-packages (from torch) (2.21.5)
Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch) (12.4.127)
Requirement already satisfied: nvidia-nvjitlink-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch) (12.4.127)
Requirement already satisfied: triton==3.2.0 in /usr/local/lib/python3.11/dist-packages (from torch) (3.2.0)
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/dist-packages (from torch) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from sympy==1.13.1->torch) (1.3.0)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas) (2025.2)
Requirement already satisfied: click in /usr/local/lib/python3.11/dist-packages (from nltk) (8.2.1)
Requirement already satisfied: joblib in /usr/local/lib/python3.11/dist-packages (from nltk) (1.5.1)
Requirement already satisfied: hf-xet<2.0.0,>=1.1.2 in /usr/local/lib/python3.11/dist-packages (from huggingface-hub<1.0,>=0.30.0->t
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinja2->torch) (3.0.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (3
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (2.4.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests->transformers) (2025.6.1
```

```
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from transformers import pipeline
nltk.download('stopwords')
stop_words = set(stopwords.words('english'))
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

```
def clean_tweet(text):
    text = re.sub(r"http\S+|www\S+|https\S+", '', text)
    text = re.sub(r'\@\w+|\#', '', text)
    text = re.sub(r'[^A-Za-z0-9 ]+', '', text)
    tokens = text.lower().split()
    cleaned = [word for word in tokens if word not in stop_words]
    return " ".join(cleaned)
```

```
df['cleaned_tweet'] = df['Tweets'].astype(str).apply(clean_tweet)
```

```
sentiment_pipeline = pipeline("sentiment-analysis", model="distilbert-base-uncased-finetuned-sst-2-english")
```

```
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning: 
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
```

```
model.safetensors: 100%                         268M/268M [00:03<00:00, 82.1MB/s]

tokenizer_config.json: 100%                       48.0/48.0 [00:00<00:00, 954B/s]

vocab.txt: 100%                            232k/232k [00:00<00:00, 2.96MB/s]
```

```
Device set to use cpu
```

```
results = sentiment_pipeline(df['cleaned_tweet'].tolist(), truncation=True)
```

```
df['sentiment_label'] = [res['label'].lower() for res in results]
df['sentiment_score'] = [res['score'] for res in results]
```

```
print(df[['Tweets', 'cleaned_tweet', 'sentiment_label', 'sentiment_score']].head())
```

```
                                                 Tweets  \
2899  Le #DessinDePresse de Sanaga : ls sont morts c...   
594   #Russia #Wagner #RussiaCivilWar https://t.co/P...   
2870  Exclusive content -https://t.co/oEiSIIB2Z1\n.\...   
52    Auch heute geht die politische Nachricht des T...   
1391  @crazyclipsonly Same type that would take a ho...   

                                          cleaned_tweet sentiment_label  \
2899  le dessindepresse de sanaga ls sont morts comm...        negative   
594                          russia wagner russiacivilwar        negative   
2870  exclusive content cosplay japan titan titanics...        negative   
52    auch heute geht die politische nachricht des t...        negative   
1391  type would take homemade playstationcontrolled...        negative   

      sentiment_score  
2899         0.981537  
594          0.962062  
2870         0.961531  
52           0.975570  
1391         0.993684  
```

```
sentiment_pipeline = pipeline("sentiment-analysis", model="cardiffnlp/twitter-roberta-base-sentiment")
```

```
Device set to use cpu
```

```
label_map = {
    "NEGATIVE": "negative",
    "NEUTRAL": "neutral",
    "POSITIVE": "positive"
}
df['sentiment_label'] = [label_map[res['label']] for res in results]
df['sentiment_score'] = [res['score'] for res in results]
```

```
print(results[:3])
```

```
[{'label': 'NEGATIVE', 'score': 0.9815365672111511}, {'label': 'NEGATIVE', 'score': 0.9620620608329773}, {'label': 'NEGATIVE', 'scor
```

Commencez à coder ou à générer avec l'IA.