import tweepy

import pandas as pd

from nltk.sentiment import SentimentIntensityAnalyzer

import nltk

nltk.download('vader\_lexicon')

# Set up your Twitter API credentials here

consumer\_key = 'YOUR\_CONSUMER\_KEY'

consumer\_secret = 'YOUR\_CONSUMER\_SECRET'

access\_token = 'YOUR\_ACCESS\_TOKEN'

access\_token\_secret = 'YOUR\_ACCESS\_TOKEN\_SECRET'

# Authenticate to Twitter

auth = tweepy.OAuth1UserHandler(consumer\_key, consumer\_secret, access\_token, access\_token\_secret)

api = tweepy.API(auth)

# Fetch tweets based on a keyword

def fetch\_tweets(keyword, count=100):

    tweets = tweepy.Cursor(api.search, q=keyword, lang="en", tweet\_mode="extended").items(count)

    tweet\_data = []

    for tweet in tweets:

        tweet\_data.append({'text': tweet.full\_text, 'created\_at': tweet.created\_at, 'user': tweet.user.screen\_name})

    return pd.DataFrame(tweet\_data)

# Sentiment analysis using VADER

def analyze\_sentiment(text):

    sia = SentimentIntensityAnalyzer()

    sentiment\_score = sia.polarity\_scores(text)

    if sentiment\_score['compound'] >= 0.05:

        return 'Positive'

    elif sentiment\_score['compound'] <= -0.05:

        return 'Negative'

    else:

        return 'Neutral'

# Apply sentiment analysis to a DataFrame of tweets

def analyze\_tweets\_sentiment(df):

    df['sentiment'] = df['text'].apply(analyze\_sentiment)

    return df

# Main function to fetch and analyze social media sentiment

def main():

    keyword = input("Enter the keyword/topic to analyze: ")

    tweet\_count = int(input("Enter the number of tweets to fetch: "))

    # Fetch tweets

    df\_tweets = fetch\_tweets(keyword, tweet\_count)

    # Analyze sentiment of the tweets

    df\_tweets = analyze\_tweets\_sentiment(df\_tweets)

    # Display results

    print(df\_tweets[['created\_at', 'user', 'text', 'sentiment']])

    # Optional: Save the output to a CSV

    df\_tweets.to\_csv(f'{keyword}\_sentiment\_analysis.csv', index=False)

    print(f"Results saved to {keyword}\_sentiment\_analysis.csv")

if \_\_name\_\_ == "\_\_main\_\_":

    main()