Applied Artificial Intelligence

Name: N. Abhinav	Sap Id: 70572200031
Roll.No : L035	Date: 07-04-2025

Mexico: Web Scraping and Sentiment Analysis with Streamlit App

Submitted To: Prof. Rajesh Prabhakar

SVKM'S NMIMS HYDERABAD

Project Overview

This project presents an end-to-end sentiment analysis system using text extracted from the Wikipedia page of **Mexico**. The goal is to build a sentiment classifier that can identify whether a given sentence is positive or negative, excluding neutral content. The project showcases the full machine learning pipeline — from web scraping, text preprocessing, exploratory analysis, model training, to deployment as a Streamlit web application.

Technical Implementation Analysis

Technologies and Tools Used

- **Python** – Main programming language
- **BeautifulSoup** – Web scraping
- **NLTK** – Text preprocessing (tokenization, stopwords)
- **TextBlob** – Sentiment analysis
- **Scikit-learn** – ML model building
- **Imbalanced-learn** – SMOTE for class balancing
- **WordCloud, Matplotlib, Seaborn** – Visualization
- **Streamlit** – Web application deployment
- **Pickle** – Model serialization

1. Data Collection and Preprocessing

- **Source**: Wikipedia - Mexico
- **Scraping Method**: BeautifulSoup used to extract all paragraph text (<p> tags).
- **Cleaning Steps**:
    - Removed citations [1], [2] etc.
    - Removed digits, punctuation, and special characters
    - Normalized whitespace
- **Sentence Tokenization**: Using nltk.sent_tokenize()
- **Sentiment Analysis**: Using TextBlob().sentiment.polarity
    - **Positive**: polarity > 0.1
    - **Negative**: polarity < -0.1
    - **Neutral**: Ignored

Data Summary:

- **Total Sentences Extracted**: *X*

- **Positive Sentences**: *Y*
- **Negative Sentences**: *Z*

2. Exploratory Data Analysis

Sentiment Distribution

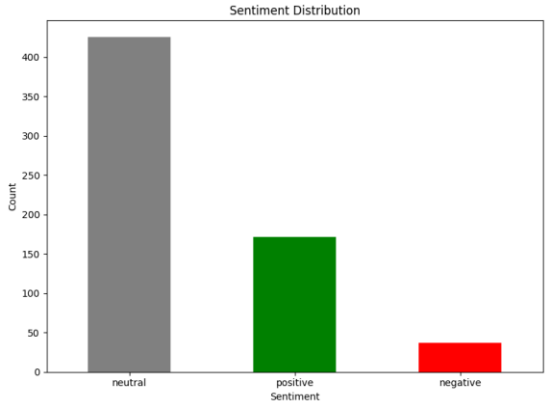
- Pie chart or bar chart representing class balance (Positive vs Negative)

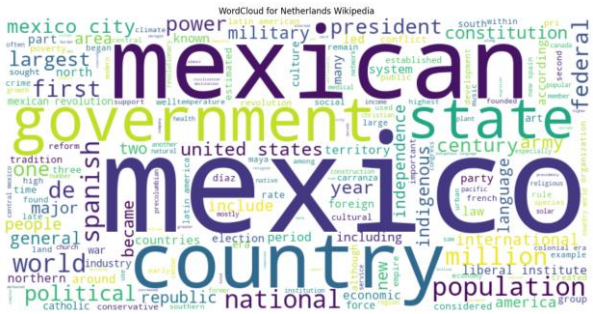
Word Frequency

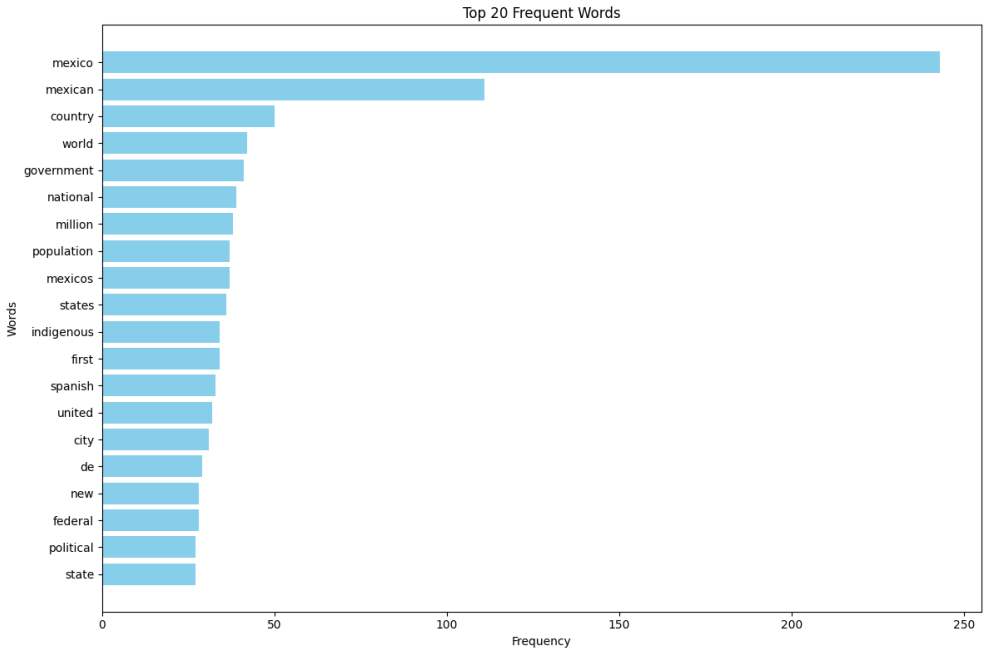
- Top frequent words after removing stopwords using nltk.corpus.stopwords

Word Cloud

- WordCloud generated for better textual insight







3. Feature Engineering

- **TF-IDF Vectorization**: TfidfVectorizer() used to transform sentences into numeric feature vectors.
- **SMOTE**: Synthetic Minority Oversampling Technique applied to balance the dataset between positive and negative classes.

4. Model Training and Evaluation

Six classifiers were trained and evaluated:

Model	**Accuracy**
Logistic Regression	0.91
Decision Tree	0.76
Random Forest	0.89
Gradient Boosting	0.88
Naive Bayes	0.81
K-Nearest Neighbors	0.60

- **Best Model Chosen**: **Logistic Regression**

5. Streamlit Application

The final model was deployed using **Streamlit**.

App Features:

- Input box for user to enter a sentence
- Output prediction: Positive or Negative
- Shows confidence score
- Compares TextBlob prediction vs ML model

Strengths

1. End-to-end pipeline including deployment
2. Clean preprocessing for quality input
3. Balanced dataset using SMOTE
4. Comprehensive model comparison
5. Interactive and user-friendly web interface

Areas for Improvement

- Could include SHAP or LIME for model explainability
- Use cross-validation instead of single train-test split
- Improve accuracy of KNN with hyperparameter tuning
- Include multi-lingual support since Mexico has Spanish content

Conclusion

This project demonstrates a complete NLP and machine learning workflow for analyzing sentiment in the Mexico Wikipedia page content. The system is capable of classifying text into positive or negative sentiment with high accuracy, and is accessible via an easy-to-use Streamlit interface. This methodology can be extended to other domains like tourism reviews, news sentiment, or political analysis in Spanish or English content.

Link of Project:

https://mexico-sentiment-analysis-su83ojzyswkahmwrjw8pjr.streamlit.app/