SENTIMENT ANALYSIS

This project focuses on sentiment analysis for Twitter data, specifically classifying tweets as positive (0) or negative (1) using machine learning models. It involves techniques such as text mining, text analysis, data analysis, and data visualization.

Table of Contents

Introduction

Sentiment analysis is a Natural Language Processing (NLP) technique used to determine the sentiment or emotional tone of a given piece of text. In this project, we focus on sentiment analysis for tweets collected from Twitter. The goal is to classify tweets as positive or negative using various machine learning models.

Data

The dataset used for sentiment analysis of tweets is expected to be in CSV format, with specific columns and formatting requirements. Training dataset in csv file contained ‘id’, ‘label’, ‘tweet’, where the ‘id’ is a unique integer identifying the tweet sentiment is either 0 or 1, and ‘tweet’ is enclosed in “ “ similarly the testing dataset is a csv file of type ‘tweet_id’, ‘tweet’.

	id	label	tweet
0	1	0	@user when a father is dysfunctional and is s...
1	2	0	@user @user thanks for #lyft credit i can't us...
2	3	0	bihday your majesty
3	4	0	#model i love u take with u all the time in ...
4	5	0	factsguide: society now #motivation
5	6	0	[2/2] huge fan fare and big talking before the...
6	7	0	@user camping tomorrow @user @user @user @use...
7	8	0	the next school year is the year for exams.ð□□...
8	9	0	we won!!! love the land!!! #allin #cavs #champ...
9	10	0	@user @user welcome here ! i'm it's so #gr...

Requirements

There are some general library requirements for the project and some which are specific to individual methods. The general requirements are as follows.

- Numpy
- Scikit-learn
- Scipy
- Nltk

The library requirements specific to some methods are:

- Keras
- Tensorflow
- Transformers
- Xgboost

Model Training

Preprocessing

Preprocessing Twitter data involves several steps, including:

- Removing URLs, mentions, and hashtags
- Handling special characters, punctuation, and emoticons
- Tokenization: Splitting text into individual words or tokens
- Removing stopwords: Eliminating common words that carry little sentiment information
- Stemming or lemmatization: Reducing words to their base form

	id	label	tweet	Tidy_Tweets
0	1	0.0	@user when a father is dysfunctional and is s...	when father dysfunct selfish drag kid into dys...
1	2	0.0	@user @user thanks for #lyft credit i can't us...	thank #lyft credit caus they offer wheelchair ...
2	3	0.0	bihday your majesty	bihday your majesti
3	4	0.0	#model i love u take with u all the time in ...	#model love take with time
4	5	0.0	factsguide: society now #motivation	factsguid societi #motiv
5	6	0.0	[2/2] huge fan fare and big talking before the...	huge fare talk befor they leav chao disput whe...
6	7	0.0	@user camping tomorrow @user @user @user @use...	camp tomorrow danni
7	8	0.0	the next school year is the year for exams.ð□□...	next school year year exam think about that #s...
8	9	0.0	we won!!! love the land!!! #allin #cavs #champ...	love land #allin #cav #champion #cleveland #cl...
9	10	0.0	@user @user welcome here ! i'm it's so #gr...	welcom here

Popular NLP libraries such as NLTK (Natural Language Toolkit) can be utilized for efficient preprocessing.

Feature Extraction

Various features were extracted from the preprocessed text, such as bag-of-words, TF-IDF, or word embeddings.

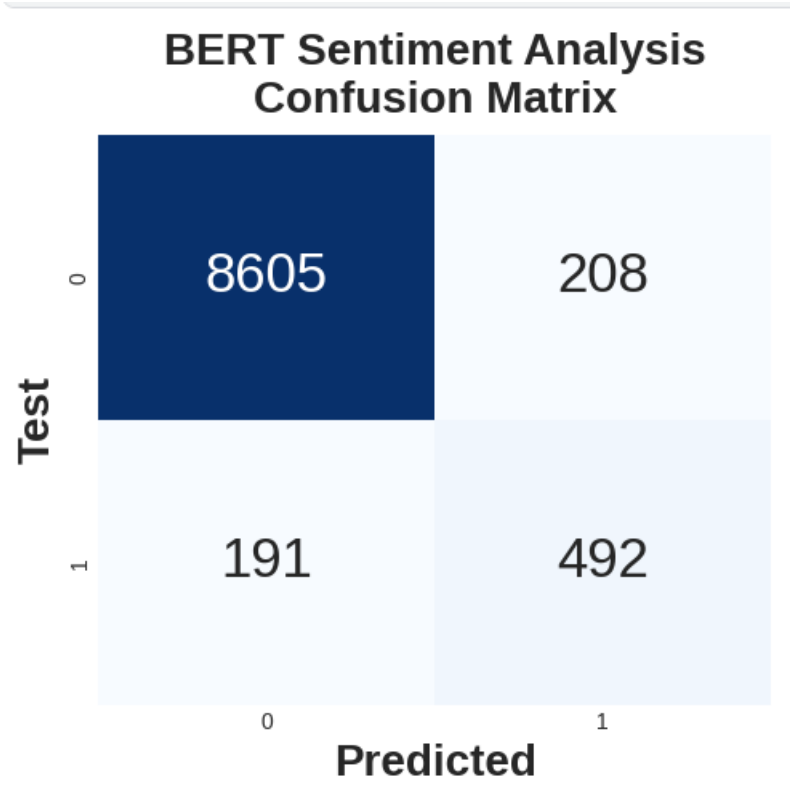
Model selection and training

Different machine learning algorithms were experimented with, including Logistic Regression, XGBoost , Decision tree, Recurrent Neural Networks, BERT . The models were trained on the labeled dataset.

Model Evaluation

The trained models were evaluated using appropriate evaluation metrics such as Confusion matrix, accuracy, precision, recall, and F1-score.

CONFUSION MATRIX



CLASSIFICATION REPORT

Classification Report for BERT:

	precision	recall	f1-score	support
0	0.98	0.98	0.98	8813
1	0.70	0.72	0.71	683
accuracy			0.96	9496
macro avg	0.84	0.85	0.84	9496
weighted avg	0.96	0.96	0.96	9496

Results

The trained sentiment analysis model achieved an accuracy of 96% on the evaluation dataset. However, the performance may vary depending on the specific dataset and the choice of machine learning algorithms.