Real-Time Emotion Detection using Kafka and ML

Authors

Ann Maria John, Divya Neelamegam, Kartik Mukkavilli, Poojitha Venkat Ram, Shruti Badrinarayanan

This document is a comprehensive README file containing all details about the project's data and code files. It includes the link to our GitHub repository. It has all the necessary prerequisites, package requirements, and setup instructions listed.

Introduction

The primary objective of the project is to develop a system capable of immediately recognizing and assessing the emotional context of textual input. This involves training and deploying machine learning models, including classification algorithms such as Naive Bayes, Support Vector Machines, and Random Forest, on a steady stream of Twitter data. To seamlessly handle the real-time flow of information, the trained model is integrated into a Spark Streaming application. The communication between the system's producer and consumer components is facilitated by Apache Kafka, which acts as the central messaging system. The project focuses on ensuring high throughput and low latency through fine-tuning the system architecture, leveraging Spark Streaming's parallel processing capabilities to effectively accommodate large-scale data streams.

Data Files

1. Raw Data (Folder) - Contains all raw data obtained from Kaggle.
2. Preprocessed Data (Folder) - Contains the saved preprocessed data as .npz files.
3. Offset (Folder) - Contains last processed Kafka offset in external_offset.txt file
4. Tweets_Input (Folder) - Contains the consumer Kafka messages in ''tweets.txt' file (unprocessed message)
5. Tweets_Output (Folder) - Contains the 'predicted_tweet.txt' final output.

Code Files

1. Data Preprocessing Notebook - *Data_Preprocessing.ipynb*
2. Modeling Notebooks
    a. Implementation of SVM with RBF kernel for high-accuracy classification - *SentimentAnalysisSVC.ipynb*
    b. Implementation of Naive Bayes - *Modeling-Naive Bayes.ipynb*
    c. Implementation of Random Forest - *SentimentAnalysisRF.ipynb*
3. GUI Application using PyQt - *EmotionDetectionApp.py*
4. Joblib files for the Saved Models
    a. *naive_bayes_model.joblib*

   b. *Random_forest_model.joblib*
   c. *svc_model.joblib*

5. Kafka Producer - *Emotion_kafka_producer.py*
6. Kafka Consumer - *Emotion_kafka_consumer.py*

Github
https://github.com/divneela/Real_Time_Emotion_Detection

Dataset
The dataset was chosen from Kaggle (https://www.kaggle.com/datasets/parulpandey/emotion-dataset/data). It is a collection of English Twitter messages designed for emotion recognition tasks. It is structured to reflect six basic emotions: anger, fear, joy, love, sadness, and surprise. The tweets were collected using the Twitter API by the authors, with a set of hashtags corresponding to eight basic emotions, adding anticipation and trust to the aforementioned six. The data consists of two features: “text” and “label”. The text is a tweet collected with the label being the corresponding emotion, the target variable.

Prerequisites
Before running this project, ensure you have the following installed:

- Python
- Pip/Anaconda/Conda for Package Management
- JupyterLab/Jupyter Notebook IDE
- PyQT
- Kafka

Package Requirements

- Scikit-learn
- Pandas (For reading data and data manipulation)
- NumPy (For data manipulation)
- Matplotlib, Seaborn (For Visualisations)

Kafka Setup
DOWNLOAD KAFKA 3.6.0 : https://kafka.apache.org/downloads

JDK Installation
```
brew install --cask adoptopenjdk/openjdk/adoptopenjdk8 export

JAVA_HOME=$(/usr/libexec/java_home -v1.8)
Java -version
```

Kickstart Zookeeper

```
sh bin/zookeeper-server-start.sh config/zookeeper.properties
```

Kickstart Kafka Server

```
sh bin/kafka-server-start.sh config/server.properties
```

TOPIC Creation

```
bin/kafka-topics.sh --create --bootstrap-server localhost:9092 --replication-factor 1
--partitions 1 --topic emotion-detection-stream sh bin/kafka-topics.sh --list
--bootstrap-server localhost:9092
```

PRODUCER IN CLI FOR TESTING

```
bin/kafka-console-producer.sh --broker-list localhost:9092 --topic
emotion-detection-stream
```

CONSUMER IN CLI FOR TESTING

```
bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --from-beginning
--topic emotion-detection-stream --partition 0
```