Emotion Detection Using Spelling Checker

Eyad Salman

May 7, 2024

1 Introduction

This project utilizes a common technique in Natural Language Processing called Sentiment Analysis for detecting emotion in a text using spelling checker where the expressed emotion is either positive, negative or neutral.

2 Dataset Description

The dataset used in this project consists of 997 rows and 2 columns. The columns include a ”text” column containing the textual data and an ”emotion” column indicating the sentiment associated with each text entry.

2.1 Preprocessing Steps

1. **Text Cleaning using Beautiful Soup**: Beautiful Soup library was utilized for text cleaning to remove any HTML tags or remove white spaces from text

2. **Removing Stop Words**: Stop words, such as ”the,” ”is,” ”and,” etc., were removed from the text data as they do not carry significant meaning for sentiment analysis.

3. **Removing Emojis**: Emojis were removed from text because the key part was to detect emotion from words instead of emojis in text.

3 Workflow

The workflow for the sentiment analysis project is as follows:

1. **Data Preprocessing**: Perform preprocessing steps including removing emojis, stop words, and text cleaning.

2. **Feature Engineering**: Convert the text data into numerical features suitable for machine learning models.

3. **Model Training**: Train the selected models on the preprocessed data.
4. **Model Selection**: Choosing appropriate machine learning and deep learning models for sentiment analysis.
5. **Model Evaluation**: Evaluating the performance of the trained models using appropriate evaluation metrics.
6. **Results Analysis**: Analyze the results and draw insights from the sentiment analysis.

4 Methodologies and Approaches

4.1 Feature Engineering

- Text Vectorization: Convert the text data into numerical vectors using techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) and Bag of Words.

4.2 Model Selection

- Logistic Regression
- Support Vector Machines (SVM)
- Multinomial Naive Bayes
- Random Forest Classifier
- Deep Learning Models (e.g ANN, RNN)

4.3 Evaluation Metrics

- Accuracy

5 Results

The performance of the sentiment analysis models is as follows:

Model	Accuracy
Logistic Regression	0.70
SVM	0.76
Random Forest Classifier	0.80
Multinomial Naive Bayes	0.67
ANN	0.72
RNN (LSTM)	0.91

6 Challenges and Resolution

One of the main challenges of was detecting accurate spellings for words not used mainly in English grammar like slang's. That's why many words were stored in ignored words dictionary. Over-fitting was another problem in RNN but it was solved by increasing input layers.

7 References

- Reddit: `https://www.kaggle.com/datasets/armitaraz/chatgpt-reddit`
- Youtube: `https://www.kaggle.com/datasets/reihanenamdari/youtube-toxicity-data`
- Twitter: `https://www.kaggle.com/datasets/yasserh/twitter-tweets-sentiment-dataset`