Journal of Current Research in Engineering and Science
Bi-Annual Online Journal (ISSN: 2581 - 611X)
VOLUME:8, ISSUE:2, AUGUST 2025, PAPER NUMBER:18

Multilingual Sentiment Analysis

Rachapally Pavan Kumar, Vellore Anand Kumar Sahil, Edgar Dcunha Ashley, Harshith Sai Chelmela, Sasishara Kashyap Chaturvedula, K. Anusha

Students and Faculty
Dept. of Computer Science Engineering,
Gokaraju Rangaraju Institute of Engineering and Technology
Hyderabad, India.
velloresahil2002@gmail.com

Abstract: People's sentiments are known to have a large impact on changes in stock prices, product sales, and trends. On the internet, people share their thoughts in many different languages. This is why it's important to create a way to understand how people feel in different languages when analyzing web text. Most of the text also includes emojis. So, it is important to take emojis into account when figuring out how people feel online. In our project, we are working on a method that can understand how people feel in many languages and even consider the emotions shown by emojis. This way, we can better understand what people are saying online, no matter what language they use. This will help us make better predictions and decisions in the field of sentiment analysis. Understanding the sentiments expressed by people across diverse languages and cultures is crucial in today's interconnected world. By incorporating emojis into sentiment analysis, we can capture nuances and emotions that may not be apparent through text alone. Emojis serve as a universal language, transcending linguistic barriers and providing valuable insights into the feelings and attitudes of online users.

Keywords: *Sentiment, linguistic, emotion*

1. INTRODUCTION:

In the era of globalization, the proliferation of social media, e-commerce, and digital communication has generated an unprecedented amount of text data in various languages. Understanding and analyzing this vast multilingual textual content has become crucial for businesses, researchers, and policymakers. Sentiment analysis, which involves determining the emotional tone behind a series of words, is a powerful tool in this context. It helps in gauging public opinion, monitoring brand reputation, enhancing customer service, and even predicting market trends.

Multilingual sentiment analysis aims to extend the capabilities of traditional sentiment analysis beyond a single language, allowing for a more comprehensive and inclusive understanding of global sentiments. This project focuses on developing and implementing advanced techniques for analyzing sentiments expressed in multiple languages, leveraging natural language processing (NLP) and machine learning.

By addressing the challenges of linguistic diversity and varying contextual nuances, this project seeks to contribute to the field of NLP by providing scalable solutions for multilingual sentiment analysis. The outcomes are expected to enhance the ability of organizations to understand and respond to sentiments expressed by a global audience, ultimately fostering better communication and more informed decision-making.

2. MULTILINGUAL SENTIMENT ANALYSIS:

This project is dedicated to developing a robust **multilingual sentiment analysis** system capable of processing and understanding text inputs in multiple languages, including Hindi and Telugu. By leveraging advanced natural language processing (NLP) techniques and machine learning models, the project aims to accurately determine the sentiment expressed in textual data across diverse languages. The system collects and preprocesses data from various sources such as social media, news platforms, and customer reviews, ensuring comprehensive coverage. It then employs language-agnostic models that are trained to predict sentiments consistently, irrespective of the language.

The project also focuses on evaluating the performance of these models to ensure high accuracy and reliability. Ultimately, the developed sentiment analysis system is designed to be applied in real-world scenarios, such as monitoring social media trends, analyzing customer feedback, and supporting market research, thereby enabling organizations to better understand and respond to multilingual sentiments.

3. ARCHITECTURE:

At first, we gather diverse text data from various sources such as social media platforms, news articles, and customer reviews in multiple languages, including Hindi and Telugu, using a web scraping tool. Then, we preprocess the collected data to clean and normalize it, ensuring it is ready for analysis. This involves tasks like text normalization, tokenization, and language detection. Subsequently, we use a powerful natural language processing framework to build and train multilingual sentiment analysis models. These models are designed to accurately predict the sentiment of the text, whether positive, negative, or neutral. After training, we evaluate the models' performance using a set of predefined metrics to ensure their accuracy and reliability across different languages and domains. The architecture of our project is depicted below.

Fig. 1. Architecture diagram for sentiment analysis

4.FLOWCHART:

Fig.2.Flowchart

1) First, we collect diverse text data from various sources such as social media platforms, news articles, and customer reviews, including texts in Hindi and Telugu, using web scraping tools and APIs.
2) Then, we preprocess the collected data to clean and normalize it, performing tasks like text normalization, tokenization, and language detection to ensure the data is suitable for analysis.
3) The sentiment results are stored in a database and visualized through interactive dashboards for real-time insights and analysis.
4) The sentiment analysis reports are automatically generated and sent to the respective stakeholders via email using an email automation tool.
5) Steps 3, 4, and 5 are repeated continuously as new data comes in to ensure up-to-date sentiment analysis.
6) At last, the bot execution stops.

Modules:

Data Collection and Preprocessing: This stage entails acquiring a large number of text samples in Hindi and English from a variety of sources, including social media, news articles, and product evaluations. The obtained data is then preprocessed to clean and standardize the text. Tokenization, stemming, and stop word removal are among the tasks assigned.

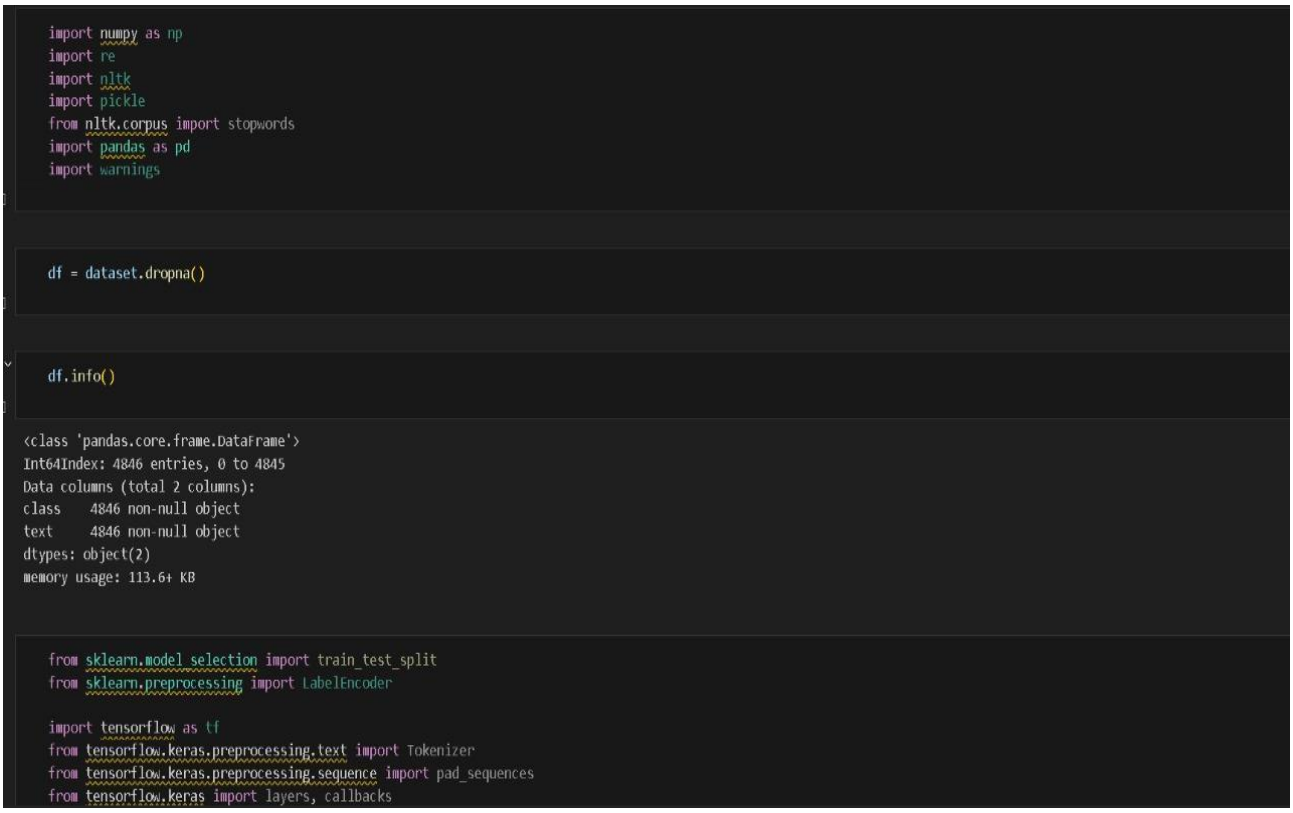
Language Identification: A language identification module is required to successfully handle Hindi and English texts. This component determines the language of each incoming text and routes it to the appropriate analytical pipeline for further processing.

Sentiment Analysis Model Training: This program would involve training sentiment analysis machine learning models in Hindi and English. To train accurate models, a sizable labeled dataset for every language would be needed. For this, methods like transformers and recurrent neural networks (RNNs), which are deep learning architectures, could be employed.

Emoji Analysis: In online communication, emojis are essential for conveying emotion. The task of locating, deciphering, and integrating emojis into the sentiment analysis process would fall under the purview of an emoji analysis module.

5. PICTORIAL REPRESENTATION:

Fig.3.Code



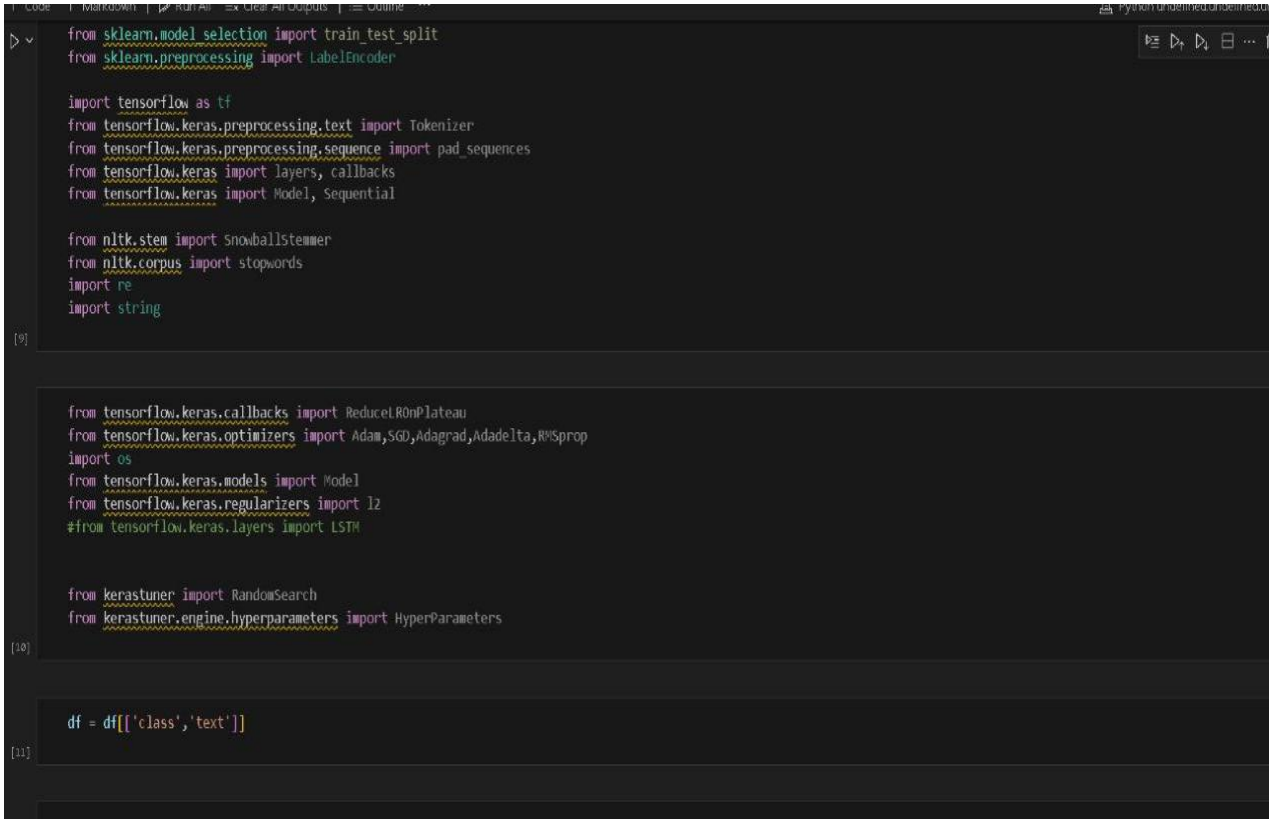
```
import numpy as np
import re
import nltk
import pickle
from nltk.corpus import stopwords
import pandas as pd
import warnings
```

```
df = dataset.dropna()
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4846 entries, 0 to 4845
Data columns (total 2 columns):
class    4846 non-null object
text     4846 non-null object
dtypes: object(2)
memory usage: 113.6+ KB
```

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder

import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras import layers, callbacks
```

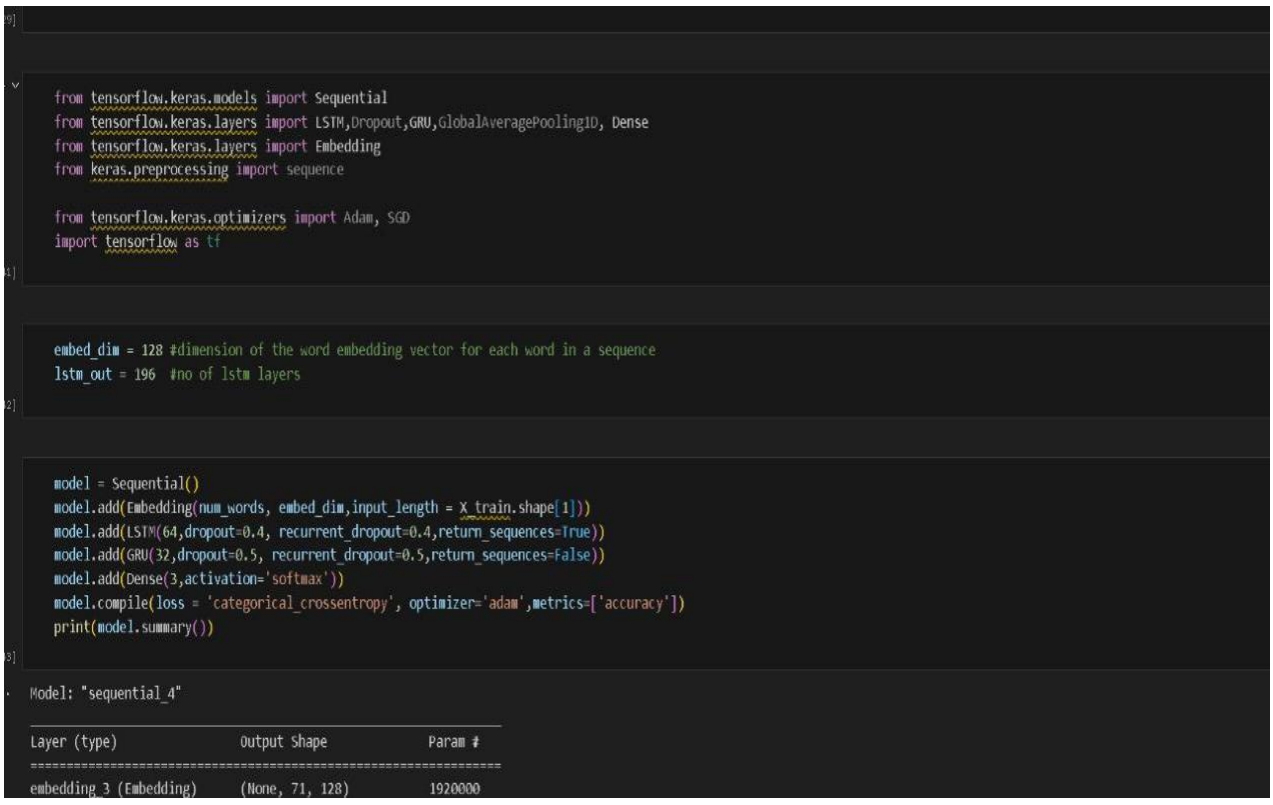


```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder

import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras import layers, callbacks
from tensorflow.keras import Model, Sequential

from nltk.stem import SnowballStemmer
from nltk.corpus import stopwords
import re
import string
```

```
from tensorflow.keras.callbacks import ReduceLROnPlateau
from tensorflow.keras.optimizers import Adam,SGD,Adagrad,Adadelta,RMSprop
import os
from tensorflow.keras.models import Model
from tensorflow.keras.regularizers import l2
#from tensorflow.keras.layers import LSTM


from kerastuner import RandomSearch
from kerastuner.engine.hyperparameters import HyperParameters
```

```
df = df[['class','text']]
```

Fig.4.Code



```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM,Dropout,GRU,GlobalAveragePooling1D, Dense
from tensorflow.keras.layers import Embedding
from keras.preprocessing import sequence

from tensorflow.keras.optimizers import Adam, SGD
import tensorflow as tf
```

```
embed_dim = 128 #dimension of the word embedding vector for each word in a sequence
lstm_out = 196  #no of lstm layers
```

```
model = Sequential()
model.add(Embedding(num_words, embed_dim,input_length = X_train.shape[1]))
model.add(LSTM(64,dropout=0.4, recurrent_dropout=0.4,return_sequences=True))
model.add(GRU(32,dropout=0.5, recurrent_dropout=0.5,return_sequences=False))
model.add(Dense(3,activation='softmax'))
model.compile(loss = 'categorical_crossentropy', optimizer='adam',metrics=['accuracy'])
print(model.summary())
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_3 (Embedding)      (None, 71, 128)           1920000
```

Fig.5.Code

6. SAMPLE OUTPUT:

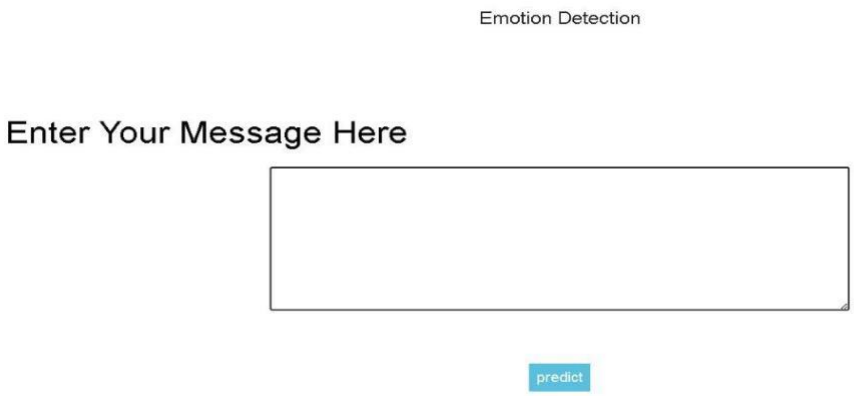


Fig.6.Sample Output

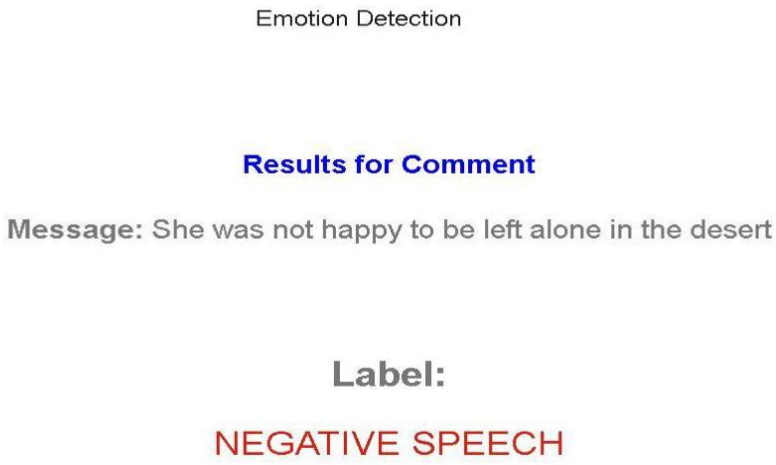


Fig.7.Sample Output

7. CONCLUSION:

While the project has achieved its primary objectives, we can improve the project by expanding the dataset that is the performance can be enhanced by increasing the size and diversity of the dataset. Experimenting with different machine learning algorithms, such as deep learning and fine-tuning hyperparameters can enhance the model's performance. Extending the project to handle text in

multiple languages would broaden its impact and explore techniques to perform emotion detection in real time, such as analyzing streaming data.

With these, the project can be further enhanced to provide more accurate and robust emotion detection capabilities, enabling its application in a wider range of real-world scenarios terminated after all the participants certificates are generated and they are distributed to each participant.

8. References:

Cambria, E., Poria, S., & Gelbukh, A. (2014). Emotion Analysis Using SenticNet. In Sentiment Analysis and Ontology Engineering (pp. 61-85). Springer.

Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. Proceedings of the 7th International Workshop on Semantic Evaluation (SemEval 2013), 321-327.

Zhang, X., & LeCun, Y. (2015). Text Understanding from Scratch. arXiv preprint arXiv:1502.01710.

Bravo-Marquez, F., Mendoza, M., Poblete, B., & Baeza-Yates, R. (2017). Large-scale comparison of sentiment classification methods using linguistic and gated recurrent neural networks. Information Retrieval Journal, 20(5-6), 559-580.

Poria, S., Cambria, E., Hazarika, D., Majumder, N., Zadeh, A., & Morency, L. P. (2017). Context- dependent sentiment analysis in user-generated videos. In Proceedings of the 2017 ACM on Multimedia Conference (pp. 709-717). ACM.