Project Presentation

on

Sentiment Analysis And Opinion Mining Using LLM

Presented By:

Kashish Gupta (2021021140)
Pragati Yadav (2021021152)

Computer Science and Engineering Department

Madan Mohan Malaviya University of Technology, Gorakhpur

Content

Abstract

The Project's Goal:

This project addresses the critical need for accurate sentiment analysis in a world overwhelmed by textual data. By leveraging BERT the system achieves a nuanced understanding of text, far surpassing traditional methods. Our tool performs real-time predictions and bulk analysis, offering sentiment classifications (Positive, Negative, Neutral) with high contextual accuracy. It finds applications in industries such as e-commerce, social media monitoring, and customer feedback systems.

Model used :

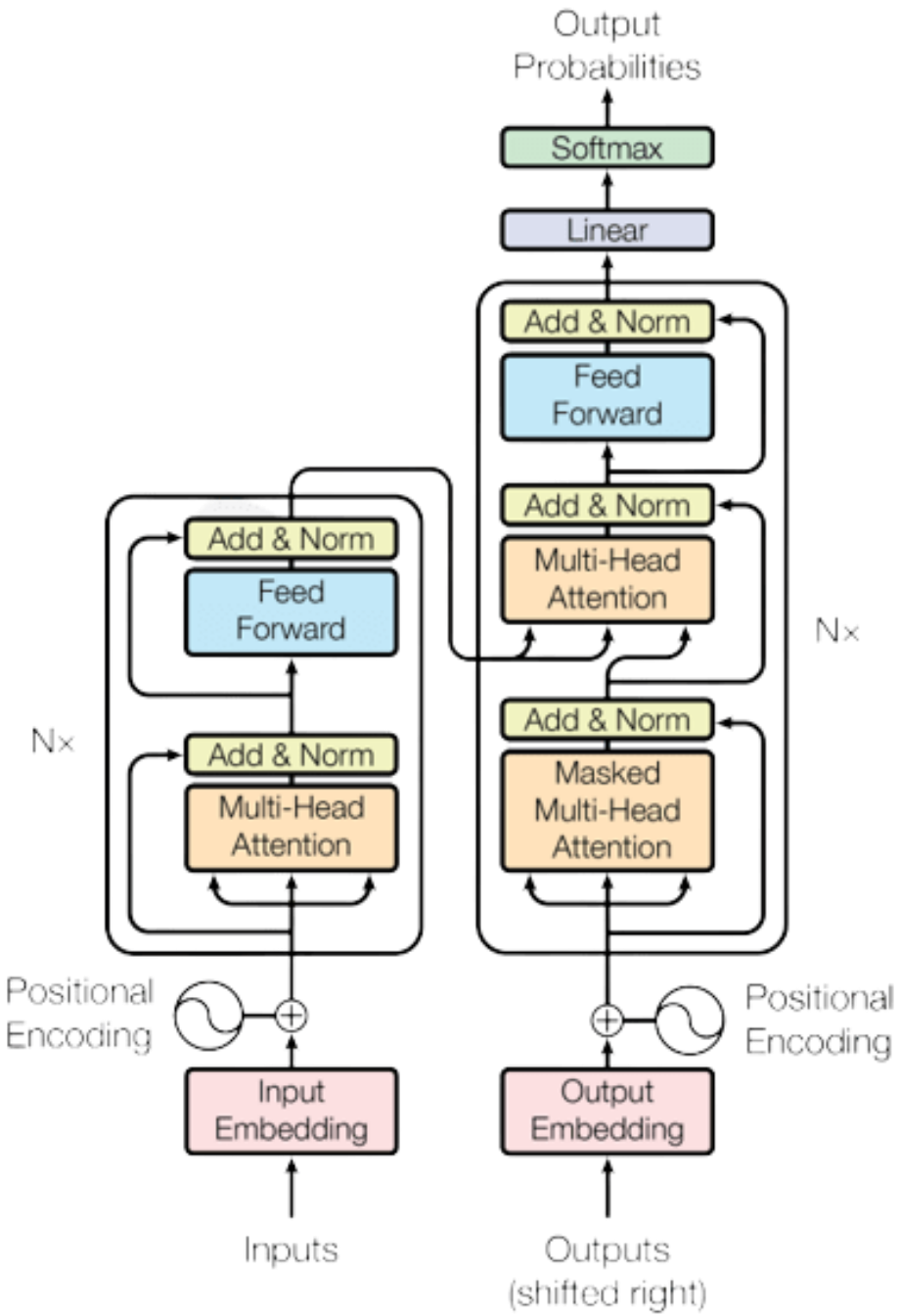
- BERT (Bidirectional Encoder Representations from Transformers): Pre-trained transformer-based model known for its bidirectional context understanding. Fine-tuned specifically for sentiment classification tasks using the IMDB dataset and some public datasets.
- **Softmax Activation**: Converts raw logits into class probabilities for sentiment predictions.

Objective

- **Achieve Contextual Precision**: Exploit BERT's bidirectional transformer architecture for sentiment predictions that consider the entire context of a sentence, unlike unidirectional models.
- **Real-World Scalability**: Create a system that works seamlessly for individual inputs and large datasets, ensuring flexibility across domains.
- **Insight-Driven Decision Making**: Equip users with actionable insights from textual data, fostering informed decision-making in marketing, product development, and customer experience.
- **User-Friendly Deployment**: Offer a streamlined interface that combines advanced machine learning with accessibility, enabling non-technical users to perform sentiment analysis effortlessly.

INTRODUCTION

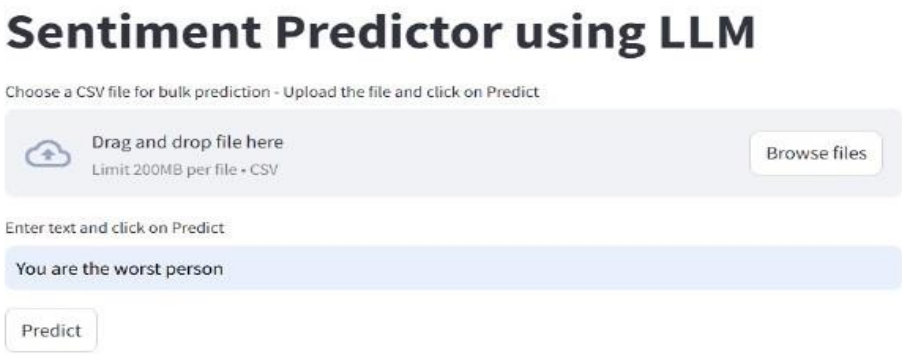
- **Understanding Sentiment Analysis**: Sentiment analysis involves determining the emotional tone behind a body of text, essential for interpreting opinions and attitudes.
- Large Language Models (LLMs) like ChatGPT are increasingly being used in Sentiment Analysis due to their ability to understand and generate human-like text. These models are trained on vast amounts of data, allowing them to generate coherent and contextually relevant responses.
- The model used in this project, BERT, adopts the **encoder** part of the Transformer. The encoder uses **Multi-Head Self-Attention** to capture contextual relationships, **Feed-Forward Neural Networks** to refine token embeddings, **Positional Encoding** to retain the order of words in a sentence.

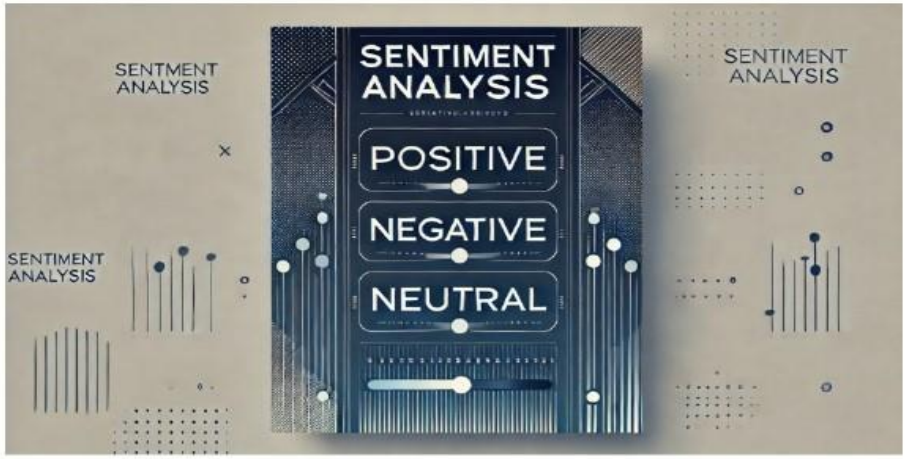

Output Probabilities
Softmax
Linear
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Add & Norm
Masked Multi-Head Attention
Nx
Add & Norm
Feed Forward
Add & Norm
Multi-Head Attention
Nx
Positional Encoding
Positional Encoding
Input Embedding
Output Embedding
Inputs
Outputs (shifted right)

Description of Software structure

FRONTEND / USER INTERFACE:

- The frontend of the sentiment analysis tool is designed to ensure an intuitive and seamless user experience, enabling users to interact with the system effortlessly. The interface is built using **Streamlit**, a Python library known for its simplicity and rapid development capabilities. Input can be taken as a single statement or bulk.
- **Single Text Analysis**: Users can enter any text (e.g., "The product quality is amazing!") for instant sentiment prediction.
- **Bulk Prediction**: CSV files with hundreds of reviews can be uploaded, enabling large-scale sentiment classification.





BACKEND :

- **Preprocessing**: Text is tokenized using the **BertTokenizer**, of transformers library. Each tokenized sequence is padded or truncated to a fixed length of 512 for compatibility with the model.
- **Model Architecture**: The core is a pre-trained **Bert-base-uncased model**, fine-tuned for sentiment classification tasks.
- **Prediction**: The model outputs logits (raw scores) for each sentiment class, converted to probabilities using the softmax function.
- **Visualization**: Results are presented interactively using **Plotly**, showcasing sentiment distributions for clear insights.

DATA COLLECTION

The dataset used : 1) "imdb" dataset for Binary Sentiment classification
2) Kaggle public dataset for muti-class Sentiment classification

"imdb"

- Its size : 25,000
- Input Column: 2
- Output Labels: 1

"public dataset"

- Its size : 27,481
- Input Column: 2
- Output Labels: 1

Optimizing Parameters

- **Learning rate-** for “imdb” – 2e-5 and for “public dataset” – 1e-5.
- **Hyperparameter Tuning**: Experimentation with batch sizes (6 and 4) revealed that smaller batches improved gradient updates and maintained performance.
- **Epoch –** The model was trained multiple times to determine the optimal epoch number, ensuring a balance between underfitting and overfitting.
- **Handling Imbalanced Data**: Weighted loss metrics ensure minority classes like neutral reviews are adequately learned by the model.

TECHNOLOGIES USED

Data Collection Tool :

- Kaggle : Kaggle is an online platform that provides data science competitions, datasets, and a collaborative environment for data science.

Libraries :

- **Streamlit (st):** This is used to create the web application. It helps display things like text, buttons, images, and other components on a webpage.
- **Pandas (pd), numpy:** This is used for reading and processing data, especially from CSV files.
- **Transformers & TensorFlow:** These are used for loading a pre-trained machine learning model (BERT) to perform sentiment analysis.

OTHER TOOLS:

- Github
- Kaggle
- Google Colab

WORKING

- **Initial Data Handling and Input Preparation**: The backend begins with receiving input from the user, either as single text entries or bulk data uploaded via a CSV file**.**
- The core of the backend is the pre-trained BERT model. This model is loaded using the Hugging Face Transformers library. Specifically, the TFBertForSequenceClassification class is used, which is pre-configured for text classification tasks. This model has been fine-tuned on a sentiment-labeled dataset (e.g., IMDB reviews) to specialize in distinguishing positive, negative, and neutral sentiments.
- It internally uses a prediction function that loads the data and generates predictions.
- If the input is a single sentence, the model simply returns its sentiment. For bulk files, it processes all sentences and provides the percentage distribution across classification classes.
- **Optimizer** – Adam Optimizer
- **Loss Function** - SparseCategoricalCrossentropy

PERFORMANCE METRICS

“imdb”-

Accuracy on test dataset: 90.98%

Accuracy on valid dataset: 97.72%

“public dataset”-

Accuracy on test dataset: 81.76%

Accuracy on valid dataset: 76.48%

CONCLUSION

This project demonstrated the power and versatility of **transformer-based models**, specifically **BERT**, in performing **sentiment analysis**. By leveraging BERT's bidirectional context understanding, we created a tool capable of discerning complex sentiments. The combination of **pre-processing techniques**, **model fine-tuning**, and an **interactive frontend** ensures that the solution is both effective and user-friendly.

- **Key Achievements**: Implemented state-of-the-art sentiment analysis with 97.72% accuracy on "imdb" dataset. Integrated user-friendly features like CSV upload, real-time feedback, and interactive visualizations.
- **Impact**: Empowers businesses to make data-driven decisions based on customer sentiment. Enhances trust in reviews by providing consistent and reliable predictions.

REFRENCES

- **Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł., & Polosukhin, I. (2017).** Attention is all you need. *Proceedings of NeurIPS 2017*. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf
- Public Dataset - https://www.kaggle.com/datasets/abhi8923shriv/sentiment-analysis-dataset/code
- **Hugging Face. (2021).** Transformers Documentation. *Hugging Face*. Retrieved from https://huggingface.co/docs/transformers
- **Streamlit. (2021).** Streamlit Documentation. *Streamlit*. Retrieved from https://docs.streamlit.io/
- **Plotly. (2021).** Plotly Python Graphing Library. *Plotly*. Retrieved from https://plotly.com/python/
- **Google Colab. (2021).** Google Colab Documentation. *Google Colab*. Retrieved from https://colab.research.google.com/

THANK YOU