Automate Detection of Different Emotions from Textual Comments and Feedback

Industry Project – TCS iON Digital Learning

This project focuses on building an AI-based system that automatically detects emotions from textual comments and feedback using deep learning and Natural Language Processing (NLP) techniques.

Submitted by:
Jenas Renjan Parakkal

Platform:
TCS iON Digital Learning

Year: 2026

College:
Yenepoya Deemed To Be University

1. Introduction

In the modern digital era, people express their opinions, feelings, and experiences through online platforms such as social media, product review websites, blogs, and feedback forms. These textual comments contain valuable emotional information that can help organizations understand customer satisfaction and public opinion. However, manually analyzing such a large volume of text data is time-consuming and inefficient.

To solve this problem, Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques can be used to automatically detect emotions from text. This project focuses on building an AI-based emotion detection system that can classify emotions from textual comments and feedback. The system uses a transformer-based deep learning model to accurately understand the context and sentiment of the input text.

2. Problem Statement

Organizations receive thousands of textual comments and feedback every day. Analyzing this feedback manually is not practical and may lead to delays and errors. There is a need for an automated system that can efficiently process text data and identify the underlying emotions. The challenge lies in understanding context, sarcasm, and variations in language. This project aims to address these challenges by developing an automated emotion detection model using deep learning techniques.

3. Objectives of the Project

The main objectives of this project are:

- To collect and preprocess textual data for emotion detection
- To build a deep learning model using transformer architecture
- To classify emotions such as positive, negative, and neutral
- To calculate confidence scores for each prediction
- To deploy the trained model using a simple and user-friendly interface

4. Scope of the Project

The scope of this project is limited to emotion detection from English textual comments. The system is designed to classify emotions based on text input provided by the user. The model is deployed locally using Streamlit and provides real-time predictions along with confidence scores. The project can be extended in the future to support multiple languages and additional emotion categories.

---

5. Technology Stack Used

The following technologies and tools were used in this project:

- **Programming Language:** Python
- **Deep Learning Framework:** PyTorch
- **NLP Library:** Hugging Face Transformers
- **Model Used:** BERT (Bidirectional Encoder Representations from Transformers)
- **Machine Learning Tools:** Scikit-learn
- **Frontend Framework:** Streamlit
- **Development Environment:** Google Colab and Local Machine

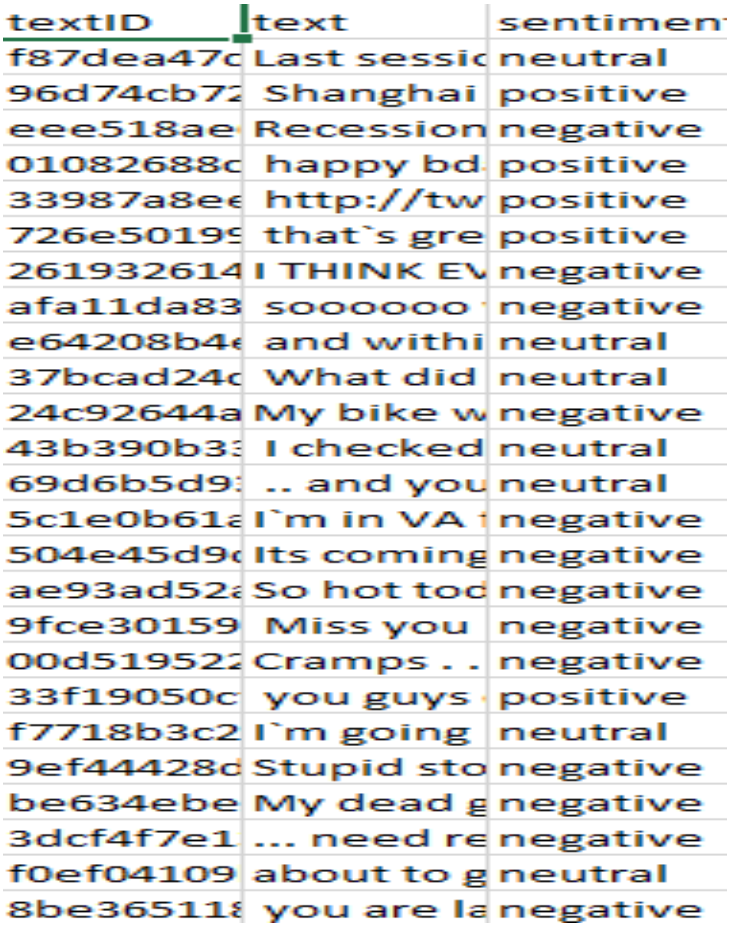
6. Dataset Description

The dataset used in this project consists of textual comments along with sentiment labels such as positive, negative, and neutral. The dataset was used to train, validate, and test the emotion detection model. Each record in the dataset contains a text sentence and its corresponding sentiment label. The dataset was explored to understand its structure and distribution before training the model.

---

7. Data Preprocessing

Before training the model, the dataset was preprocessed to improve model performance. Textual data was cleaned by removing unnecessary characters, URLs, and extra spaces. The text was normalized to maintain consistency. Sentiment labels were encoded into numerical values using LabelEncoder. The dataset was then split into training, validation, and test sets.



textID	text	sentimen
f87dea47c	Last sessic	neutral
96d74cb72	Shanghai	positive
eee518ae	Recession	negative
01082688c	happy bd	positive
33987a8ee	http://tw	positive
726e50195	that`s gre	positive
261932614	I THINK EV	negative
afa11da83	soooooo	negative
e64208b4e	and withi	neutral
37bcad24c	What did	neutral
24c92644a	My bike w	negative
43b390b33	I checked	neutral
69d6b5d93	.. and you	neutral
5c1e0b61a	I`m in VA	negative
504e45d9c	Its coming	negative
ae93ad52a	So hot toc	negative
9fce30159	Miss you	negative
00d519522	Cramps . .	negative
33f19050c	you guys	positive
f7718b3c2	I`m going	neutral
9ef44428c	Stupid sto	negative
be634ebe	My dead g	negative
3dcf4f7e1	... need re	negative
f0ef04109	about to g	neutral
8be365118	you are la	negative

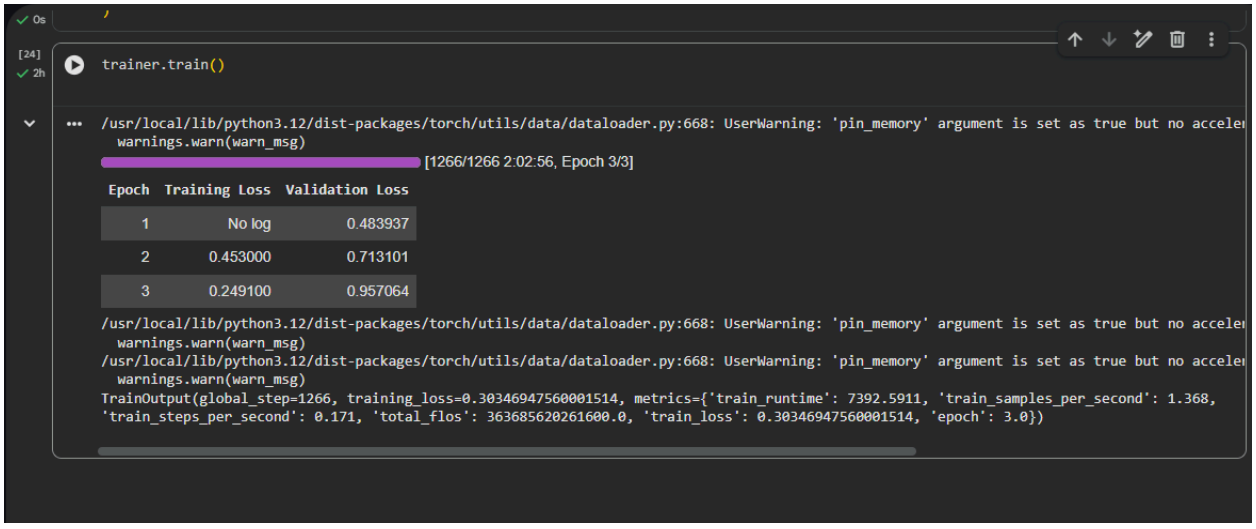
8. Model Development

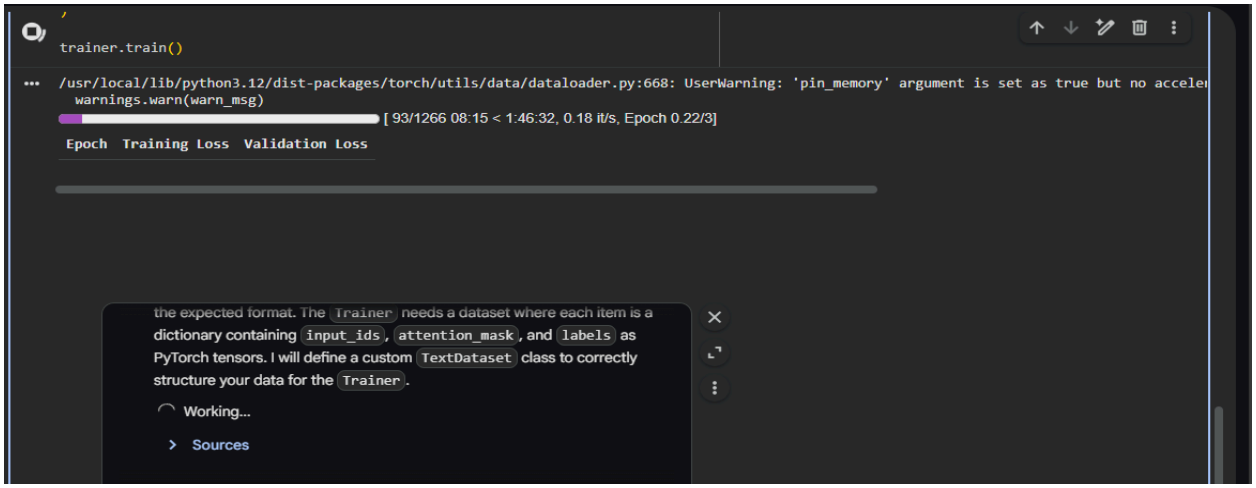
A pre-trained BERT model was selected for emotion classification due to its strong performance in understanding contextual information. Transfer learning was applied by fine-tuning the BERT model on the emotion dataset. This approach reduces training time and improves accuracy. The model architecture includes a tokenizer for text processing and a classification head for predicting emotions.

---

9. Training and Validation

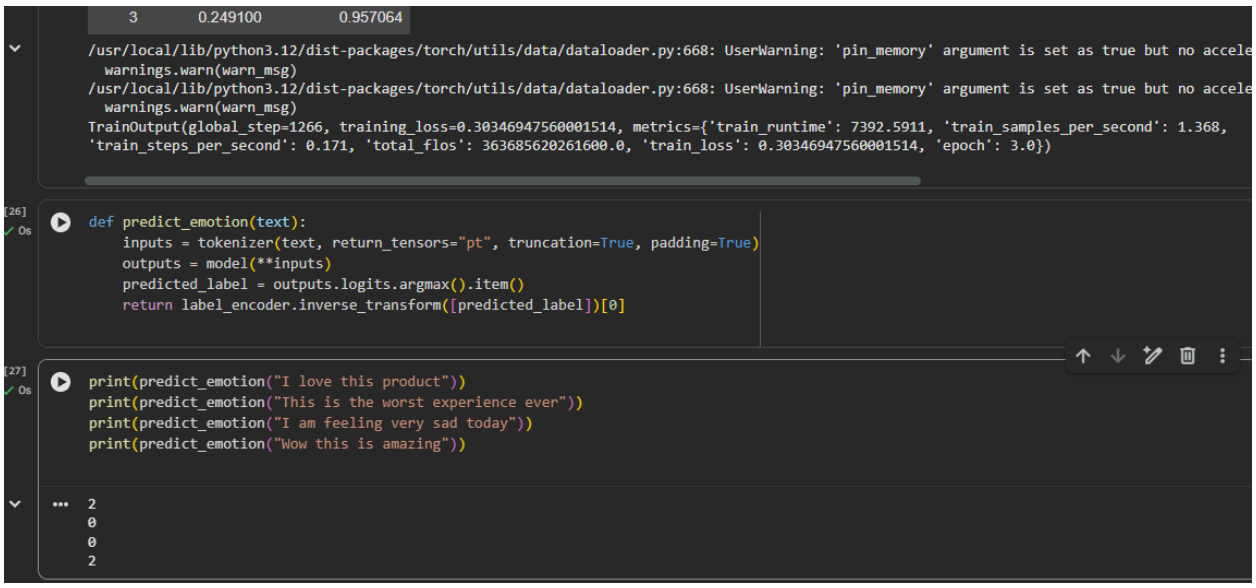
The model was trained using the training dataset and evaluated using the validation dataset. Multiple epochs were used to improve learning. During training, training loss and validation loss were monitored to evaluate model performance. The model showed stable learning behavior and achieved good performance across epochs.





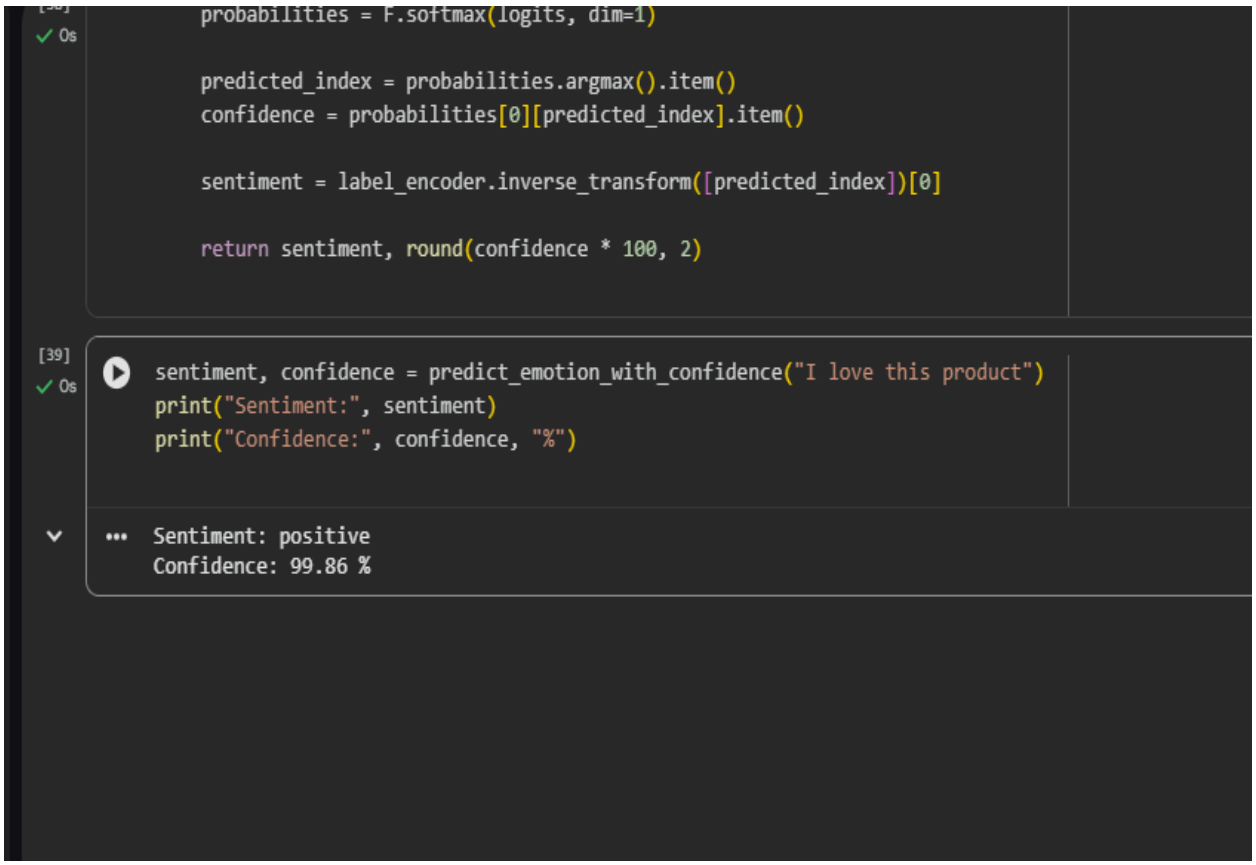
10. Model Testing

After training, the model was tested using unseen text samples. The model successfully predicted emotions for different types of sentences. Testing helped verify that the model can generalize well to new input data and perform reliably in real-world scenarios.



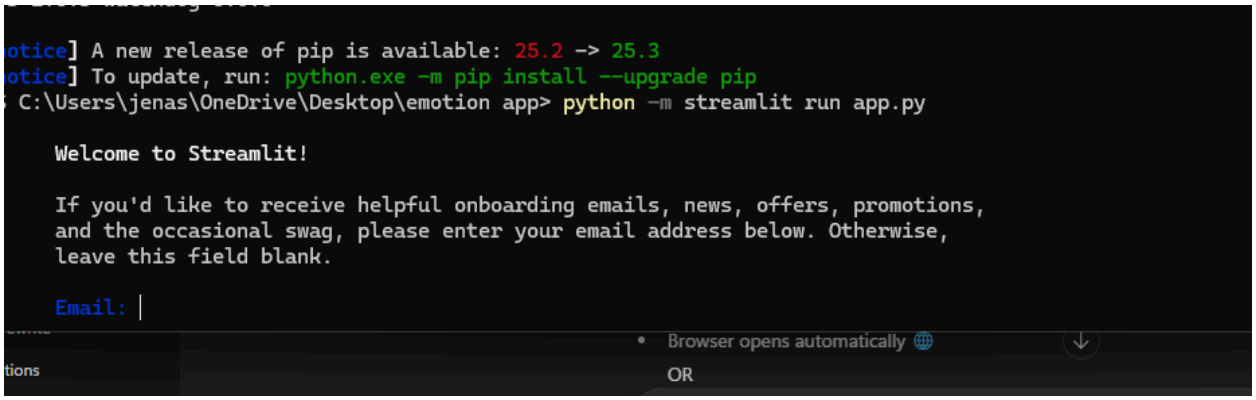
11. Confidence Score Calculation

To provide transparency in predictions, confidence scores were calculated using the Softmax function. The Softmax function converts raw model outputs into probability values. The emotion with the highest probability is selected as the final prediction, and the corresponding probability is displayed as the confidence score.



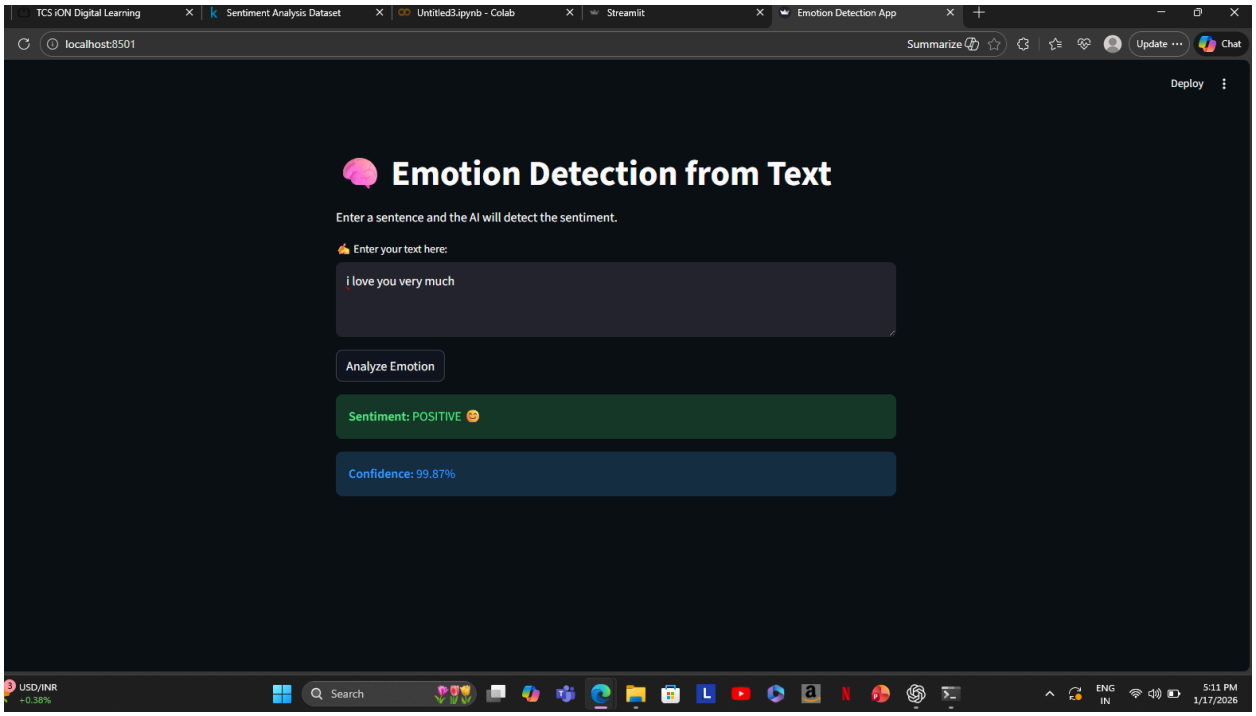
12. Frontend Development Using Streamlit

A Streamlit-based web application was developed to deploy the trained model. The frontend provides a text input field where users can enter a sentence or paragraph. After clicking the analyze button, the predicted emotion and confidence score are displayed. The interface is simple, interactive, and easy to use.



13. Results

The system successfully detects emotions from text with high confidence. The model accurately classifies positive and negative sentiments and provides confidence scores above 90% for most test cases. The results demonstrate the effectiveness of transformer-based models in emotion detection tasks.



14. Advantages of the System

- Automated and fast emotion detection
- User-friendly interface
- High accuracy and confidence scores
- Real-time emotion prediction

---

15. Limitations

- Works only for English text
- Accuracy depends on dataset quality
- Requires computational resources for training

---

16. Future Enhancements

The system can be enhanced in the future by adding multilingual support, detecting emotion intensity, integrating social media platforms, and using explainable AI techniques to improve model transparency.

17. Conclusion

This project successfully demonstrates the use of deep learning and NLP techniques for emotion detection from textual comments and feedback. By using a transformer-based BERT model and deploying it through a Streamlit application, the system provides accurate emotion classification along with confidence scores. The project has strong potential for real-world applications in customer feedback analysis and social media monitoring.