Final Year B.Tech. Project-II Report

On

**AI-Powered Therapy Analytics for Mental Health Using Sentiment Analysis**

*For the Degree of*

**Bachelor of Technology**

*In*

**Computer Science and Engineering**

Submitted By

21510016: Mr. Anurag Takalkar

21510017: Mr. Onkar Yemul 21510042: Mr. Omkar Auti

22520012: Mr. Akash Metkari

Guided By

**Prof. Manik Chavan**

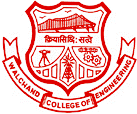


**Department of Computer Science and Engineering**

**Walchand College of Engineering, Sangli**

(*An Autonomous Institute*)

**AY 2024-25**

****

**Walchand College of Engineering, Sangli.**

(*An Autonomous Institute*)

**Department of Computer Science and Engineering**

**CERTIFICATE**

This is to certify that the Project Report entitled, **AI-Powered Therapy Analytics for Mental Health Using Sentiment Analysis**

submitted by

21510016: Mr. Anurag Takalkar

21510017: Mr. Onkar Yemul 21510042: Mr. Omkar Auti

22520012: Mr. Akash Metkari

to **Walchand College of Engineering, Sangli**, India, is a record of bonfire Project work of course **“PROJECT II - (6CS492)”** carried out by Onkar Yemul, Anurag Takalkar, Omkar Auti, Akash Metkari under my supervision and guidance and is worthy of consideration for the award of the degree of Bachelor of Technology in Computer Science & Engineering during the academic year ***2024-25***.

|  |  |  |
| --- | --- | --- |
| **Prof. M. K. Chavan**  Guide |  | External Examiner |
| **Mrs. Dr. M. A. Shah**  Head of Department  (Computer Science and Engineering) | | | |

**Declaration**

We hereby declare that work presented in this project report titled **" AI-Powered Therapy Analytics for Mental Health Using Sentiment Analysis "** submitted by us in the partial fulfilment of the requirement of the award of the degree of Bachelor of Technology (B.Tech.) Submitted in the Department of Computer Science & Engineering, Walchand College of Engineering, Sangli, is an authentic record of my project work carried out under the guidance of Prof. M. K. Chavan.

21510016: Mr. Anurag Takalkar

21510017: Mr Onkar Yemul

21510042: Mr Omkar Auti

22520012: Mr. Akash Metkari

Date :

Place: Sangli.

**Acknowledgement**

Firstly, we express our sincere gratitude to **Prof. M. K. Chavan**, the supervisor of this project, for his invaluable guidance, encouragement, and expertise throughout the research process. His mentorship has been instrumental in shaping our approach and driving us toward the successful completion of this project.

We are deeply thankful to the faculty members of the **Computer Science and Engineering Department** for their continuous support and insightful feedback, which have directly or indirectly enriched the quality of this report. Special appreciation is extended to **Dr. M. A. Shah**, the Head of the Department, for her constant motivation and assistance in navigating the administrative aspects of this project.

Our heartfelt thanks go to our **family, friends, and peers** for their unwavering support, patience, and encouragement throughout the course of this project. Their belief in us served as a great source of strength during challenging times.

Finally, we are grateful to our **institution** for providing us with the opportunity and an environment conducive to learning and innovation. This project has not only enhanced our technical and managerial skills but also instilled in us a sense of responsibility, adaptability, teamwork, and professional ethics that will guide us in our future endeavours.

**Abstract**

Mental health is a critical determinant of an individual's overall well-being, influencing physical health, cognitive functioning, and social interactions. According to the World Health Organization (WHO), nearly 970 million people worldwide suffer from mental disorders, with depression and anxiety being the most prevalent. Studies indicate that mental health disorders contribute to 14.3% of global deaths annually, either directly or through related physical health conditions. Furthermore, untreated mental health conditions can reduce life expectancy by 10 to 20 years, emphasizing the urgent need for early detection and intervention.

This research leverages Natural Language Processing (NLP) and Sentiment Analysis to assess emotional states by analysing Cognitive Behavioural Therapy (CBT) audio recordings and textual inputs. Research suggests that social media behaviour and speech patterns can predict depressive symptoms with up to 80% accuracy, making linguistic and acoustic analysis a valuable tool for mental health assessment. Machine learning models process CBT session transcripts and audio features to extract emotional cues, helping individuals gain deeper insights into their mental well-being.

To enhance accessibility and user engagement, the results are visualized on an interactive dashboard, providing real-time mental health insights based on audio and text analysis. Users can track emotional trends, monitor progress, and receive data-driven feedback to support self-improvement strategies.

Despite the technological advancements, privacy concerns remain a key challenge. Studies show that over 60% of individuals are apprehensive about the misuse of personal health data. To address this, the proposed system integrates end-to-end encryption, federated learning models, and anonymization techniques to ensure data security while enabling effective mental health assessments.

This study aims to bridge the gap between mental health awareness and early detection by combining CBT-driven AI analytics with interactive visualization tools. By integrating machine learning with behavioural data analysis, this research contributes to personalized, data-driven mental health interventions, ultimately fostering improved psychological well-being and quality of life.

***Keywords:***

***Sentiment Analysis, Mental Health, BERT, FL-BERT+DO, Deep Learning, Text Classification, CNN, LSTM, Federated Learning, Visualization Dashboards***

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Figure names** | **Page no.** |
| Fig 1 | Distribution of Emotions of Given Dataset | 12 |
| Fig 2 | Pie-Chart representing Percentage of statements of Emotions across Dataset | 13 |
| Fig 3 | Distribution of Text Length vs Count | 14 |
| Fig 4 | Distribution of Characters per statement | 14 |
| Fig 5 | Average of words per statement | 15 |
| Fig 6 | Word cloud of various Emotions | 15 |
| Fig 7 | Architectural Diagram of CNN | 17 |
| Fig 8 | CNN Model Accuracy vs Epoch | 18 |
| Fig 9 | CNN Model Loss vs Epoch | 19 |
| Fig 10 | CNN Confusion Matrix | 19 |
| Fig 11 | Architectural Diagram LSTM | 20 |
| Fig 12 | Confusion Matrix LSTM | 21 |
| Fig 13 | LSTM Model Loss vs Epoch | 22 |
| Fig 14 | Architectural Diagram of CNN + LSTM | 22 |
| Fig 15 | Confusion Matrix – Ensemble CNN + LSTM | 24 |
| Fig 16 | Ensemble CNN + LSTM Accuracy vs Epoch | 24 |
| Fig 17 | Ensemble CNN + LSTM Loss vs Epoch | 25 |
| Fig 18 | Architectural diagram of BERT | 26 |
| Fig 19 | Confusion Matrix – BERT | 26 |
| Fig 20 | Architectural diagram of FL-BERT with Dropout Optimization | 28 |
| Fig 21 | Confusion Matrix – FL-BERT with Dropout Optimization | 29 |
| Fig 22 | Use Case Diagram (Dashboard) | 30 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No.** | **Table Name** | **Page No.** |
| Table 1 | Emotions vs Count | 12 |
| Table 2 | Different Techniques of Sentiment Analysis with strength and Limitations | 16 |
| Table 3 | Evaluation Metrics used over Models | 34 |
| Table 4 | Accuracy Obtained over Models | 35 |

**CONTENTS**

|  |  |
| --- | --- |
| **TOPICS** | **Page No**. |
| **CHAPTER 1: INTRODUCTION**   * 1. **Motivation**   2. **Objective**   3. **Problem Statement** | 8  9  10 |
| **CHAPTER 2: LITERATURE REVIEW** | 11 |
| **CHAPTER 3: PROPOSED METHODOLOGY**   * 1. **Dataset Description**   2. **Data Preprocessing**      1. **Text Cleaning**      2. **Text Normalisation**      3. **Word Frequency Analysis**   3. **Algorithmic Methodology**      1. **Why do we require algorithm for sentiment analysis ?**      2. **Algorithms and Their Mechanisms / Architecture with Sentiment Analysis** | 12  12  13  16 |
| **CHAPTER 4: IMPLEMENTATION**   * 1. **Key Features**. | 31 |
| **CHAPTER 5: Application**   * 1. **Application in medical field**   2. **Application in non-medical field** | 33  33 |
| **CHAPTER 6: RESULTS AND ANALYSIS** | 34 |
| **CHAPTER 7: CONCLUSION AND FUTURE SCOPE** | 36 |
| **CHAPTER 8: REFERENCES** | 37 |

**1.Introduction**

Mental health has become a growing concern in today’s fast-paced and digitally driven world, with increasing cases of stress, anxiety, depression, and other psychological disorders being reported globally. The growing volume of user-generated textual data on digital platforms provides an opportunity to explore linguistic patterns associated with mental health conditions. Sentiment analysis and emotion detection techniques, powered by Natural Language Processing (NLP), have emerged as effective tools for understanding psychological states through text data.

Recent advancements in Deep Learning and NLP have significantly improved the ability to classify and interpret emotional and psychological content. Models such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and transformer-based architectures like BERT (Bidirectional Encoder Representations from Transformers) have demonstrated enhanced performance in mental health classification tasks. Furthermore, hybrid models combining CNN and LSTM and optimized variants like Fine-Tuned BERT with Dropout Optimization (FL-BERT + DO) offer greater accuracy and contextual understanding.

In addition to classification, the integration of interactive dashboards, AI-powered session analysis tools, and client progress visualization platforms has further enabled the practical application of these technologies in therapeutic environments. Ethical considerations such as data privacy, model bias mitigation, and fairness are also critical to ensure responsible deployment of such systems in real-world mental health care settings.

* 1. **Motivation:**
* **Rising Mental Health Challenges:** In today’s fast-paced and digitally connected world, mental health issues such as depression, anxiety, stress, and suicidal thoughts are becoming increasingly common. This project is motivated by the need to proactively address these growing psychological concerns using technology.
* **Lack of Early Detection Systems:** Mental health conditions often go undiagnosed due to the lack of timely detection and professional support. This project aims to fill that gap by using machine learning and NLP techniques to identify early signs of mental health disorders through textual data.
* **Need for Technology-Driven Psychological Support:** Traditional therapy approaches alone are often not scalable. By leveraging AI-powered classification models and interactive dashboards, we aim to enhance support systems available to both therapists and individuals seeking help.
* **Providing Meaningful Insights from Everyday Communication**: People often express their emotions through messages, social media posts, or journals. This project uses such data to uncover hidden emotional patterns and psychological conditions, thereby offering deeper insights into mental well-being.
* **Improving Clinical Decision-Making and Therapy Outcomes**: Through visual dashboards, session analysis modules, and automated documentation tools, the project supports mental health professionals in making informed decisions, monitoring progress, and enhancing therapy outcomes.
* **Bridging the Gap Between Data and Actionable Solutions**: A major problem today is the abundance of mental health-related data without actionable insights. This project bridges that gap by transforming raw textual data into meaningful classifications and visualizations that drive real-world interventions.
* **Ensuring Ethical and Inclusive Mental Health Solutions:** The motivation also lies in creating responsible AI systems that ensure data privacy, reduce bias in model predictions, and promote inclusivity in mental health care delivery**.**
  1. **Objective:**
* **To Address the Medical Challenge and Enhance Mental Health Assessment through AI**  
  - To address the growing concern of mental health issues by leveraging AI-driven techniques that aid in early detection, diagnosis, and therapeutic monitoring.  
  - To contribute toward a scalable solution that bridges the gap in mental health diagnostics and clinical support through automated classification and visualization tools.
* **To Acquire and Preprocess a Comprehensive Mental Health Dataset**  
  - To acquire and analyze a curated Kaggle dataset, integrated from multiple mental health-related sources to ensure diversity and comprehensiveness.  
  - To preprocess text data through techniques such as noise removal, tokenization, lemmatization, and stemming to prepare high-quality inputs for analysis.  
  - To perform Exploratory Data Analysis (EDA) to identify data distribution, class imbalance, and linguistic characteristics of mental health categories.  
  - To conduct word frequency analysis using statistical and visual techniques (e.g., word clouds, frequency ranking) to determine key terminologies across categories.
* **To Design and Develop Mental Health Classification Models**  
  - To develop classification models capable of categorizing text into seven mental health statuses: Normal, Depression, Suicidal, Anxiety, Stress, Bipolar, and Personality Disorder.  
  - To implement machine learning algorithms such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and CNN-LSTM ensembles to capture spatial and sequential features.  
  - To utilize advanced transformer-based models such as Fine-Tuned BERT with Dropout Optimization (FL-BERT + DO) to enhance classification performance and generalization.
* **To Evaluate Model Performance and Extract Analytical Insights**  
  - To assess model performance using evaluation metrics including accuracy, precision, recall, and F1-score to determine the most effective algorithm.  
  - To extract meaningful insights into language patterns and emotional expressions characteristic of different mental health conditions.
* **To Develop Interactive Visualization Dashboards for Clinical Use**  
  - To implement a Client Progress Dashboard for tracking mood patterns, coping strategies, and therapeutic goals over time.  
  - To create Clustering-Based Visualizations of client daily logs representing the flow from Triggers → Thoughts → Responses, aiding behavioural analysis.  
  - To integrate an AI-Powered Session Analysis Module for interpreting therapy sessions and deriving actionable insights.  
  - To design comprehensive analytics dashboards with interactive charts that assist therapists in evaluating patient progress.  
  - To automate clinical documentation through a Session Note Generator that provides summarized reports and transcripts.
* **To Explore Future Scope, Ethical Considerations, and Real-World Applications**  
  - To explore future applications such as AI-powered mental health chatbots, early detection systems, and support tools for enhanced healthcare delivery.  
  - To ensure strict adherence to ethical practices, including anonymization, data privacy, and secure handling of sensitive information.  
  - To identify and mitigate biases in dataset labelling and model predictions to promote fairness, inclusivity, and responsible AI deployment.
  1. **Problem Statement:**

To develop and analyse an **AI-driven mental health assessment system** that leverages Cognitive Behavioural Therapy (CBT) **audio recordings and text inputs**, integrating Natural Language Processing (NLP) and **Sentiment Analysis** for emotional state detection. The system will provide **real-time insights through an interactive dashboard**, enabling users to track their mental well-being and progress. Additionally, robust privacy measures, federated learning, will be implemented to ensure secure and confidential data handling.

1. **Literature Review**

Recent advancements in Natural Language Processing (NLP) have significantly improved the accuracy and efficiency of sentiment analysis systems.

Bello et al. [1] proposed a BERT-based sentiment analysis framework integrated with deep learning classifiers like CNN, RNN, and BiLSTM, achieving state-of-the-art performance on tweet datasets. Their study demonstrated that the BERT transformer model outperforms traditional methods such as Word2Vec, CNN, and RNN due to its superior contextual word representation capabilities.

Similarly, Atandoh et al. [2] introduced an integrated deep learning paradigm called B-MLCNN, which achieved remarkable accuracy across multiple datasets including IMDB and Amazon reviews. The model outperformed baseline architectures such as BERT and CNN, and statistical validation using the Friedman test confirmed its superior performance.

Zhang et al. [3] advanced this field further by combining BERT embedding with a sliced multi-head self-attention Bi-GRU model, enhancing both classification accuracy and training efficiency compared to conventional RNN-based approaches.

In another study, Aslan et al. [4] developed a TCNN–Bi-LSTM model using FastText embeddings to analyze sentiments of tweets related to COVID-19 vaccines. Their model significantly outperformed baseline deep learning and traditional machine learning techniques, highlighting the effectiveness of FastText over other embedding techniques like GloVe and TF-IDF in this context.

In the legal domain, Abimbola et al. [5] designed a CNN-LSTM hybrid model for sentiment classification in Canadian maritime case law, achieving an impressive accuracy of 98.05%. The CNN component extracted prominent features from the text, while the LSTM effectively captured temporal dependencies and contextual meaning.

A broader survey conducted by Rahman et al. [6] presented a state-of-the-art review on sentiment analysis, discussing various models, applications, and ongoing challenges, while emphasizing the importance of NLP in decision-making and customer behaviour analysis.

Furthermore, Ahsan et al. [7] addressed privacy concerns in mental health sentiment analysis by proposing a FL-BERT+DO model, which integrates federated learning with data obfuscation techniques. Their framework not only provided high predictive performance but also ensured enhanced data privacy, outperforming baseline FL-DP methods in both accuracy and privacy protection.

**3.Proposed Methodology**

**3.1 Dataset Description**

**Data Overview**

The dataset is categorized into six mental health statuses:

|  |  |
| --- | --- |
| Emotions | Count |
| happy | 7028 |
| sadness | 6265 |
| anger | 2992 |
| fear | 2651 |
| love | 1641 |
| surprise | 879 |

Table 1: Emotions vs count

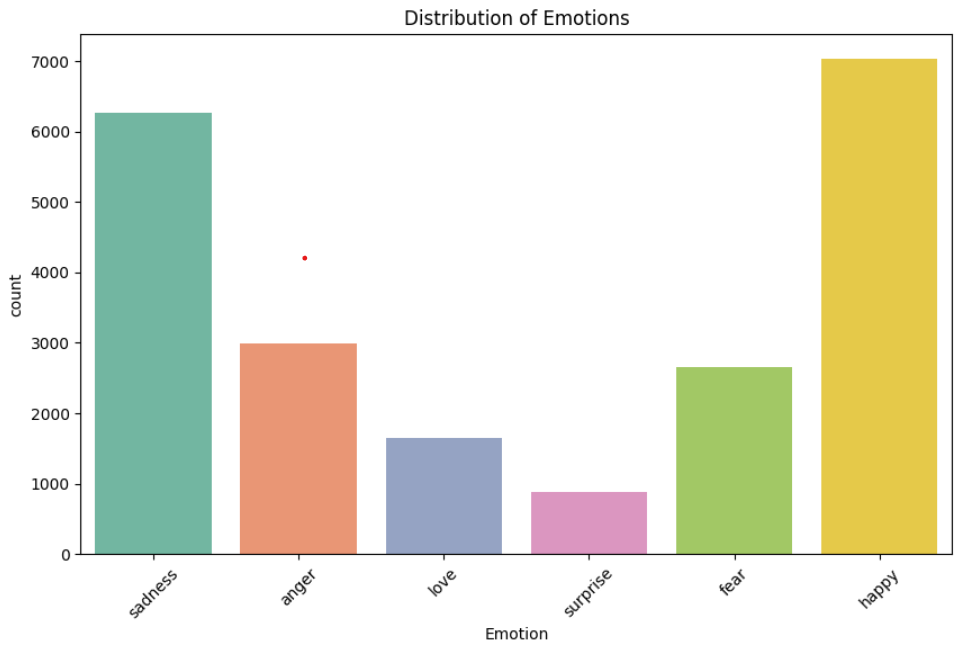


Fig1. Distribution of Emotions of Given Dataset

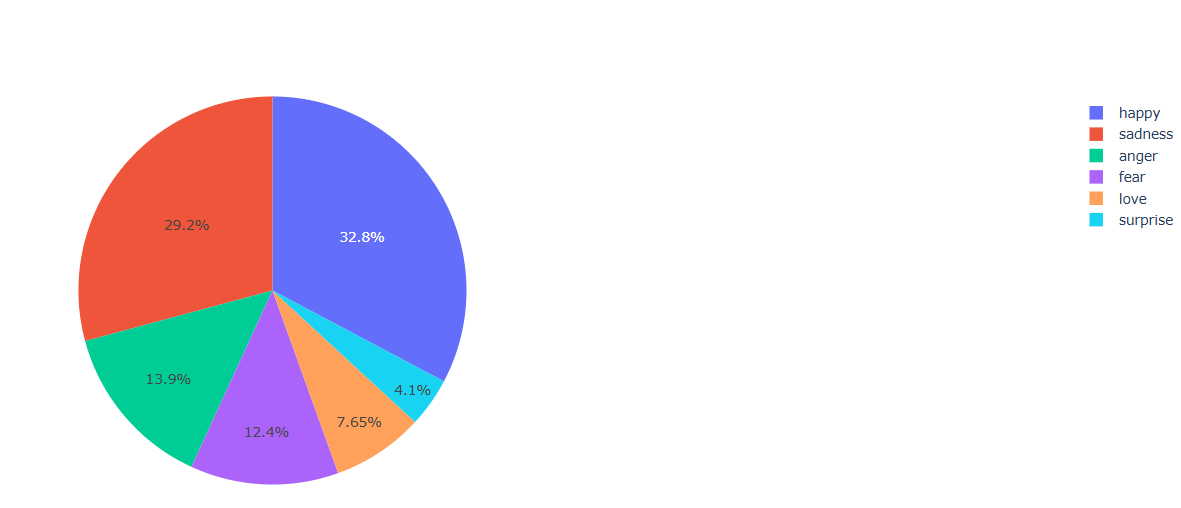


Fig2. Pie-Chart representing Percentage of statements of Emotions across Dataset

Each record in the dataset comprises:

* unique\_id: A unique identifier for each entry.
* statement: The textual data or post.
* Emotion: The assigned category based on sentiment and mental health analysis.
  1. **Data Preprocessing**

To prepare the data for sentiment analysis, the text was subjected to cleaning, tokenization, stopword removal, lemmatization, and stemming using NLTK. This ensured that the processed statements retained essential linguistic features while reducing noise.

**3.2.1 Text Cleaning**

* Lowercasing: Converts all text to lowercase.
* Removing Special Characters & Punctuation: Eliminates non-alphabetic characters.
* Tokenization: Splits text into individual words.
* Stopword Removal: Removes common words that do not contribute to sentiment (e.g., "the," "is," "and").

**3.2.2 Text Normalization**

* Lemmatization: Converts words to their base form (e.g., "running" → "run").
* Stemming: Further reduces words to their root form (e.g., "running" → "run")

**3.2.3 Word Frequency Analysis**

A word frequency analysis was conducted on the processed text for each mental health category. Using word clouds and word importance ranking, the most commonly occurring words in each category were identified. The key findings include:

* + Sadness: feel, im, know, time, im feel, want, think.
  + Anger: feel, im, time, want, im feel, peopl, think.
  + Love: feel, love, im, support, realli, know, want.
  + Surprise: feel, im, feel amaz, amaz, realli, time, look.
  + Fear: feel, im, im feel, go, know, time, feel littl.
  + Happy: feel, im, im feel, time, go, love, know.

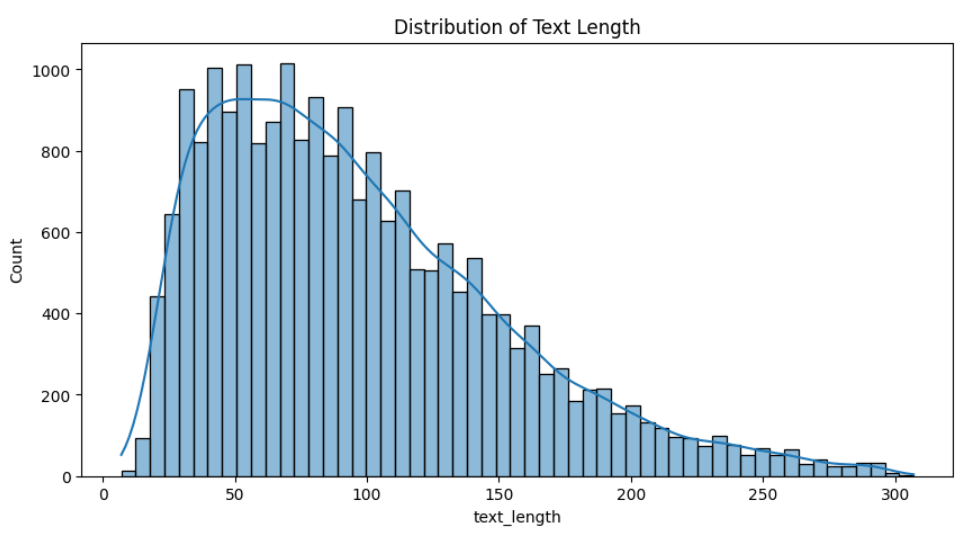


Fig3. Distribution of Text Length vs Count

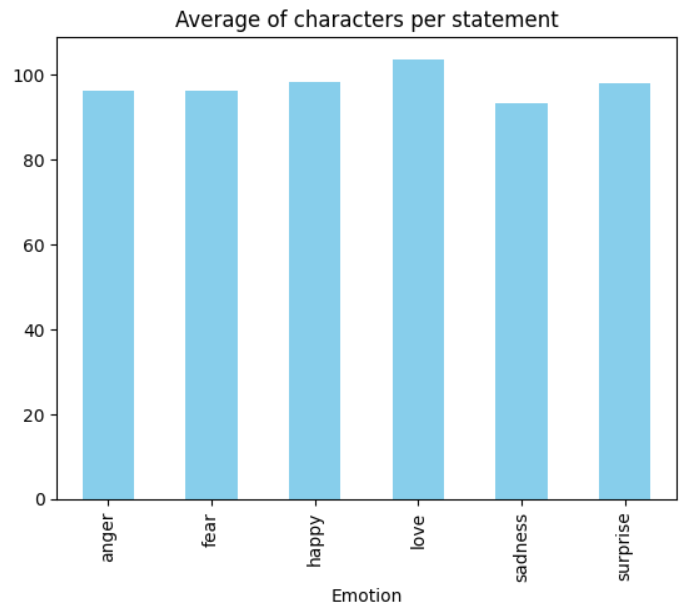


Fig4. Distribution of Characters per statement

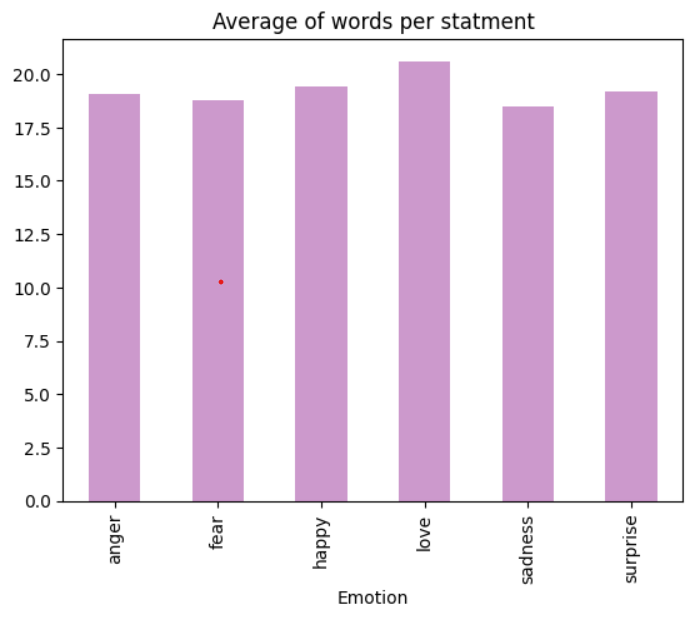


Fig5. Average of words per statement



Fig.6 Word cloud of various Emotions

Figure 6 illustrates the word cloud of various Emotions in the dataset

**3.3 Algorithmic Methodology**

**3.3.1 Why Do We Require Algorithms for Sentiment Analysis?**

|  |  |  |
| --- | --- | --- |
| **Technique** | **Strengths** | **Limitations** |
| **Lexicon-Based Approach** | Simple and does not need training data. | Limited to predefined words, lacks context. |
| **Corpus-Based Approach** | Expands vocabulary dynamically. | Requires large datasets for good accuracy. |
| **Dictionary-Based Approach** | Uses existing dictionaries for analysis. | Cannot handle slang or new words. |
| **Machine Learning Approach** | Learns patterns from data; adapts to different domains. | Requires labelled data; computationally expensive. |
| **Naïve Bayes Classifier** | Fast, scalable, and works well for text classification. | Assumes feature independence, which is often incorrect. |
| **Support Vector Machine** | Handles high-dimensional data effectively. | Computationally expensive, slow for large data. |
| **Neural Networks** | Captures complex patterns; highly accurate. | Requires high computing power; prone to overfitting. |
| **Decision Tree Classifier** | Easy to interpret and implement. | Less precise; requires feature engineering. |
| **Hybrid Approaches** | Combines multiple methods for better accuracy. | Computationally expensive and difficult to tune. |

Table1: Different Techniques of Sentiment Analysis with strength and Limitations

Sentiment analysis involves the process of extracting opinions, emotions, and attitudes from textual data. However, human language is highly complex, with varying grammatical structures, contextual meanings, and emotional undertones. Manually analysing large volumes of text is impractical and inaccurate. Therefore, machine learning and deep learning algorithms are employed to automate this process effectively.

These algorithms are essential because:

* They **learn hidden patterns** from data automatically.
* They **capture contextual sentiment** in varying linguistic forms.
* They **scale easily** to massive datasets.
* They **improve classification performance** over time with more data.

Using deep learning models like CNN, LSTM, ensemble models, and Fine-Tuned BERT with Dropout Optimization (FL-BERT + DO) helps capture both local syntactic patterns and long-range dependencies in text while ensuring contextual awareness.

**3.3.2. Algorithms and Their Mechanisms / Architecture with Sentiment Analysis**

**Convolutional Neural Network (CNN)**

1.Sequence Analysis and Padding:

* Length Distribution: Histograms of tokenized sentence lengths are plotted for both train and test sets to determine an appropriate maximum sequence length.
* Padding: All sequences are padded to a fixed maximum length (30 tokens) using pad sequences, ensuring uniform input dimensions for the CNN.

2. Label Encoding:

* Encoding Labels: Emotion labels are encoded into integers using Label Encoder.
* One-Hot Encoding: These integer labels are then one-hot encoded using to\_categorical to prepare them for training with a SoftMax output layer.
* Class Mapping: A mapping of original emotion labels to encoded integers is printed for reference.

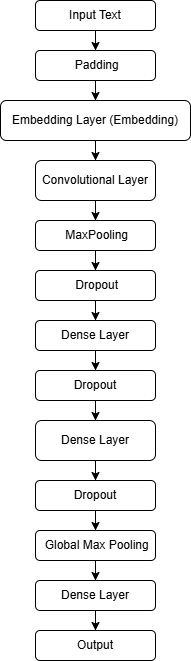


Fig7. Architectural Diagram of CNN

3. Model Architecture (CNN):

* Embedding Layer: Converts input word indices to 300-dimensional dense vectors.  
  Convolutional Layer: A Conv1D layer with 64 filters and a kernel size of 8 is applied to capture local n-gram patterns in text.
* Pooling and Dropout: A MaxPooling1D layer reduces dimensionality, and a Dropout layer (rate = 0.1) helps prevent overfitting.
* Fully Connected Layers: Two dense layers (with 8 and 4 units respectively) and additional dropout are used to learn complex representations.
* Global Pooling: GlobalMaxPooling1D reduces the feature map into a single vector.  
  Output Layer: A Dense layer with 6 units (one per emotion class) and SoftMax activation is used for multi-class classification.
* Compilation: The model is compiled using the Adam optimizer (with a learning rate of 0.0001) and categorical\_crossentropy as the loss function.

4. Model Training:

* Training Loop: The model is trained for 10 epochs with a batch size of 64.
* Validation: Evaluation is performed on the test set after each epoch to track overfitting and generalization.

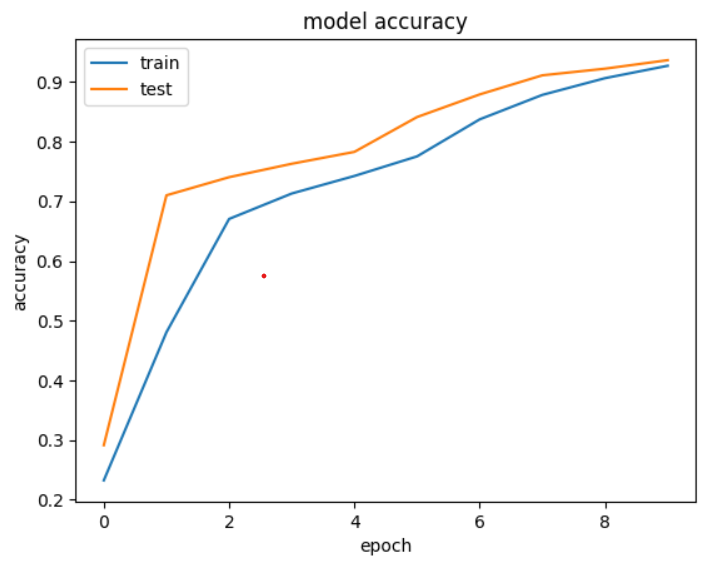


Fig8. CNN Model Accuracy vs Epoch

5. Evaluation and Visualization:

* Classification Metrics: Predictions on the test set are compared with true labels using classification\_report, showing precision, recall, and F1-score for each emotion class.
* Confusion Matrix: A confusion matrix is plotted using Seaborn to visualize classification performance across classes.

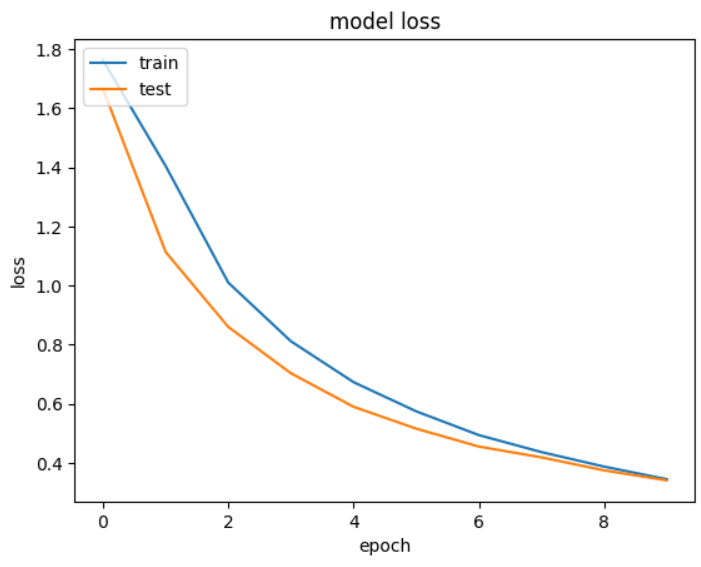


Fig9. CNN Model Loss vs Epoch

* Training History: Accuracy and loss curves for both training and validation sets are plotted to monitor learning progress.

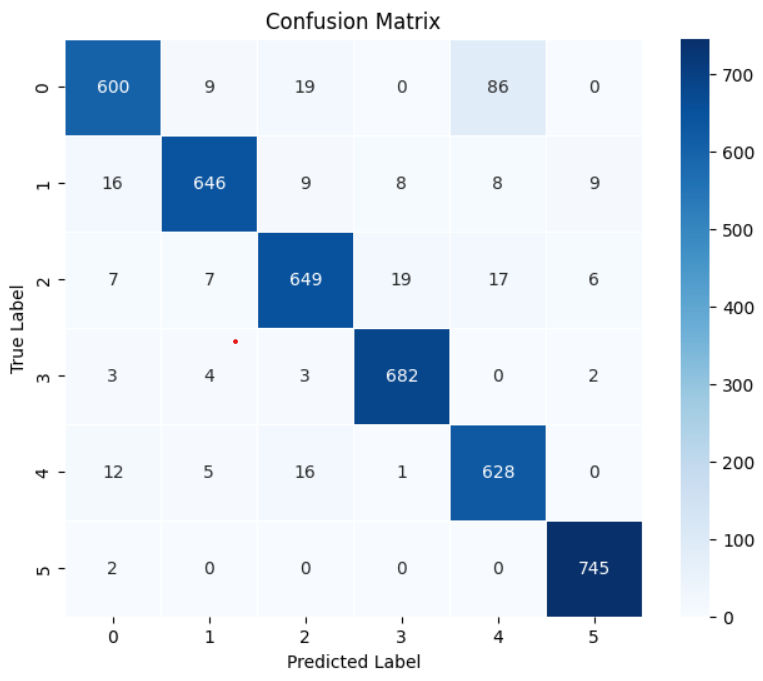


Fig10. CNN Confusion Matrix

8. Inference on New Texts:

* Text Input: Example sentences and paragraphs are tokenized, padded, and passed to the trained model.
* Prediction and Decoding: Predicted class indices are converted back to emotion labels using the saved Label Encoder.
* Confidence Scores: SoftMax probabilities for each emotion class are displayed for interpretation of model certainty.

9. Model Saving and Deployment:

* Saving Artifacts: The trained model (.h5), tokenizer (tokenizer.pkl), and label encoder (label\_encoder.pkl) are saved to disk for reuse.
* Reloading for Inference: These saved files can be reloaded to classify new text without retraining the model.

**Long Short-Term Memory (LSTM)**

The model was built using the Sequential API from TensorFlow Keras, designed for multi-class emotion classification based on text input. The architecture comprises the following components:

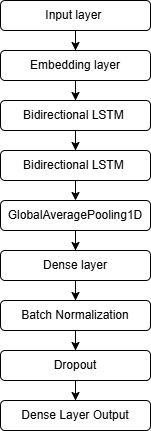


Fig11. Architectural Diagram LSTM

* Input Layer: Receives padded sequences of 50 tokens per text input.
* Embedding Layer: Converts tokens into dense 128-dimensional vectors, enabling the model to learn semantic relationships between words.
* Bidirectional LSTM Layers (×2): Two stacked Bidirectional LSTM layers, each with 256 units and a dropout rate of 0.2, allow the model to capture contextual dependencies from both past and future word sequences.
* GlobalAveragePooling1D: Aggregates the LSTM outputs by averaging across time steps, reducing the dimensionality for the dense layers.
* Dense Layers:
  + One dense layer with 512 units and ReLU activation, followed by Batch Normalization and Dropout (0.1) to prevent overfitting.
  + Another dense layer with 128 units, also followed by Batch Normalization and Dropout (0.1).
* Output Layer: A SoftMax-activated layer with 6 units, corresponding to the six emotion classes, to output class probabilities.

The model was compiled using the Adam optimizer with a learning rate of 0.001 and categoricalcross entropy loss, suitable for handling multiple emotion categories. To enhance training efficiency and generalization, the following callbacks were applied:

* Early Stopping: Stops training if validation loss doesn't improve for 3 consecutive epochs and restores the best model weights.
* ReduceLROnPlateau: Automatically reduces the learning rate by half if the validation loss plateaus for 2 epochs.
* Model Checkpoint: Saves the best-performing model (based on validation loss) to a file named best\_model.h5.

Evaluation and Visualization:

* Classification Metrics: Predictions on the test set are compared with true labels using classification\_report, showing precision, recall, and F1-score for each emotion class.
* Confusion Matrix: A confusion matrix is plotted using Seaborn to visualize classification performance across classes.

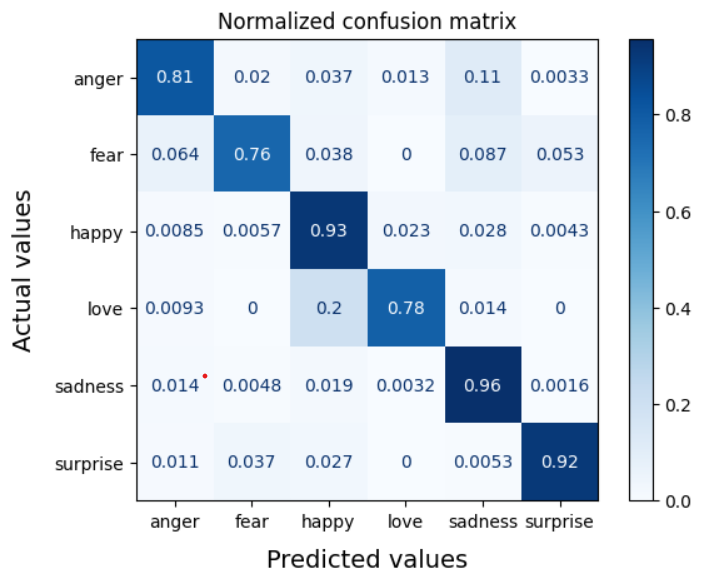


Fig12. Confusion Matrix LSTM

* Training History: Accuracy and loss curves for both training and validation sets are plotted to monitor learning progress.

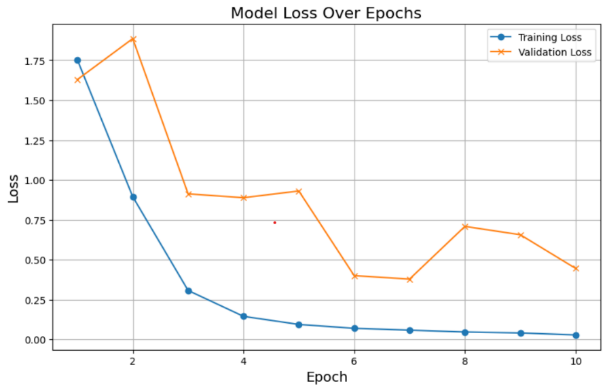


Fig13. LSTM Model Loss vs Epoch

**Ensemble of CNN and LSTM**

The Below figure shows the architecture of CNN and LSTM ; the details of each step is given below :

**Input Layer:** The model begins with the input layer, where each text sample is converted into a sequence of tokens. To ensure uniform input dimensions, all sequences are either padded or truncated to a fixed length, typically 100 tokens. This step allows the model to process batches of data efficiently while preserving sufficient context from the text for accurate sentiment analysis.

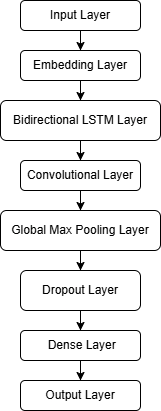


Fig14. Architectural Diagram of CNN + LSTM

**Embedding Layer:** The embedding layer transforms each token in the sequence into a dense vector representation. Pre-trained word embeddings such as GloVe or Word2Vec are commonly used here, which help the model capture the semantic meaning of words. Each token index is mapped to a vector of dimension 100 or 300, resulting in an output of shape *(sequence length, embedding dimension)*. This layer essentially converts sparse token indices into meaningful, dense vectors that carry contextual information about the words.

**Convolutional Layer (1D CNN):** Next, the embedded sequences are passed through a 1D convolutional layer. This layer applies multiple filters (typically 100) with a kernel size of 3 to capture local patterns and key phrases in the text. The convolution operation slides across the sequence, detecting important n-gram features like bigrams and trigrams that are crucial for sentiment cues. For example, phrases such as "not good" or "very bad" are effectively captured at this stage.

**Max Pooling Layer:** Following the convolutional layer, a max pooling layer is applied to reduce the dimensionality of the feature maps. By selecting the maximum value from each pooling window, this layer retains only the most significant features while discarding less relevant information. This not only reduces the computational complexity but also helps the model become more robust by focusing on the most important local features extracted by the CNN.

**LSTM Layer:** The pooled features are then fed into an LSTM (Long Short-Term Memory) layer. With 128 memory units, this layer processes the sequential data to capture long-term dependencies and contextual relationships within the text. The LSTM’s gating mechanisms enable it to remember important information over longer distances, which is especially beneficial in sentiment analysis where words like "not" can change the meaning of words that appear later in the sentence.

**Dense Layer:** After the LSTM layer, the output is passed through a fully connected dense layer with 64 neurons. This layer applies a ReLU activation function to introduce non-linearity and further transforms the learned features into a higher-level representation. This helps the model to better differentiate between subtle variations in sentiment expressions, thereby enhancing its classification ability.

**Output Layer:** Finally, the model ends with an output layer that generates the sentiment prediction. For binary classification tasks, a single neuron with a sigmoid activation function is used to output a probability between 0 and 1, indicating positive or negative sentiment. In the case of multi-class sentiment classification, the output layer uses a SoftMax activation function with multiple neurons, each representing a sentiment class such as positive, neutral, or negative.

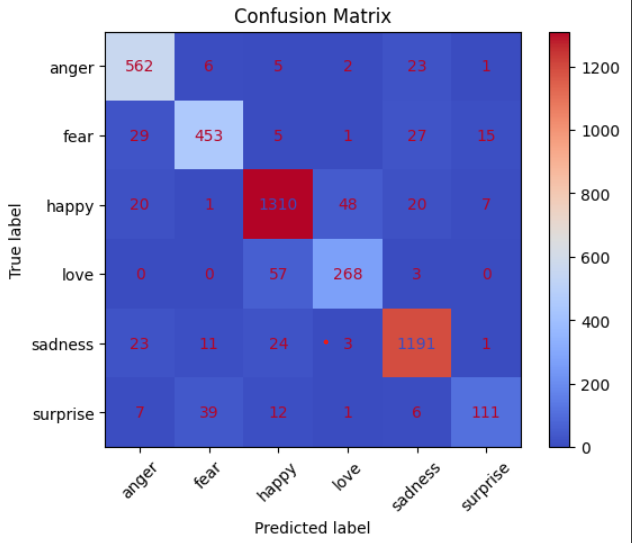


Fig15. Confusion Matrix – Ensemble CNN + LSTM

Evaluation and Visualization:

* Classification Metrics: Predictions on the test set are compared with true labels using classification\_report, showing precision, recall, and F1-score for each emotion class.
* Confusion Matrix: A confusion matrix is plotted using Seaborn to visualize classification performance across classes.
* Training History: Accuracy and loss curves for both training and validation sets are plotted to monitor learning progress.

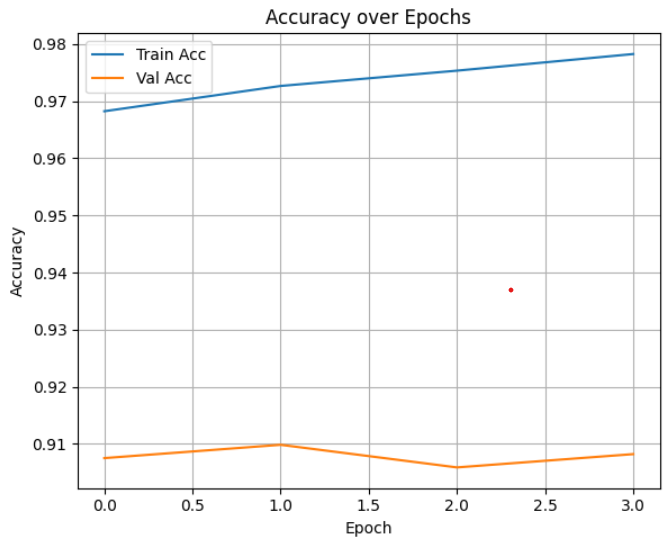


Fig16. Ensemble CNN + LSTM Accuracy vs Epoch

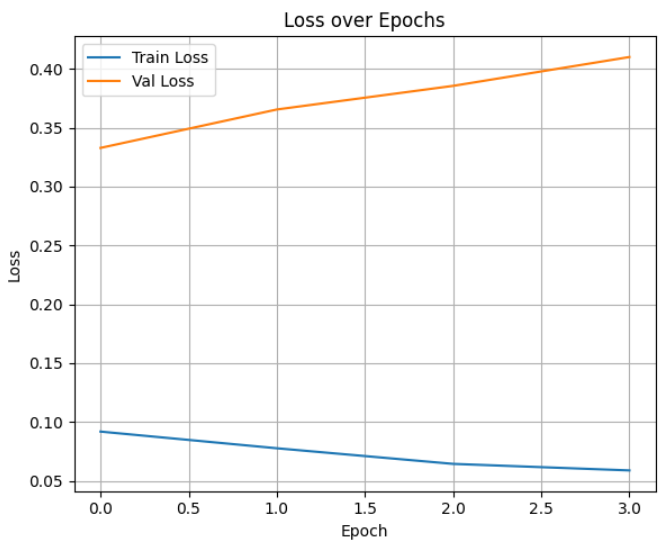


Fig17. Ensemble CNN + LSTM Loss vs Epoch

The above figure shows Loss over Epochs, Accuracy over Epochs and confusion matrix, the accuracy achieved for our dataset is 90.75%. The graph shows the loss curves over epochs for both training and validation sets. The training loss (blue line) is steadily decreasing, indicating that the model is learning and fitting well on the training data. However, the validation loss (orange line) is increasing over epochs, suggesting that the model is starting to overfit and its performance on unseen data is degrading. This divergence between training and validation loss highlights the need for regularization or early stopping to prevent overfitting.

**BERT (Bidirectional Encoder Representations from Transformers )**

1. Data Preprocessing:  
• Loading Data: The dataset (Emotion\_final.csv) is loaded and cleaned by selecting only the relevant columns (Text and Emotion).  
• Label Encoding: Emotion labels are encoded into numeric values using Label Encoder, preparing them for model training.  
• Text Tokenization: The BERT tokenizer (Bert-base-uncased) is used to tokenize the text with truncation and padding to fit BERT’s input requirements.

2. Dataset Preparation:  
• Data Splitting: The dataset is split into training and validation sets (90/10 split) using train\_test\_split.  
• Dataset Class Creation: A custom PyTorch dataset class is defined to format the tokenized text and corresponding labels into tensors.

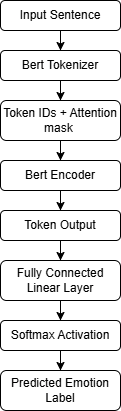


Fig18. Architectural diagram of BERT

3. Model Initialization:  
• Pre-trained Model: A pre-trained BERT model (BertForSequenceClassification) is loaded with the number of output labels equal to the number of emotion classes.  
• Device Allocation: The model is moved to GPU if available to speed up training and inference.

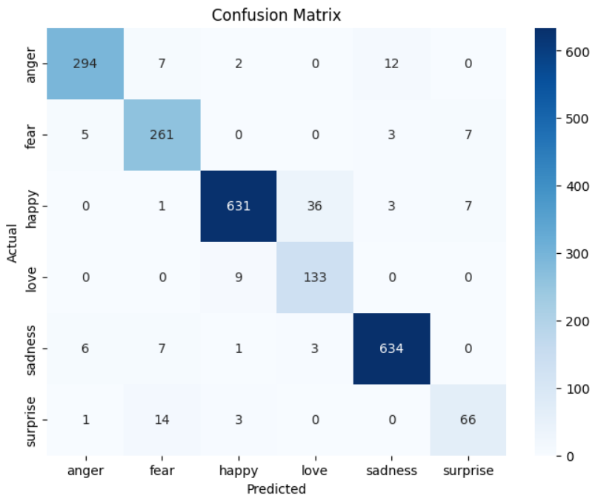


Fig19. Confusion Matrix – BERT

4. Training Setup:  
• Training Arguments: Training parameters like batch size, epochs, logging, and evaluation strategy are defined using Training Arguments.  
• Model Training: The model is trained using Hugging Face’s Trainer class on the prepared dataset.

5. Evaluation and Prediction:  
• Evaluation: The trained model is evaluated on the validation set using metrics such as precision, recall, and F1-score.  
• Prediction: The model predicts emotions on new, unseen texts. Predictions are decoded back to emotion labels using the inverse of the label encoder.  
• Visualization: Classification report and confusion matrix are used to analyze model performance visually.

**Federated Fine-Tuned BERT with Dropout Optimization (FL-BERT + DO)**

Privacy Concerns and Solutions in Federated Learning:

Privacy Concerns:

* Data Exposure: In traditional machine learning, centralized data storage exposes personal or sensitive data to risks such as data breaches.
* Data Sharing: Sharing raw data between clients and servers increases the chance of leakage of sensitive information, such as user interactions, preferences, or behaviours.

How Federated Learning Solves Privacy Issues:

1. Data Remains Local: In federated learning (FL), data stays on the client device (e.g., smartphones, edge devices). Only model updates (gradients/weights) are sent to the server, significantly reducing the exposure of sensitive data.
2. Model Updates Instead of Raw Data: Instead of sending raw data, FL sends only the model updates after local training. This ensures that the server never has access to raw user data, maintaining privacy.
3. Federated Averaging (FedAvg): The global model is updated by averaging the local updates from each client. This ensures that individual client data is not directly accessible, and only the knowledge gained from the model updates is shared.
4. Differential Privacy: To further safeguard user privacy, differential privacy techniques can be applied to the model updates. This ensures that the model updates do not inadvertently reveal sensitive information about any individual client.
5. Homomorphic Encryption: For additional security, federated learning can leverage homomorphic encryption, which allows computations to be performed on encrypted data. This means that even if a malicious actor intercepts the updates, they cannot access the raw data or model parameters.
6. Secure Aggregation: Secure aggregation protocols ensure that the server can aggregate model updates from clients without gaining access to the individual updates. This prevents the server from inferring any sensitive information about the individual clients' data.

Methodology:

1. Setup and Dependencies:

The necessary libraries are installed for federated learning (Flower), NLP (Hugging Face Transformers), deep learning (PyTorch), data handling (pandas), and machine learning tools (scikit-learn).

2. Data Preprocessing:

* Loading Data: The dataset (Emotion\_final.csv) is loaded and examined to check the data structure, missing values, and class distribution.
* Label Encoding: The categorical emotion labels are encoded into numeric labels using Label Encoder, allowing the model to process the data effectively.
* Text Tokenization: The text data is tokenized using the BERT tokenizer, with truncation and padding applied to ensure all text inputs are of the correct size for BERT.

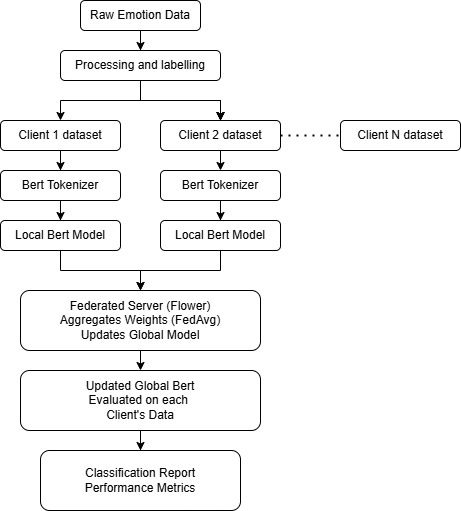
****

Fig20. Architectural diagram of FL-BERT with Dropout Optimization

3. Dataset Preparation

* Custom Dataset: A custom PyTorch Dataset class is created to handle the tokenized data and labels, converting them into tensors compatible with BERT.
* Client Data Splitting: The dataset is shuffled and divided into multiple parts (simulating clients in federated learning). Each client will train on their own data split.

4. Federated Learning Setup

* Model Initialization: A fresh BERT model is initialized for sequence classification, with the number of output labels set according to the unique emotions.
* Local Training: Each client trains a local model using their partitioned dataset. The model is updated with local data through several federated rounds.
* Model Aggregation (FedAvg): After each round, the weights from each client are aggregated to form a global model using Federated Averaging (FedAvg).

5. Model Evaluation

* Local Evaluation: After training, the performance of the model is evaluated on each client's validation dataset, and metrics are recorded.
* Global Evaluation: The final global model is evaluated on the validation data of one client (typically client 0) to assess overall performance.

6. Predictions on New Data

* Inference: The trained global model is used to predict emotions for new text inputs. The output includes predicted emotions with confidence scores and class probabilities.

7. Model Saving and Loading

* The final global model and tokenizer are saved for future use. The model can later be loaded, and inference can be performed on new data without retraining.

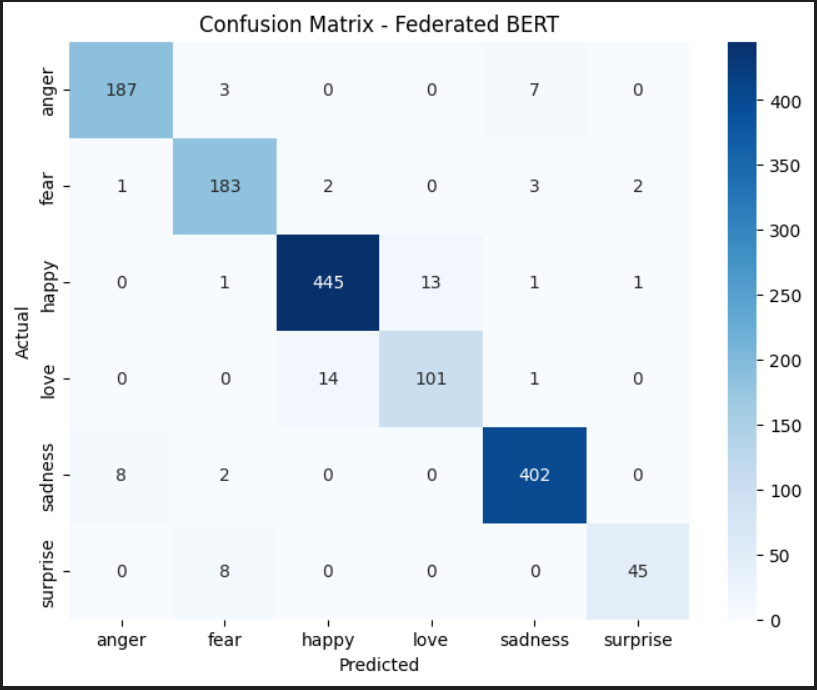


Fig21. Confusion Matrix – FL-BERT with Dropout Optimisation

8. Confusion Matrix and Classification Report

* Confusion Matrix: A confusion matrix is plotted to visualize model performance, showing how well the model distinguishes between different emotions.
* Classification Report: A detailed classification report is generated for each client and the final global model, showing precision, recall, F1-score, and accuracy for each emotion class.

**3.4 Treatment Progress Tracking & Dashboard Integration**

The Dashboard is the central hub of the Therapy Management Platform, designed to provide therapists and clients with real-time insights, collaborative tools, and secure access to critical session data. Built with responsiveness and usability in mind, it ensures a seamless experience across all devices.

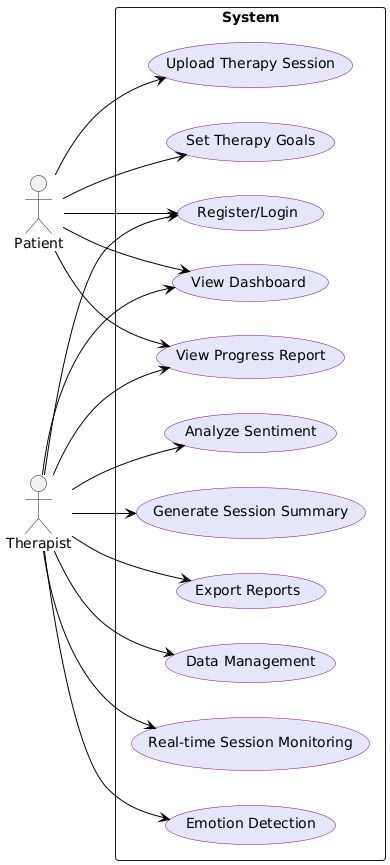


Fig22. Use Case Diagram (Dashboard)

**Key Features:**

Real-Time Updates:

* Leveraging Supabase’s subscription service, the dashboard delivers instant updates for:
  + New messages
  + Active sessions
  + Session logs
* This ensures both clients and therapists stay synchronized without needing to refresh the page.

Data Visualization:

* Bar charts and line charts are used to illustrate trends in:
  + Emotional patterns over time
  + Session frequency
  + Client engagement levels
* These visuals make it easier for therapists to make informed decisions and track client progress effectively.

Session Notes and Summaries:

* Session transcripts are stored as JSON data, which is:
  + Parsed and cleaned
  + Displayed in a readable, structured format
* Notes can be edited and shared between therapist and client in real-time.

Emotion Analysis:

* Each message and audio recording is analysed using the CNN/BERT-based emotion detection model.
* Emotions like happiness, sadness, anger, fear, love, and surprise are detected and displayed on the dashboard, giving therapists deeper insight into client states.

Role-Based Functionality:

* The dashboard interface and features adapt based on the user’s role (therapist or client).
* Security controls ensure each user only sees information relevant to their responsibilities.

Supabase Integration:

* Supabase handles:
  + Authentication (login, signup)
  + Database operations (session storage, updates)
  + Real-time communication (message sync, alerts)

Responsive Design:

* The dashboard is fully responsive, optimized for mobile, tablet, and desktop devices.
* A modern UI framework ensures smooth transitions and an intuitive layout.

User-Friendly Interface:

* Clean typography, clear iconography, and intuitive navigation make the dashboard accessible to users with varying levels of tech proficiency.

Client-Therapist Collaboration:

* Shared visibility into notes, analytics, and conversations improves communication and trust.
* Enables therapists to tailor sessions based on emotional trends and past feedback.

Security and Privacy:

* Role-based access control (RBAC) is enforced throughout.
* All sensitive data is encrypted and securely transmitted between users and servers.

The dashboard empowers therapists and clients by providing real-time data, emotional insights, and collaborative tools. It acts as a powerful, user-centric interface that enhances therapy sessions with modern technology, intuitive design, and strong security practices.

**5. Applications**

1. **Applications in the Medical Field:**
2. Mental Health Early Detection Systems: Used in healthcare facilities to detect early signs of psychological disorders through patients’ communication patterns.
3. AI-Based Clinical Decision Support Tools: Assists therapists and psychiatrists in diagnosing and monitoring mental health conditions by analyzing patient records and conversations.
4. Therapeutic Progress Monitoring Dashboards: Empowers clinicians with real-time visual dashboards that track patient mood patterns, triggers, and therapeutic milestones.
5. Mental Health Chatbots and Virtual Counsellors: AI-powered chatbots integrated with sentiment analysis can provide 24/7 support and pre-screening for mental health assistance.
6. **Applications in the Non-Medical Field:**
7. Workplace Mental Wellness Monitoring Systems: Can be deployed in organizations to monitor employee stress and emotional well-being through anonymous textual feedback analysis.
8. Sentiment Analysis in Customer Support: Helps businesses gauge customer mood and mental state from service chats or reviews, improving empathy in customer service.
9. Content Moderation on Social Media Platforms: Detects suicidal or depressive content to initiate timely interventions or flag harmful language trends.
10. Educational Mental Health Tools: Used in schools and universities to identify students struggling with anxiety, stress, or other psychological challenges via written assignments or feedback.

**6. Result**

|  |  |
| --- | --- |
| **Metric** | **Definition** |
| Accuracy | Measures correct predictions overall. |
| Precision | Measures the proportion of correctly identified positive sentiments. |
| Recall | Measures the model’s ability to identify positive cases. |
| F1-Score | Harmonic mean of precision and recall for balanced evaluation. |

Table3: Evaluation Metrics used over Models

For CNN the model achieved an overall accuracy of 94%, demonstrating strong performance across all six emotion classes. Precision, recall, and F1-scores for each class ranged between 0.88 to 0.99, indicating reliable and balanced classification. The highest F1-score was observed in class 5-surprise (0.97), while all other classes maintained competitive metrics. These results confirm the effectiveness of the model in emotion recognition from text data.

Evaluation Metrics

For LSTM the model achieved an overall accuracy of **89%** on the test set, indicating robust performance in classifying emotional text. Precision, recall, and F1-scores across all six emotion classes ranged from **0.76 to 0.96**, with particularly strong performance in class 4-Sadness (F1-score: 0.92) and class 5-surprise (F1-score: 0.91). The balanced scores reflect the model’s consistency in detecting various emotional tones effectively.

For LSTM+CNN ensemble the model attained an overall **accuracy of 90.75%**, demonstrating effective emotion classification across six categories. The highest F1-scores were observed for *sadness* (0.94) and *happy* (0.93), indicating strong detection of these emotions. While *surprise* had a relatively lower F1-score of 0.71, the macro and weighted averages remained high, showcasing balanced performance across classes.

For BERT the model achieved an **accuracy of 94%**, with strong performance across most emotion categories. The highest F1-scores were for *sadness* (0.97) and *happy* (0.95), indicating accurate classification in these areas. While *love* and *surprise* had slightly lower F1-scores, the overall results remain consistent, with a balanced performance reflected in both the macro and weighted averages.

For FL- BERT- DO the model achieved a high **accuracy of 95%** with consistently strong performance across different emotion categories. The F1-scores for *happy* (0.97) and *sadness* (0.97) were particularly high, demonstrating strong classification abilities for these emotions. While *love* and *surprise* had slightly lower F1-scores, the overall results reflect a solid and balanced model performance, as indicated by both the macro and weighted averages.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  |  | | --- | --- | | Model/ Metric | Accuracy | | CNN | 94% | | LSTM | 89% | | Ensemble CNN + LSTM | 90.75% | | BERT | 94% | | FL-BERT + DO | 95% | |
|  |  |

Table4: Accuracy Obtained over Models

The performance of various models was evaluated based on accuracy, with FL-BERT + DO achieving the highest accuracy at 95%, demonstrating the effectiveness of Federated Learning combined with Dropout optimization. BERT and CNN both performed well, each achieving an accuracy of 94%, indicating strong capabilities in emotion classification. The Ensemble CNN + LSTM model achieved 90.75%, which is a reasonable improvement over the LSTM’s individual performance of 89%. These results suggest that combining models like CNN and LSTM can provide a performance boost, but the use of advanced models like BERT and FL-BERT shows significant promise in achieving higher accuracy levels.

1. **Conclusion**

The proposed project has successfully demonstrated the effectiveness of deep learning and transformer-based models in detecting and classifying various mental health conditions from textual data. Through comprehensive preprocessing and model development, significant improvements in classification accuracy were observed using models such as **CNN, LSTM, and the fine-tuned BERT with Dropout Optimization (FL-BERT + DO**). The evaluation metrics, including accuracy, precision, recall, and F1-score, indicate that the ensemble and transformer-based approaches outperform traditional models in recognizing subtle linguistic patterns associated with mental health statuses like **Depression, Anxiety, Suicidal tendencies, Stress, Bipolar Disorder, and Personality Disorders**.

Moreover, the development of interactive dashboards and session analysis modules has provided an enhanced framework for therapists to track client progress, analyze behavioural trends, and support clinical decisions with real-time insights. The project not only proves the feasibility of **AI-driven mental health monitoring** but also emphasizes its practical value in clinical support, early detection, and efficient documentation. The integration of ethical considerations, such as data privacy and bias mitigation, ensures the model's readiness for real-world deployment in mental health care systems.

1. **Future Scope**

Building on the success of this project, several future directions can further enhance the impact and applicability of AI in mental health monitoring:

1. Multimodal Data Integration: Incorporating additional data sources such as facial expressions, and physiological signals (e.g., heart rate, sleep patterns) can provide a more holistic understanding of a user’s mental state.
2. Real-time Monitoring and Intervention: Developing real-time systems that not only detect mental health risks but also trigger alerts or interventions (e.g., notifying caregivers or suggesting coping strategies) could significantly enhance patient outcomes.
3. Personalized Mental Health Models: Future models can focus on personalization by adapting to individual linguistic styles and behavioural patterns over time, thereby improving accuracy and relevance in mental health assessments.
4. Longitudinal Analysis and Progress Tracking: Expanding the system to support long-term analysis will help therapists monitor changes in a patient's mental health over extended periods, facilitating early relapse detection and adaptive treatment planning.
5. Cross-cultural and Multilingual Capabilities: Enhancing model performance across diverse cultural and language contexts can broaden the applicability of the system globally, especially in underrepresented regions.
6. Integration with Healthcare Systems: Seamless integration into Electronic Health Record (EHR) systems can streamline documentation, promote collaborative care, and support evidence-based decision-making.

**9.References**

[1] A. Bello, S.-C. Ng, and M.-F. Leung, “A BERT Framework to Sentiment Analysis of Tweets,” *Sensors*, vol. 23, no. 1, 2023. [Online].

Available: <https://www.mdpi.com/1424-8220/23/1/506>

[2] P. Atandoh, F. Zhang, D. Adu-Gyamfi, P. H. Atandoh, and R. E. Nuhoho, “Integrated deep learning paradigm for document-based sentiment analysis,” *Journal of King Saud University - Computer and Information Sciences*, 2023. [Online].

Available: <https://www.sciencedirect.com/science/article/pii/S1319157823001325>

[3] X. Zhang, Z. Wu, K. Liu, Z. Zhao, J. Wang, and C. Wu, “Text Sentiment Classification Based on BERT Embedding and Sliced Multi-Head Self-Attention Bi-GRU,” *Sensors*, 2023. [Online].

Available: <https://pmc.ncbi.nlm.nih.gov/articles/PMC9920561/>

[4] S. Aslan and M. Turgut, “A novel TCNN–Bi‐LSTM deep learning model for predicting sentiments of tweets about COVID‐19 vaccines,” *2022*. [Online].

Available: <https://pmc.ncbi.nlm.nih.gov/articles/PMC9874433/>

[5] B. Abimbola, E. D. L. C. Marin, and Q. Tan, “Enhancing Legal Sentiment Analysis: A Convolutional Neural Network–Long Short-Term Memory Document-Level Model,” *Machine Learning and Knowledge Extraction*, vol. 6, no. 2, 2024. [Online].

Available: <https://www.mdpi.com/2504-4990/6/2/41>

[6] J. Rahman et al., “Recent advancements and challenges of NLP-based sentiment analysis: A state-of-the-art review,” *Natural Language Processing Journal*, 2024. [Online].

Available: <https://www.sciencedirect.com/science/article/pii/S2949719124000074>

[7] S. I. Ahsan, D. Djenouri, and R. Haider, “Privacy-Enhanced Sentiment Analysis in Mental Health: Federated Learning with Data Obfuscation and Bidirectional Encoder Representations from Transformers,” *Electronics*, vol. 13, no. 23, 2024. [Online].

Available: <https://www.mdpi.com/2079-9292/13/23/4650>

[8] Dataset Link –

<https://www.kaggle.com/datasets/ishantjuyal/emotions-in-text>