International Journal of Advance Research in Science and Engineering
Volume. No. 15, Issue No. 04, April, 2026
ISSN: 2319-8354

Emotion Detection using Text: Integrating Diverse Explainable AI Methods

Ms. D. Ramya
Assistant Professor
Tirumala Engineering college
Andhrapradesh, India
Ramyadannavarapu@gmail.com

Yaddanapalli Joseph Mahesh
Department of IT
Tirumala Engineering College
Andhrapradesh, India
Maheshyaddanapalli17@gmail.com

Namineni Rakesh
Department of IT
Tirumala Engineering College
Andhrapradesh, India
rakeshnamineni659@gmail.com

Rettadi Omkar
Department of IT
Tirumala Engineering College
Andhrapradesh, India
omkarrettadi@gmail.com

Tanniru Rohith Sai
Department of IT
Tirumala Engineering College
Andhrapradesh, India
tannirurohithsai@gmail.com

***Abstract*—This paper proposes a strong and interpretable method for emotion recognition from text, using the ISEAR dataset and the Emotion sentiment Dataset. We preprocess the data by cleaning the text and removing stop words, and then extract features using TF-IDF, and Bag of Words vectorization methods. The feature matrix obtained is used to train several classifiers, such as Support Vector Machines (SVM), Decision Trees, Random Forests, Logistic Regression, and XGBoost. Based on many evaluations, XGBoost is the most accurate model. For in- creasing transparency and interpretability, we incorporate Local Interpretable Model-Agnostic Explanations (LIME) to explain model predictions. Furthermore, we used SHAP and Anchor methods to get both global and instance-based explanations of feature importance. These tools enable us to identify and visualize the dominant features driving the model's decisions, providing important information into its reasoning process. In addition, we perform further experiments with a Neural Tangent Kernel (NTK) and Support Vector Machines to measure the robustness of our model on various subsets of data. Our study has found the effectiveness of combining comprehensive feature extraction with explainable AI techniques, focusing on the vital position of transparency of emotion detection apps.**

***Index Terms*—Emotion Detection, Text Classification, Explainable AI, LIME, SHAP, Anchor, XGBoost, Neural Tangent Kernel, ISEAR, Emotion Sentiment Dataset.**

I. Introduction

Emotion detection from text is a successful field of study. It has numerous applications in natural language processing such as lie detection, mental health assessment for disease diagnosis, marketing, etc. Multi-label emotion classification improves human-computer interaction through comprehending the user's emotions. As already stated, the effectiveness of different autonomous or automated systems can be greatly improved through proper identification and understanding of

Our study entails a comparison of the effectiveness of our model for just the ISEAR data set (a data set with 4 different emotions) to evaluate scores, and then creating a new benchmarking data with two datasets and classifying 12 different emotions from those data. Our methodology involves several critical steps. First, to eliminate noise and to decrease the portion of the data, certain preprocessing methods are used, e.g., stripping punctuation, special characters, stop words, etc, we then transformed the text data into numerical vectors to be utilized by machine learning algorithms by using two different vectorization methods, which are further explained in detail.

We compared various machine learning models, including Support Vector Machine (SVM), Decision Tree, Random Forest, Logistic Regression, and XGBoost, to find the best-performing emotion detection model and determine which is best suited for multi-label classification problems. Among them, by comparing the 10-fold cross-validation scores of all the models, the XGBoost classifier was more efficient than the other models. We employed Explainable-AI (XAI), which stripping punctuation, special characters, stop words, etc , we transformed the text data into numerical vectors to be utilized machine learning algorithms and also deeply explaining the a that used in daily life

Our study has found the effectiveness of combining comprehe feature extraction with explainable AI techniques,focusing vial position of transparency of emotion detection apps.

offers transparent explanations for the transparent and human-understandable decision-making made by machine learning and artificial intelligence models. We employed various explainable models, such as LIME, SHAP, and Anchor. LIME (Local Interpretable Model-Independent Explanation) draws attention to the words that have the highest influence on the model's predictions, through which we can comprehend the model's decision-making process and it also gives validity to our results by increasing the trust by giving the reasons behind the decision taken by our model for various datasets.

SHAP (Shapley Additive Explanations) gives a global view of feature importance by comprehending how each word contributes positively or negatively to the model's predictions. Anchor explanations give the most stable and high-confidence patterns in text, ensuring that the model's predictions are made based on key constant words. We are attempting to create a machine learning model that classifies data as well as provides insight and confidence in decision-making by merging the best-performing machine learning model (i.e., XGBoost) with LIME explanations. This study emphasizes the use of transparency in black-box type models that do not provide reasons, particularly for applications using personal and sensitive information.

II. Literature Survey

To improve the emotion recognition model, various work strategies must be assessed. Numerous studies enumerate various methods and datasets utilized in emotion classification. Table I is a compilation of such studies, enumerating their findings and various methods utilized.

The ISEAR dataset, the most widely used dataset in emotion detection research, has been applied in various studies [6], [10], [16]. Support Vector Machines (SVM) and Na¨ıve Bayes (NB) are a few of the older machine learning models that have been effective in emotion classification in certain datasets [4], [8], [13], [14], and they are likely to have competitive accuracy when paired with the right feature extraction techniques, such as TF-IDF, Bag of Words (BoW), and word embeddings (Word2Vec, GloVe, FastText, BERT) [2], [4], [5], [8], [10], [14], [16], [17]. Their greatest weakness, however, is dealing with complicated relationships and context-dependent meanings, which deep learning models have tried to overcome.

Deep learning methods, i.e., Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Bidirectional LSTMs (Bi-LSTM), have been proven to be state-of-the-art in learning sequential dependencies in text data [1], [5], [7], [11], [12], [16]. Experiments with Transformer-based models like BERT and RoBERTa have been proven to be highly effective in learning deep semantic representations, resulting in state-of-the-art performance in emotion detection [1], [3], [15], [16]. These models are generally not highly interpretable, and it becomes difficult to comprehend their decision-making process, particularly in sensitive user-generated content applications.

In an attempt to close the gap between explainability and accuracy, some of the research has integrated Explainable-AI (XAI) methods, such as SHAP (Shapley Additive Explanations), LIME (Local Interpretable Model-Independent Explanations), and Anchor-based methods [11], [12], [15]. These methods give a better insight into the decision-making process of black-box models by pointing out the most contributing features that lead to a specific classification. SHAP gives global feature importance, whereas LIME gives local explanations, which aid model interpretability.

Some of the current research has also tried hybrid methods, where they integrated traditional machine learning with deep learning and explainability methods to improve both classification performance and transparency [11], [12], [15]. Based on the findings from the literature, we decided to implement a hybrid approach combining machine learning methods (SVM, Decision Trees, Random Forest, and XG-Boost) and XAI methods (LIME, SHAP, and Anchor) for high classification performance and interpretability [12], [15]. Moreover, we added an in-depth feature extraction process, i.e., TF-IDF and Bag of Words, for robustness in the model [2], [4], [8], [14]. The implementation of these technologies allows us to achieve an adequately balanced emotion classification system that is not only beneficial but also capable of explaining its decision-making process, and hence, being more dependable and trustworthy for practical usage.

III. Methodology

A. Pre-processing

The first operation in our pipeline is massive sentence-level preprocessing of data. This makes the text data clean and ready for analysis by satisfying our specific needs and removing all the irrelevant information to predict emotions from a sentence. We train on two large datasets: the ISEAR dataset and another sentiment emotion dataset. These datasets are combined into an evenly distributed set of 24,000 sentences that are spread evenly across 12 emotional states, with 2,000 sentences for each emotion. Preprocessing involves making the text lowercase for standardization and noise removal in the form of URLs, HTML tags, emails, punctuation, special characters, numbers, and extra white spaces. We also remove stop words from the NLTK stopword list to minimize frequent words that do not have any emotional significance. These preprocessing steps help reduce feature dimensionality and improve input data quality. Dimensionality reduction is especially significant for us, as we intend to use a sparse feature set that increases exponentially with increasing data. By eliminating redundant stop words, numbers, and punctuation, we reduce the dimensionality of the feature set quite significantly, making the model more efficient and effective.

TABLE I: Summary of Studies on Emotion Detection Using Text:

Paper Title	Dataset	Features Extracted	Methodology	Results
”Explainable Emotion Recognition from Tweets using Deep Learning and Word Embedding Models” [1]	Emotion Dataset from Kaggle (2021)	Word2Vec, GLoVe, FastText, BERT, XLNet, and DistilBert	Classification using deep learning models, including CNN, LSTM, BiLSTM, RNN, GRU, and BiGRU. SHAP used for model interoperability	DistilBert + CNN gave highest accuracy of about 99.0% and f1-score 98.0%. FastText obtain 97.0% accuracy and f-score of 91.0%
”Deep learning approach to text analysis for human emotion detection from big data” [2]	Custom data	Bag of Words, N-grams (Bigrams), NRC Lexicons, Social media and punctuation features.	Word2Vec for word embeddings, One-vs-Rest SVM with a linear kernel, Ensemble of Bag of Words and Word Embedding outputs with weighted averages.	Detection rate: up to 98.2% (Neutral) Classification accuracy: 98.02% (overall) Highest detectio rate: 97.22%.
”Nearest neighbour approaches for Emotion Detection in Tweets” [3]	Tweets dataset labelled into Anger, Joy, Sadness, Fear	Embeddings: roBERTa-based, DeepMoji, USE, SBERT, Word2Vec Lexicons: VAD, EMOLEX, AI, ANEW, Warriner	Evaluated individual embedding models and lexicons. Combined five embedding models using weighted kNN, assessed lexicons influence.	With the best lexicon and roBERTa combined with the best lexicon gave PCC scores of Anger: 0.7190, Joy: 0.7526, Sadness: 0.7566, Fear: 0.6804 Average PCC scores on test data were 0.635.
”Computational Approaches for Emotion Detection in Text” [4]	Web blog data, online student reviews	Keywords, Part of Speech (POS) tags, syntactic and semantic data	Keyword-based: Collect log data, tokenize, POS tagging, and use gazetteer lists for semantic annotation. Learning-based: SVM model with pre-processed data, feature vectors, and convert them to LibSVM format.	Libsvm accuracy: 96.43%
”Emotion Detection of Contextual Text using Deep learning” [5]	SemEval(2019)	Word embeddings, Preprocessing features: tokenization, lowercasing, stemming, whitespace removal, spelling correction	Bi-directional Long Short-Term Memory (Bi-LSTM) , Evaluated Word2Vec, FastText, and Glove; selected Glove for final model	The Bi-LSTM model with Glove embeddings achieved an average F1-score of 69.63 and an overall F1-score of 0.7189.
”CBE : Corpus-Based of Emotion for Emotion Detection in Text Document” [6]	WNA, ANEW, ISEAR Dataset	Valence (V), Arousal (A), Dominance (D), and categorical emotion labels	Merged WNA and ANEW, automatic tagging using Adapted LESK, expanded with ISEAR	F-Measure (WNA+ANEW): 0.50 , F-Measure (CBE with expansion): 0.61.
”Emotion Detection in Text using Nested Long Short-Term Memory” [7]	Twitter dataset (2012)	For SVM: TF-IDF	Classified using LSTM, Nested LSTM, and SVM	Nested LSTM (Highest accuracy): 99.167%, LSTM accuracy : 99.154% .
”Emotion Detection from Bangla Text Corpus Using Na¨ıve Bayes Classifier” [8]	User comments from Facebook posts	Text segmentation, stop word removal, Stemming, POS tagging, Word n-grams, Tf-idf vectorizer	Classified text using Multinomial Na¨ıve Bayes (MNB)	Best accuracy achieved: 78.6% using bigram-based tf-idf with POS tagging and both stopword and emoticon removal.
”Emotion Detection from Text and Sentiment Analysis of Ukraine Russia War using Machine Learning Technique” [9]	Twitter dataset about the conflict between Russia and Ukraine(custom dataset), Kaggle dataset(racism identification), Facebook(custom dataset)	N-grams(bigrams and trigrams), Lexical features	Classifiers: DT, LR, NB, SVM, RF. Ensemble Methods: AdaBoost, Gradient Boost, XGBoost ,VADER for sentiment analysis	Ensemble Accuracy - 90.45%, XGBoost: 90% accuracy.
”A BERT based dual-channel explainable text emotion recognition system” [10]	Isear, Aman, Affective Text, Emotion Lines Dataset.	Word Embeddings: BERT, GloVe and Weight calculation for emotion triggering words.	LSTM-CNN dual-channel system, CNN, Bi-LSTM, FFN.	Overall Best is for Aman Dataset which is recorded by an accuracy of 80.67% and the precision, recall, and F1-score are found to be 0.86, 0.81, and 0.83, respectively.
”Multimodal Attentive Learning for Real-time Explainable Emotion Recognition in Conversations” [11]	IEMOCAP, MELD dataset.	Sematic features, Acoustic features, Utterances, Embeddings: BERT, Word2Vec.	Bi-LSTM, ERLDK, DDIN. LIME used for model interoperability	Best accuracy and f1-score were achieved by Proposed method with an Accuracy – 63.89%, Avg F1 score – 65.4%.
”An explanation framework and method for AI-based text emotion analysis and visualisation” [12]	Emotion-stimulus dataset	Emotion, Emotion Cause Extraction, Emotion-triggering words.	Logistic Regression, SVM, Bi-LSTM, Bi-LSTM + Attention Mechanism and LIME , SHAP, and EAX models for explanation framework.	Bi-LSTM + Attention being the most effective among all by giving the best scores. Precision – 98.19%, Recall – 94.77%, F1-score – 96.4%.
”Sentimental Analysis on Student Feedback using NLP and POS Tagging” [13]	Real-time feedback collected from students via the institution’s online educational portal.	Tokenization, Part-of-Speech (POS) Tagging, SentiWordNet (Positive and Negative words are extracted).	The system uses NLP-based sentiment classification on textual feedback. The polarity (positive/negative/neutral) is predicted for each comment. Results are visualized using Matplotlib and Plotly via a web-based interface.	The system emphasizes Positive Feedback Trends and Negative Feedback trends. Performance increases steadily with training time. Visual trend shows performance improves over time.
”A model for sentiment and emotion analysis of unstructured social media text” [14]	Emotion dataset, SMS dataset.	Unigram, Bigram, POS(Parts-of-speech)tags, Syntactic and semantic features.	Unsupervised approach: SentiWordNet lexicon for sentiment classification and Supervised approach: Multinomial Naive Bayes(MNB), SVM.	The scores recorded for unsupervised approach are with an accuracy of 80.68% and for supervised approach the accuracy score 87.78% for the Multinomial Naive Bayes(MNB)model.
”Social Media Hate Speech Detection Using Explainable Artificial Intelligence (XAI)” [15]	Google jigsaw Dataset, HateXplain Dataset	Count Vectorizer, TF-IDF	For Google jigsaw dataset: LSTM, Multinomial Naive Bayes(MNB), Decision Tree, Random Forest. For HateXplain: BERT+ANN(Artificial neural network), BERT+MLP(Multi-layer perceptron) and LIME for explainability.	For Google jigsaw dataset: LSTM gives the best accuracy score of 97.6% and For HateXplain: BERT+ANN gives the best accuracy score of 93.67%.
”Semantic-Emotion Neural Network for Emotion Recognition from Text ” [16]	DailyDialogs, ISEAR, CrowdFlower, Tales-Emotion, Electoral-Tweets, TEC, EmoInt, Grounded Emotions, Emotion cause, SSEC Datasets.	BoW(Bag-of-words), TF-IDF and Word embeddings: Word2vec, GloVe, FastText, EWE(emotion word embeddings).	Proposed Model: SENN(Semantic-Emotion Neural Network) consists of two sub-networks Bi-LSTM and CNN and GRU, BiGRU, LSTM.	The best-performing model was the SENN model with FastText word embedding achieved for the Grounded emotions dataset with an f1-score of 98.8%.
”Automated Human Emotion Recognition and Analysis using Machine Learning” [17]	FER2013, Sentiment analysis Dataset(Custom).	Tokenization, Stopword Removal, TF-IDF Vectorization	Multi-Layer Perceptron, Genetic Algorithm which is Optimized MLP weights through evolution, Fuzzy systems (Guassian)	Overall Feed-Forward Neural Network model has got the best accuracy with 96.37% train, 88% test and Genetic Algorithm Optimized MLP with 91.3% train, 86.867% test and tuned Multi-Layer Perceptron got an train accuracy of 95%.

B. Feature Generation

After preprocessing, we used two prominent vectorization techniques, Term Frequency-Inverse Document Frequency(TF-IDF) and Bag of Words(BoW), to generate features from the preprocessed clean text. We incorporated ngram(1-2) for generating the feature matrix from the above-mentioned techniques, i.e., both TF-IDF and BoW are extracted for unigram, bigram(set of two consecutive words), as the feature set.

Term Frequency-Inverse Document Frequency (TF-IDF) TF-IDF captures the semantic meaning of the data by converting the text data into a matrix of features which reflect the importance of words in the context of the entire dataset. TF-IDF is computed as follows:

$$\text{TF-IDF}(t, d) = \text{TF}(t, d) \times \text{IDF}(t) \quad (1)$$

where:

$$\text{TF}(t, d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \quad (2)$$

Here, $f_{t,d}$ is the number of times term t appears in document d, and the denominator is the total count of all terms in document d.

$$\text{IDF}(t) = \log \frac{N}{1 + \text{DF}(t)} \quad (3)$$

where N is the total number of documents, and $\text{DF}(t)$ is the number of documents containing term t. The "+1" in the denominator prevents division by zero.

Bag of Words (BoW) BoW generates a matrix of token counts from the text data similar to one-hot encoding, representing the frequency of words within each document. Each document d is represented as a vector:

$$\mathbf{V}_d = [c_1, c_2, \ldots, c_n] \quad (4)$$

where c_i represents the count of the i^{th} word in the vocabulary appearing in document d.

We combined all the features obtained using both techniques to obtain the final feature matrix, which is then used for model training.

C. Model Training and Evaluation

We evaluated the performance of various machine learning classifiers, namely SVM, Decision Trees, Random Forest, Logistic regression, and XGBoost, to determine the most efficient model for our task. First we encoded the target variable, which represents 12 different emotions using LabelEncoder. Then the dataset is then split into training and testing sets to perform model evaluation. We trained all the classifiers mentioned above on the combined feature matrix of TF-IDF and BoW. We compared the performance of each model based on its accuracy in predicting emotions from the test set. Among these, the XGBoost classifier demonstrated superior performance than other classifier, which can be attributed to its ability to handle complex patterns and interactions in the data.

D. XGBoost Classifier

The XGBoost classifier was the highest-ranked model among others. We used the XGBClassifier, which was trained using a combined feature matrix, and cross-validated its performance with 10-fold cross-validation. This guarantees the model's performance to be robust and generalizable. Cross-validation allows us to test the performance of the model on unseen data, hence reducing the risk of overfitting. The XGBoost's high accuracy shows its suitability for emotion detection tasks, which can identify subtle text nuances well, and this is crucial for sentiment analysis to be accurate.

E. Explainability with LIME

To provide interpretability to the predictions of the model, We adopt a Local interpretable model-agnostic explanation(LIME). Due to the sensitivity of emotion detection applications, it is necessary to know the reason behind the model's decisions. LimeTextExplainer is employed to interpret the predictions provided by the XGBoost model. By generating the reasons behind individual predictions, LIME finds and visualizes the most important features that are responsible for the model's decisions. Such interpretability is a necessity to attain insights into the behavior of the model, transparency, and establishing user trust. LIME helps in finding out which words or phrases in the text are responsible for the emotion classification. This ensures the model's performance to be robust and generalizable.

F. Explainability with SHAP

SHAP (Shapely Additive explanations) is applied to explain further how the features are forcing the model towards its predictions. SHAP provides global feature importance and also supports local interpretability. The SHAP values are indicative of the magnitude to which every word in a sentence is elevating or dropping the probability of a specific emotion being predicted.

SHAP provides a breakdown of why the model predicts a certain thing by giving each word in a piece of text a contribution score that can either add to or detract from the probability of a given emotion being predicted. It gives an explanation of the degree to which a feature contributes to a prediction and aids in explaining which words have greater influence on classification.

G. Explainability with Anchor

Anchor explanations derive the key decision rules of the model. They act like rules in a rule-based system since they specify the conditions or if-then patterns that need to occur to make a prediction. They are called anchors because of their decisive role in the prediction. Anchor explanations specify a set of words or phrases that the model uses to make a specific prediction when recognized.

H. Robustness Checks with NTK and SVM

To ensure our model remains stable, we conduct additional studies using a Neural Tangent Kernel (NTK) and a Support Vector Machine (SVM). We employ an SVM to ensure the model's performance is stable and reliable with respect to different subsets of data by utilizing a single NTK. This includes measuring the accuracy of the model over a number of cycles using different randomly selected subsets of the data.

The Neural Tangent Kernel (NTK) is given by:

$$\Theta_{\text{NTK}}(x, x') = \sigma_w^2 \, \mathbb{E}_{z\sim N(0,I)} \; \phi(x)^T \phi(x') \; + \sigma_b^2 \qquad (5)$$

where:

- $\Theta_{\text{NTK}}(x, x')$ is the NTK function.
- σ_w^2 and σ_b^2 are the variances of weights and biases, respectively.
- $\phi(x)$ is the activation function applied to input x.
- $\mathbb{E}$ denotes the expectation over a standard normal distribution $N(0, I)$.

With the aid of robustness checks, we improve the overall performance of the model in different scenarios, validating the credibility of our results. The NTK captures the relationships in the data which aids with the complexity and provides additional robustness to the model.

We have developed an effective and explainable emotion detection model using multi-level data preprocessing, feature generation, model training, and interpretability techniques. The model uses advanced machine learning and model-agnostic interpretability methods to ensure accuracy and transparency. This is especially crucial for sensitive emotions model predictions, where trust in the model along with high performance is imperative.

IV. Experiments & Results

A. About Dataset

Choosing a dataset plays an important role in the model's predictions and analysis so after a through investigation and indepth reaserch on 17 other related works we came up with a combined dataset,out of all the datasets we found in used in others works, either we encounted the problem of class imbalance / data insufficiency.

We have chosen 2 datasets to work on this problem, namely ISEAR and Emotion sentiment dataset. The ISEAR dataset contains a total of 9621 values categorized into four emotion classes. The classes are distributed as follows: Anger with 2252 instances, Fear with 1701 instances, Joy with 1616 instances, and Sadness with 1533 instances. The distribution of these classes is visually represented in Fig. 2. The Emotion sentiment dataset contains a total of 840,000 values categorized into several emotion classes. The classes are distributed as follows: Fun with 2655 instances, Enthusiasm with 2481 instances, Surprise with 1912 instances, Empty with 1454 instances, Worry with 1242 instances, Boredom with 33 instances, Neutral with 180,137 instances, Love with 10,343 instances, Happiness with 7289 instances, Sadness with 4596 instances, Relief with 4416 instances, Hate with 4127 instances, and Anger with 3352 instances. A bar plot visualizing this distribution is shown in Fig.1. Both dataset is structured into a single DataFrame where each class has been uniformly sampled to contain 2000 instances.

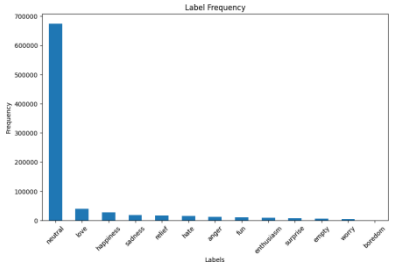

Fig. 1: Label frequency Bar plot for dataset-1

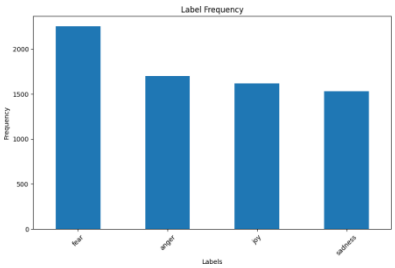

Fig. 2: Label frequency Bar plot for dataset-2

B. Dataset-t-SNE plot

The plots below indicate the non-uniformity of the data as the colors are not properly clustered together. We can observe a dense cluster of different colors in the t-SNE plot, indicating uniformity in the data.

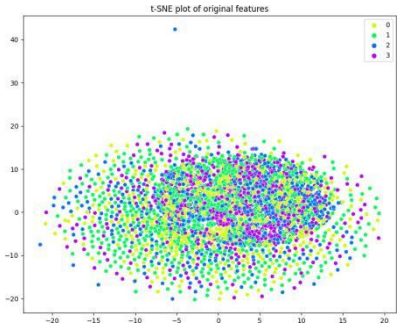

Fig. 3: t-SNE plot for ISEAR dataset

The t-SNE visualization for the ISEAR dataset in Fig. 3 shows that the data points are not well separated, suggesting overlapping features among emotion classes. Similarly, the t-SNE plot for the combined dataset in Fig. 4 further illustrates the distribution of different emotion categories, where clusters

remain dispersed and overlapping, highlighting the complexity of classification.

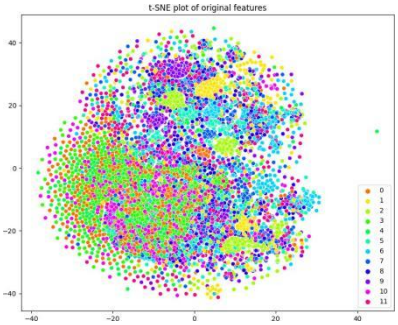

Fig. 4: t-SNE plot for combined dataset

C. Accuracies for ISEAR dataset

The 10-fold cross-validation score for the XGBoost model is 0.8541, outperforming the other models. We can have an overview from Table II.

ML-model	Accuracy
SVC (linear)	0.897
SVC (rbf)	0.8
Decision Tree	0.835
Random Forest	0.85
XG Boost	0.855

TABLE II: Accuracies for 4-class ISEAR dataset

D. Accuracies for combined 12-class dataset

As we can see from Table III that XG Boost model stood out with the best accuracy and a 10-fold cross-validation [0.9196] score among other ML models.

ML-model	Accuracy
SVC (linear)	0.92
SVC (rbf)	0.88
Decision Tree	0.89
Random Forest	0.90
XG Boost	0.92

TABLE III: Accuracies for 12-class combined dataset

E. XG Boost classification report

Class	Precision	Recall	F1-Score	Support
Class 0	0.91	0.74	0.81	406
Class 1	0.99	0.95	0.97	398
Class 2	0.99	0.99	0.99	401
Class 3	0.70	0.89	0.79	407
Class 4	0.95	0.84	0.89	390
Class 5	0.97	0.98	0.97	398
Class 6	0.96	0.98	0.97	399
Class 7	0.98	0.96	0.97	447
Class 8	0.82	0.95	0.88	365
Class 9	0.98	0.95	0.97	397
Class 10	0.86	0.81	0.84	385
Class 11	0.99	0.99	0.99	407
Accuracy	0.92			
Macro Avg	0.93	0.92	0.92	4800
Weighted Avg	0.93	0.92	0.92	4800

TABLE IV: Classification Report

Evaluation scores for XGBoost, as summarized in Table IV show precision, recall, F1-score for each class. The model performed well with an accuracy of 92%.

F. Confusion Matrix

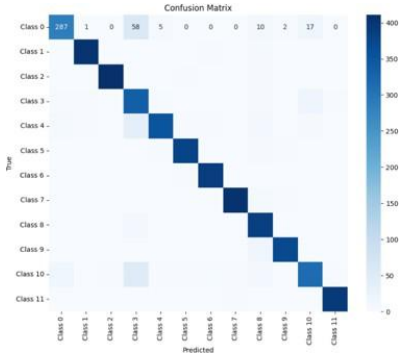

Fig. 5: Confusion matrix

The Confusion matrix is analyzed to see how our XGBoost model is performing. The diagonal line in Fig. 5 shows that most of the classes are correctly classified. Class 1 (Joy), Class 6 (Love), and Class 11 (Anger) show good accuracy with minor misclassifications. Class 0 (Sadness) and Class 10 (Surprise) are misclassified as Class 2 (Neutral) and Class 3 (Fear), respectively, in some cases.

G. Lime explanations

From Fig. 6 the model suggests that the text likely expresses ”sadness” with a high probability of 87%. The word ”lost” strongly influences this prediction. Other words like ”nice” and ”good” have little impact on changing this classification. Overall, the text’s emotional tone is predominantly ”sad,” with minimal probabilities for other emotions like ”fear,” ”anger,” and ”fun.”

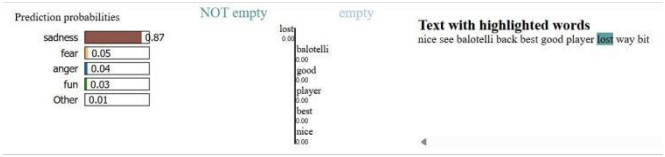

Fig. 6: LIME Explanation 1 - The word “lost” strongly influences classification as Sadness.

The model output from Fig. 7 is sure that the text is in the ”empty” category. The word ”avoid” is the main reason for this prediction.

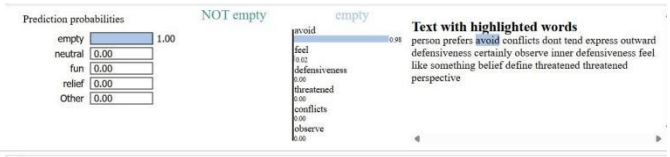

Fig. 7: LIME Explanation 2 - The word “avoid” contributes significantly to classification as Empty.

H. SHAP explanation

Fig. 8: SHAP force plot showing word contributions to classification.

As shown in Fig. 8, words like ”people” and ”excited” try to push the score up (red). Words like ”feel”, ”happy” and ”unhappy” reduced the score (blue). The final score of 0.01 is obtained by a combination of all these, indicating a shift away from the original class.

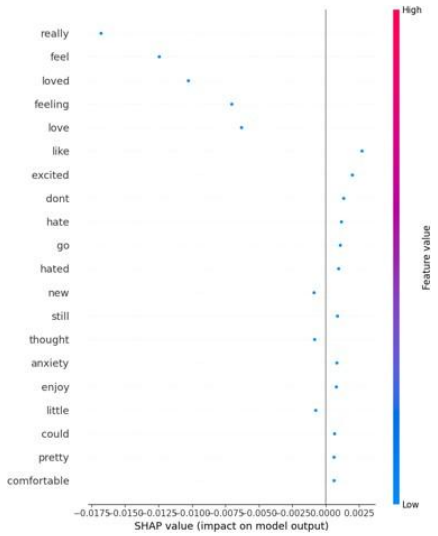

Fig. 9: SHAP summary plot displaying the most influential words.

The Fig. 9 visualizes the most influential words across multiple predictions. Words like ”feel”, ”love”, ”excited”, ”happy” had a positive impact whereas words like ”hate”, ”anxiety”, ”hated” pulled predictions towards negative emotions.

I. Anchor explanation

```
Anchor explanation for:  feel life lead simply
    contented god given us instead troubled wedding
    reception block grand wanted
predicted value:  happiness
Anchor (if any): ['contented', 'feel']
Precision: 1.0
Coverage: 0.0
```

The Anchor model is evaluated for the preprocessed sentence as shown in the above output. The model predicted the class Happiness using features ”feel” and contented”. Precision 1 implies that it always predicts happiness whenever ”feel” and ”contended” are used. Whereas coverage 0 tells us that the rule applies to a very specific subset of data.

```
Examples where anchor applies and model predicts happiness:

feel UNK lead UNK contented UNK given us instead troubled wedding UNK UNK UNK UNK
feel UNK UNK UNK contented UNK UNK us UNK UNK UNK UNK block UNK UNK
feel UNK lead simply contented god UNK UNK instead troubled UNK UNK block grand wan
feel UNK lead simply contented UNK given us instead UNK wedding UNK UNK UNK UNK
feel life lead simply contented god UNK UNK instead troubled UNK UNK block UNK UNK
feel life UNK UNK contented UNK UNK UNK instead UNK wedding UNK block UNK UNK
feel life UNK UNK contented god UNK us instead UNK wedding reception UNK grand want
feel UNK UNK UNK contented UNK UNK UNK UNK troubled UNK UNK UNK grand wanted
feel life UNK simply contented god UNK UNK instead troubled UNK reception block UNK
feel UNK lead simply contented UNK UNK us instead troubled UNK reception UNK UNK wa
```

Fig. 10: Examples where the model predicts happiness.

Furthermore, from Fig. 10 we can see that multiple similar sentences were tested, and the model correctly predicted happiness if there was the presence of ”feel” and ”contended”.

J. ROC-AUC Curve

The Receiver Operating Characteristic (ROC) curve evaluated the performance by plotting the true positive rate (TPR) against the false positive rate (FPR). Area Under the Curve (AUC) tells how well the model can classify.

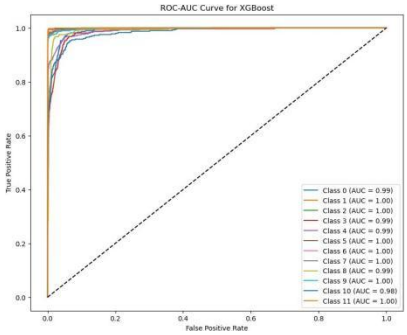

Fig. 11: ROC-AUC Curve for XGBoost model.

From Fig. 11, we observe that XGBoost achieves high AUC scores across multiple classes, with most values close to 1. This indicates a strong classification ability with minimal misclassifications.

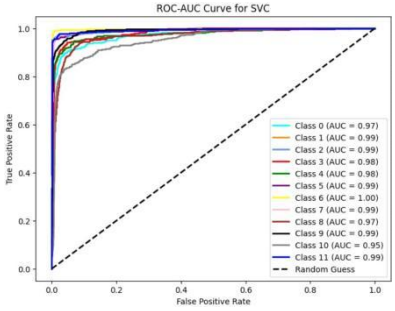

Fig. 12: ROC-AUC Curve for SVC model.

Fig. 12 represents the ROC-AUC curve for the Support Vector Classifier (SVC). The performance is slightly lower than XGBoost in some classes, with a few AUC values below 1.

Final Observation: Area Under the Curve only shows True Positives vs False Positives, so it’s possible to have a high

accuracy here, and when true negatives and false negatives are included, the accuracy can drop. You can refer to the confusion matrix in Fig. 5 to give us more and better reliable insights.

V. Discussion

The key observations concluded in our experiments are discussed below:

A. Model Performance

XGBoost was the best performing model. It had an accuracy of 92% and an AUC close to 1 for most classes. The results of the confusion matrix were also promising, with only minor emotion misclassifications such as sadness and surprise.

B. Explainability Analysis

To interpret the predictions of our models, we employed three explainability techniques: LIME, SHAP, and Anchor explanations.

1) LIME: Local model explanations suggested that words carrying strong emotional overtones had the greatest impact on predictions. In one instance, the occurrence of the word *lost* resulted in a high prediction probability of sadness.

2) SHAP: The SHAP analysis provided global feature importance. The words *feel*, *love*, and *happy* positively impacted predictions, whereas the words *hate* and *anxiety* influenced the model towards negative emotions.

3) Anchor: The Anchor explanations revealed that the model heavily relied on specific keywords when making predictions. For instance, the words *contented* and *feel* were crucial for predicting happiness.

VI. CONCLUSION

Lastly, we have used different machine learning models for classification of emotions. In our study XGBoost was the best model. It gave 92% accuracy and was better than other models such as SVC and Decision tree. AUC curve and Confusion matrix also confirmed and displayed it classified right for the majority of the classes.

For interpretation, we have utilized LIME, SHAP and Anchor explanations. LIME tells us which words contribute most towards that class prediction. For example, Class sadness was anticipated due to the input of the word ”lost”. SHAP tells us with a wider perspective what words are responsible on average basis. For example, word like ”happy” has a positive effect on emotion prediction and words like ”anxiety” predict it negatively. Anchor tells us the model decision relies on the key term pattern.

Our paper suggests that XGBoost, combined with such interpretable models, provides an explainable and effective method of emotion classification. The experimental use of deep learning methods in future research can be employed to further improve the model.

VII. References

References

[1] A. M. Abubakar, D. Gupta, and S. Palaniswamy, “Explainable emotion recognition from tweets using deep learning and word embedding models,” in *2022 IEEE 19th India Council International Conference (INDICON)*, pp. 1–6, IEEE, 2022.

[2] J. Guo, “Deep learning approach to text analysis for human emotion detection from big data,” *Journal of Intelligent Systems*, vol. 31, no. 1, pp. 113–126, 2022.

[3] O. Kaminska, C. Cornelis, and V. Hoste, “Nearest neighbour approaches for emotion detection in tweets,” *arXiv preprint arXiv:2107.05394*, 2021.

[4] H. Binali, C. Wu, and V. Potdar, “Computational approaches for emotion detection in text,” in *4th IEEE International Conference on Digital Ecosystems and Technologies*, pp. 172–177, 2010.

[5] U. Rashid, M. W. Iqbal, M. A. Skiandar, M. Q. Raiz, M. R. Naqvi, and S. K. Shahzad, “Emotion detection of contextual text using deep learning,” in *2020 4th International symposium on multidisciplinary studies and innovative technologies (ISMSIT)*, pp. 1–5, IEEE, 2020.

[6] F. H. Rachman, R. Sarno, and C. Fatichah, “Cbe: Corpus-based of emotion for emotion detection in text document,” in *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pp. 331–335, IEEE, 2016.

[7] D. Haryadi and G. P. Kusuma, “Emotion detection in text using nested long short-term memory,” *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, 2019.

[8] S. Azmin and K. Dhar, “Emotion detection from bangla text corpus using naive bayes classifier,” in *2019 4th International Conference on Electrical Information and Communication Technology (EICT)*, pp. 1–5, IEEE, 2019.

[9] A. Al Maruf, Z. M. Ziyad, M. M. Haque, and F. Khanam, “Emotion detection from text and sentiment analysis of ukraine russia war using machine learning technique,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 12, 2022.

[10] P. Kumar and B. Raman, “A bert based dual-channel explainable text emotion recognition system,” *Neural Networks*, vol. 150, pp. 392–407, 2022.

[11] B. Arumugam, S. D. Bhattacharjee, and J. Yuan, “Multimodal attentive learning for real-time explainable emotion recognition in conversations,” in *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1210–1214, 2022.

[12] Y. Li, J. Chan, G. Peko, and D. Sundaram, “An explanation framework and method for ai-based text emotion analysis and visualisation,” *Decision Support Systems*, vol. 178, p. 114121, 2024.

[13] N. R, P. M. S, P. P. Harithas, and V. Hegde, “Sentimental analysis on student feedback using nlp pos tagging,” in *2022 International Conference on Edge Computing and Applications (ICECAA)*, pp. 309–313, 2022.

[14] J. K. Rout, K.-K. R. Choo, A. K. Dash, S. Bakshi, S. K. Jena, and K. L. Williams, “A model for sentiment and emotion analysis of unstructured social media text,” *Electronic Commerce Research*, vol. 18, pp. 181–199, 2018.

[15] H. Mehta and K. Passi, “Social media hate speech detection using explainable artificial intelligence (xai),” *Algorithms*, vol. 15, no. 8, 2022.

[16] E. Batbaatar, M. Li, and K. H. Ryu, “Semantic-emotion neural network for emotion recognition from text,” *IEEE Access*, vol. 7, pp. 111866–111878, 2019.

[17] K. S. Raj and P. Kumar, “Automated human emotion recognition and analysis using machine learning,” in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 1–9, 2021.