2025 International Conference on Electrical, Computer and Communication Engineering, (ECCE) February 2025 in CUET, Bangladesh

2025 International Conference on Electrical, Computer and Communication Engineering (ECCE) | 979-8-3503-5750-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/ECCE64574.2025.11013403

Enhancing Multi-Emotion Detection in Text: A Comparative Study of Feature Extraction Techniques and Machine Learning Models

Meherunnesa Hossain Ibnath*, Khan Md Hasib†, Md. Rahid Parvez ‡, M.F. Mridha§
*Department of Computer Science and Engineering, Ahsanullah University of Science and Technology, Dhaka, Bangladesh
†Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka, Bangladesh
‡ ICT Cell, Bangabandhu Sheikh Mujib Medical University (BSMMU), Dhaka, Bangladesh
§Department of Computer Science, American International University Bangladesh, Dhaka, Bangladesh
meherunnesa.cse.20200204101@aust.edu, khanmdhasib.aust@gmail.com, rahid@bsmmu.edu.bd, firoz.mridha@aiub.edu

***Abstract*—Emotion detection in text is one of the major tasks in NLP that might have tremendous applications in sentiment analysis, human-computer interaction, and mental health diagnosis. This study evaluates the performance of various feature extraction techniques and classification models for multi-emotion detection, using a dataset of 15,996 samples divided with six emotion labels: Sadness, Joy, Love, Anger, Fear, and Surprise. Three feature extraction methods—TF-IDF, Count Vectorization, and N-grams—were applied to preprocess the text data. Six models—KNN, SVM, NB, DT, LR, and BiLSTM—were then used to classify emotions. The best results among the 18 feature-model combinations are TF-IDF with SVM at 86.15%, Count Vectorization with Logistic Regression at 86.25%, and N-grams with SVM at 86.06%. N-grams combined with LR also performed well, achieving an accuracy of 88.09%, underlining the importance of capturing contextual and sequential information in text classification. This research contributes to improving NLP-based emotion detection systems and provides insights for future work, particularly in domains requiring nuanced emotion recognition from text data.**

***Index Terms*—Emotion detection, NLP, Multi-emotion detection, TF-IDF, Count Vectorization, N-grams, KNN, SVM, Naive Bayes, Decision Tree, Logistic Regression, Bi-LSTM, Feature extraction, Comparative analysis.**

I. Introduction

Emotion detection in text is important because of applications related to social media analysis, customer feedback interpretation, and human-computer interaction. Accurately identifying emotions helps in understanding user sentiment, leading to more responsive and intuitive applications. Despite the developments made in NLP. Our aims is to refine emotion classification by exploring and evaluating different feature extraction techniques and learning models, with the goal of improving the accuracy and effectiveness of emotion detection in text. In our opinion, dealing with such challenges will surely be a step ahead in significant development towards strong and effective systems in NLP.

This research tries to fill these gaps in existing literature by comparing several techniques for feature extraction and utilizing BiLSTM to identify emotions in multilabels.

- **Limited exploration of feature extraction methods**: Previous studies [1], [2], and [3] primarily focus on single techniques (e.g., TF-IDF) without comparing different approaches, whereas our research explores multiple feature extraction methods to enhance model performance.
- **Underutilization of advanced deep learning models**: Many studies [2], [4], and [5] emphasize traditional machine learning models, neglecting advanced models like BiLSTM. Our research integrates BiLSTM, which offers improved performance for emotion classification tasks.
- **Lack of multi-label emotion classification**: Prior works [3], [6], and [7] primarily target binary sentiment classification, while our approach addresses six distinct emotion categories through multi-label classification.
- **Absence of hybrid feature-model combinations**: Existing research [4], [6], and [8] typically investigates individual methods or models, But we propose a hybrid approach which couples the feature extraction methods with DL models for better performance.
- **Inconsistent performance benchmarks:** Some studies [5] lack standardized metrics for evaluating emotion detection, while our study uses accuracy, precision, recall, and F1 score for comparison.

This study sets a benchmark by evaluating models like SVM, Logistic Regression, and BiLSTM, which are commonly used in emotion detection as shown in previous works. For example, Chaffar and Inkpen [1] and Gupta and Srinivasan [2] achieved good results with SVM and Random Forest, respectively. Our results show that SVM and LR with N-grams or TF-IDF outperform other models, aligning with literature findings. Additionally, BiLSTM's performance confirms the effectiveness of deep learning models in capturing contextual and sequential information for emotion detection.

II. Literature Review

Emotion detection in text has been extensively studied, with various approaches proposed to enhance classification accuracy. This literature review summarizes key contributions in the field, highlighting the diverse methods and results reported by different researchers.

979-8-3503-5750-9/25/$31.00 ©2025 IEEE

Chaffar and Inkpen [1] found that Support Vector Machines (SVM) outperformed other methods, demonstrating the effectiveness of combining diverse data sources and features for improved emotion detection. Gupta and Srinivasan [2] showed that while Random Forest achieved high training accuracy, SVM performed best in test accuracy, emphasizing the need for dataset balancing and hyperparameter tuning to avoid overfitting. Nandwani and Verma [3] reviewed sentiment and emotion detection techniques, and highlighted the crucial role of preprocessing and feature extraction in enhancing classification performance. Aman and Szpakowicz [4] focused on identifying emotions in text using blog posts and Ekman's basic emotions. Their study achieved 73.89% accuracy with SVM and underscored the value of external knowledge resources like WordNet-Affect. Alm et al. [5] classified emotion categories in children's fairy tales using machine learning methods; the highest accuracy was achieved with Naive Bayes. This work illustrated the capability of machine learning in emotion recognition and how context helps in improving the accuracy of classification. The study conducted by Rout et al. [6] on sentiment and emotion analysis of social media text presented results from an unsupervised approach for sentiment analysis with an accuracy of 80.68% and Multinomial Naive Bayes for emotion classification with a result of 95.3%. Alfarizi et al. [7] combined TF-IDF with LSTM for emotion classification, achieving a high accuracy of 97.50%. Their research demonstrated the superior performance of LSTM with TF-IDF compared to simpler models like LinearSVC. Nahin et al. [8] proposed a novel method integrating keystroke dynamics with text pattern analysis for emotion detection. Their approach, using SVM with Jaccard similarity, achieved over 80% accuracy. Gupta et al. [9] introduced the SS-LSTM model, which integrates sentiment and semantic embeddings for emotion classification. Desmet and Hoste [10] developed an emotion detection system for analyzing suicide notes using binary SVM classifiers with lexical and semantic features. Their system achieved up to 68.86% F-scores, demonstrating variable performance across different emotion categories. Ou and Li [12] proposed a cross-modal mapping network with a GCN architecture for multi-modal sarcasm detection, leveraging external knowledge and retrieval-based attention to significantly enhance performance on benchmark datasets. [13] developed a hybrid deep learning model integrating LSTM, CNN, and attention mechanisms with sentiment data aiding prediction for over 70% of studied cryptocurrencies.

By integrating advancements in natural language processing and machine learning, these works contribute to more accurate and reliable emotion classification, offering valuable insights for improving emotional understanding in text-based data.

III. Dataset

A. Dataset Collection

This study employs a dataset sourced from Hugging Face, specifically curated for emotion detection in textual data. The dataset comprises 15,996 samples, each annotated with one of six distinct emotion labels.

1) Dataset Characteristics:

- **Source**: The dataset was obtained from Hugging Face, a widely used repository for NLP datasets. It can be accessed at Emotion Detection Dataset.
- **Size**: A total of 15,996 samples, providing a robust foundation for training and evaluating models.
- **Labels and Distribution**:
  - Joy (Label 1): 5,361 samples
  - Sadness (Label 0): 4,663 samples
  - Anger (Label 3): 2,159 samples
  - Fear (Label 4): 1,937 samples
  - Love (Label 2): 1,304 samples
  - Surprise (Label 5): 572 samples

  This varied distribution ensures comprehensive coverage of emotional expressions, enabling effective model training and evaluation. Shown in Fig.1

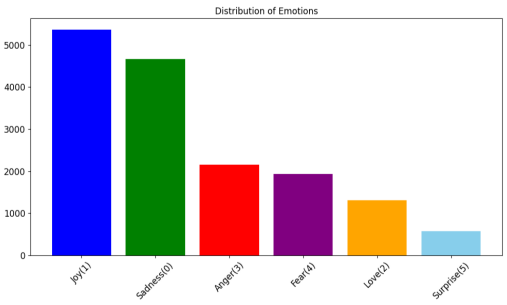

Fig. 1: Distribution of Emotion Labels

B. Dataset Cleaning

The dataset underwent a thorough cleaning process to ensure quality and consistency:

- **Removing Missing Values**: Samples with missing data were removed.
- **Checking Labels**: Each text sample was confirmed to have an accurate and complete emotion label.

This resulted in a refined dataset, free from missing entries or incomplete annotations, forming a reliable foundation for further analysis.

IV. Methodology

A. Introduction to Methodology

This section outlines our methodology for evaluating feature extraction techniques and ML models for multi-emotion detection. In our work, we utilized the Hugging Face dataset, which had six labels of emotions: Sadness, Joy, Love, Anger, Fear, and Surprise. After the preprocessing of the data, three feature extraction methods were applied: TF-IDF, Count Vectorization, and N-grams; six models assessed: kNN, SVM, DT, LR, NB, Bi-LSTM.

We defined evaluation metrics and experimental procedures to measure model performance and feature extraction effectiveness, ensuring a thorough analysis of each combination's impact on multi-emotion detection. shown in Fig.2

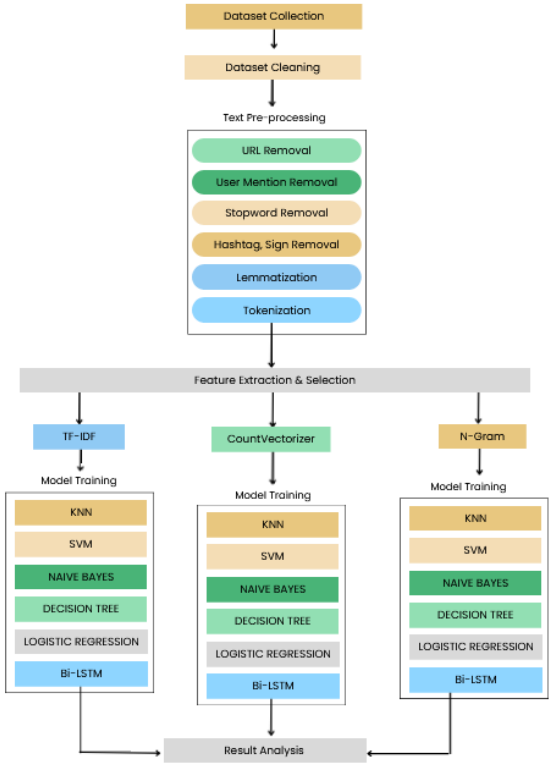

Fig. 2: Overview of the Methodology

B. Text Preprocessing

The dataset underwent several preprocessing steps to ensure clarity and relevance:

- **URL and User Mention Removal**: URLs and user mentions, for example, @username, were removed to focus on the content.
- **Stopword Removal**: Common stopwords (e.g., *"and"*, *"the"*) were removed to reduce noise.
- **Hashtag and Emoji Removal**: Hashtags and emojis were stripped out to simplify and standardize the text.
- **Sign Removal**: Punctuation and non-alphanumeric characters were removed for cleaner text.
- **Lemmatization and Tokenization**: Words were lemmatized to their root forms and tokenized into individual words for effective analysis.

These preprocessing steps, implemented using `nltk` and `spaCy` libraries, prepared the dataset for feature extraction and model training by ensuring a clean and standardized text corpus.

C. Feature Extraction and Selection

We employed three techniques—TF-IDF, Count Vectorization, and N-grams—to transform and prepare the text data.

*1) **TF-IDF (Term Frequency-Inverse Document Frequency)**:* TF-IDF technique was used to compute the feature extraction, computing TF-IDF scores. It selected the top 5,000 terms based on the TF-IDF score and hence guaranteed the inclusion of most informative terms during model training and analysis. Shown in Fig. 3

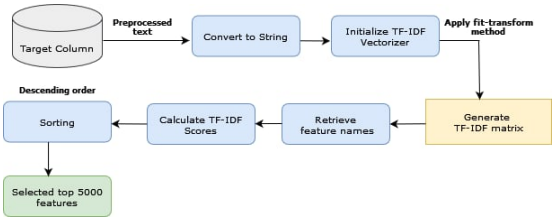

Fig. 3: TF-IDF Feature Extraction Technique

*2) **Count Vectorization**:* Count Vectorization was applied to extract features by calculating the frequency of each term across all documents, identifying the most frequently occurring terms. For feature selection, the top 5,000 terms with the highest counts were chosen to retain the most significant features.

*3) **N-gram (Unigram and Bigram)**:* N-gram analysis was employed to extract features by computing the occurrences of contiguous sequences of words, including both unigrams and bigrams. It captures individual words and their combinations, top 5,000 N-grams with the highest frequencies were selected to include the most relevant N-grams for model training.

D. Model Implementation

Here, we applied five ML models and one DL model to the text data processed through three feature extraction techniques.

1) Applied Models:

- **K-Nearest Neighbors (KNN)**: It is a classification of text based on the similarity of feature vectors to the nearest neighbors in the feature space.
- **Support Vector Machine (SVM)**: This technique finds the optimal hyperplane that separates various classes of emotions.
- **Multinomial Naive Bayes (NB)**: Employs a probabilistic approach for text classification based on term frequencies.
- **Decision Tree (DT)**: Utilizes a hierarchical tree structure to make classification decisions based on feature values.
- **Logistic Regression (LR)**: Estimates class probabilities using a logistic function applied to feature values.
- **Bidirectional Long Short-Term Memory (BiLSTM)**: An advanced LSTM variant that processes sequences in both forward and backward directions, capturing contextual dependencies from both past and future.

2) Model Training:

- **Data Preparation**: The data has been divided into an 80/20 split for training/testing.
- **Training Process**: The training set is used for the training of all the models based on their respective default hyperparameter values. The BiLSTM model was run for 10 epochs with a batch size of 64. The Adam optimizer was used, which tunes learning rates during the training process itself.

This comprehensive approach aimed to identify the most effective feature extraction and modeling strategies for accurate emotion detection.

E. Model Evaluation

Models were evaluated using several key metrics:

- **Accuracy**: Correct predictions out of total predictions.
- **Precision**: TP out of the the sum of true positives and false positives.
- **Recall**: TP to the sum of true positives and false negatives.
- **F1-Score**: The harmonic mean of precision and recall, offering a well-rounded measure of both.
- **Error Rate**: Number of incorrect predictions, calculated as $1 -$ Accuracy.

V. Results Analysis

We compare the performances of six different machine learning models that have gone through three various feature extraction techniques: TF-IDF, Count Vectorization, and N-grams. Then, we test each of our models on several metrics.

A. TF-IDF

TF-IDF stands for Term Frequency-Inverse Document Frequency method and calculates word importance in a document with respect to the rest of the corpus. It is effective in highlighting the most informative features and downweighting less significant words. The performance metrics for each model using the TF-IDF technique are displayed in Table I.

TABLE I: Performance Metrics for Models Using TF-IDF

Model	Accuracy	Precision	Recall	F1-Score	Error Rate
KNN	77.59%	77.98%	77.59%	76.93%	22.40%
SVM	**86.15%**	**86.04%**	**86.15%**	**85.92%**	**13.84%**
MNB	67.75%	76.19%	67.75%	61.09%	32.25%
DT	81.28%	81.38%	81.28%	81.31%	18.71%
LR	83.71%	84.06%	83.71%	83.03%	16.28%
Bi-LSTM	81.71%	81.66%	81.71%	81.29%	18.28%

It can be observed that the SVM model outperforms the rest, with an accuracy of 86.15%, impressive precision of 86.04%, and a recall of 86.15%. This performance can be attributed to SVM's capability of finding an optimal hyperplane in high-dimensional space, which works well with the TF-IDF features. A confusion matrix is shown in Fig. 4 to illustrate the results.

The Bi-LSTM model, although slightly behind in terms of accuracy (81.71%), exhibited competitive results. This deep learning model captures context through its bidirectional structure, making it capable of understanding both previous and future words in the text. This feature allows Bi-LSTM to perform well in understanding the sequential nature of emotions in the text, but it falls short when compared to the simpler SVM model due to its higher computational requirements and complexity. Shown in Fig. 5

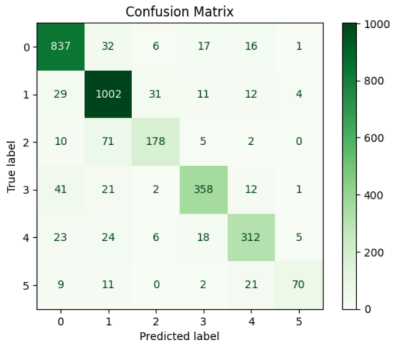

Fig. 4: Confusion Matrix for TF-IDF-SVM

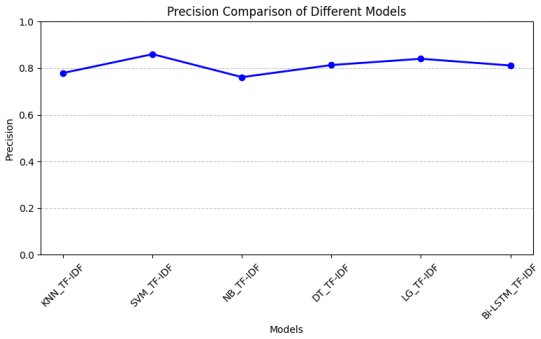

Fig. 5: Precision Comparison of Models Using TF-IDF

B. Count Vectorizer

The Count Vectorizer method creates a sparse matrix representation based solely on the frequency of words, ignoring their significance across the corpus. While it is a simpler approach compared to TF-IDF, it can still be effective when the document's vocabulary is well-distributed across emotions. The performance metrics for models using the Count Vectorizer technique are shown in Table II.

TABLE II: Performance Metrics for Models Using Count Vectorizer

Model	Accuracy	Precision	Recall	F1-Score	Error Rate
KNN	53.75%	59.33%	53.75%	52.28%	46.25%
SVM	85.56%	85.49%	85.56%	85.51%	14.44%
MNB	77.53%	78.42%	77.53%	75.07%	22.47%
DT	81.28%	81.67%	81.28%	81.27%	18.72%
LR	**86.25%**	**86.07%**	**86.25%**	**86.07%**	**13.75%**
Bi-LSTM	83.56%	83.35%	83.56%	83.05%	16.43%

This confusion matrix shown in Fig. 6 illustrating the LR model's good performance with countvectorizer.

In this case, the Logistic Regression (LR) model outperforms others with an accuracy of 86.25%, supported by its high precision and recall scores. The LR model benefits from the simplicity of the Count Vectorizer, as the model is able to use the frequency of words effectively for linear decision-making. Bi-LSTM again performs competitively with an accuracy of 83.56%, demonstrating its strength in sequential text processing. Shown in Fig. 7

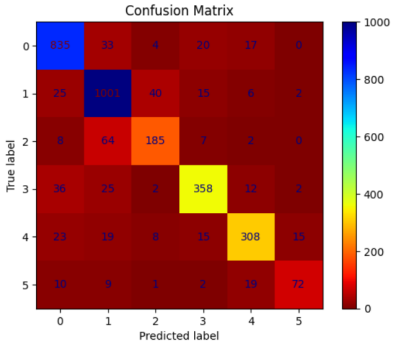

Fig. 6: Confusion Matrix for Countvectorizer-LR

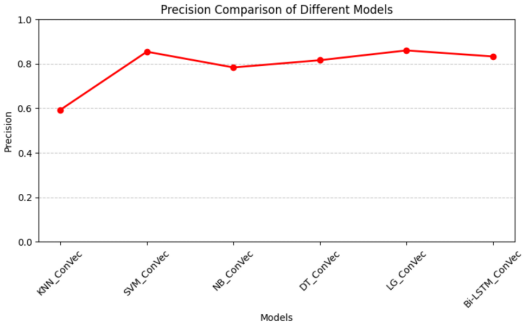

Fig. 7: Precision Comparison of Models Using Count Vectorizer

C. N-grams

The N-grams technique captures word sequences (bigrams, trigrams, etc.), offering richer contextual information compared to individual word features. This method helps in detecting multi-word phrases that are indicative of specific emotions. The performance metrics for models using N-grams are shown in Table III.

TABLE III: Performance Metrics for Models Using N-grams

Model	Accuracy	Precision	Recall	F1-Score	Error Rate
KNN	61.34%	66.14%	61.34%	60.70%	38.65%
SVM	86.06%	85.94%	86.06%	85.96%	13.93%
MNB	84.75%	84.37%	84.75%	84.39%	15.25%
DT	83.40%	83.50%	83.40%	83.41%	16.59%
LR	**88.09%**	**87.98%**	**88.09%**	**87.96%**	**11.90%**
Bi-LSTM	87.43%	87.34%	87.43%	87.09%	12.56%

This confusion matrix shown in Fig. 8 illustrating the LR model's good performance with n-gram.

The results for the N-grams technique show that Logistic Regression (LR) achieved the highest accuracy of 88.09%, followed closely by Bi-LSTM at 87.43%. The N-grams approach excels in capturing the contextual flow of words, making it particularly useful for understanding complex emotional expressions. The Bi-LSTM model benefits from its ability to capture sequential dependencies, enabling it to accurately identify emotions expressed through word combinations, which may be missed by simpler models. Shown in Fig. 9

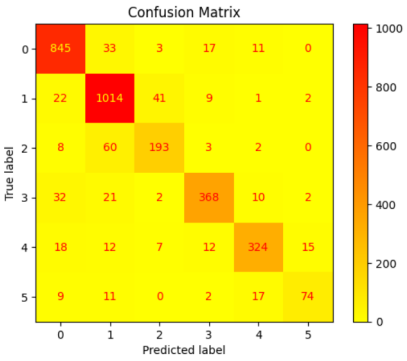

Fig. 8: Confusion Matrix for N-Gram-LR

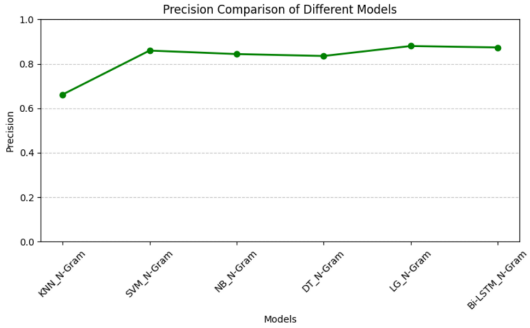

Fig. 9: Precision Comparison of Models Using N-grams

VI. Discussion

The precision of models across TF-IDF, N-grams, and Count Vectorization is compared in the following figure. SVM with TF-IDF and Logistic Regression (LR) with N-grams exhibited the highest performance. Shown in Fig. 10

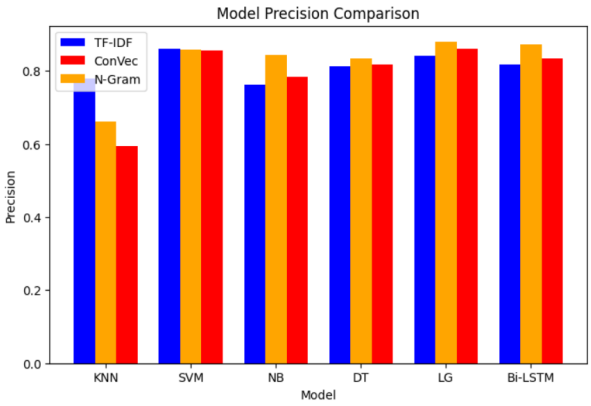

Fig. 10: Precision Comparison of All Models Across Feature Extraction Techniques

The results highlight strong performances across models and features. For TF-IDF, SVM and Decision Tree excelled in "Surprise(5)" (98.31% and 97.88%), while Logistic Regression and Bi-LSTM performed best for "Joy(1)" (94.86%) and "Sadness(0)" (89.67%). Count Vectorizer also achieved over

91% accuracy with Bi-LSTM and Logistic Regression, while N-grams enhanced Bi-LSTM and Logistic Regression results, achieving 92.63% and 93.11% for "Sadness(0)" and "Joy(1)" respectively. TF-IDF balanced precision and recall, SVM and LR leveraged relevant features, and N-grams captured context effectively. Bi-LSTM performed consistently well, understanding sequential dependencies, but simpler models like SVM and LR also delivered competitive accuracy.

TF-IDF performed exceptionally well, with SVM achieving the highest precision of 86%. This was due to its ability to identify significant words, even those with low frequency but high relevance, which SVM effectively leveraged for accurate emotion classification. N-grams excelled at capturing contextual information, enabling Logistic Regression (LR) to achieve the highest precision of 88%. By considering word pairs (or larger contexts), N-grams allowed LR to detect subtle emotional cues through contextual relationships, enhancing emotion detection compared to simpler methods. Count Vectorization, though a more straightforward approach, achieved 86% precision with LR. While it focuses on word frequencies, its inability to model relationships between words limits its performance compared to N-grams. Nevertheless, its strong results with LR demonstrate its effectiveness in emotion detection tasks.

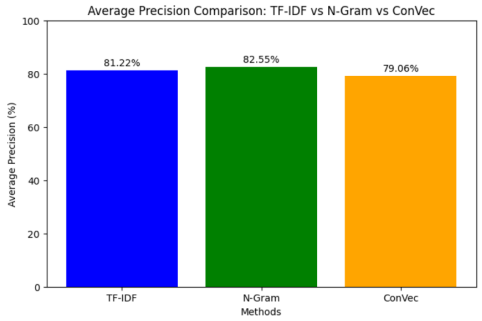

Fig. 11: Average Precision Comparison of Feature Extraction Techniques

In conclusion, N-grams shown in Fig. 11 demonstrated the most promising results in emotion detection tasks, followed closely by TF-IDF. Count Vectorization, while effective, showed slightly lower precision, reinforcing the advantage of capturing word relationships over simple frequency counts.

VII. Conclusion

This study evaluated six machine learning models for emotion detection using TF-IDF, N-grams, and Count Vectorization. Precision was prioritized due to the dataset's imbalance. The Support Vector Machine (SVM) and Logistic Regression (LR) models consistently outperformed others, with N-grams yielding the highest average precision. Our research demonstrates that leveraging Hugging Face's techniques for addressing imbalanced datasets can enhance performance, particularly by mitigating bias and improving classification for underrepresented classes.

Furthermore, the field of emotion detection continues to evolve, with a growing focus on more sophisticated approaches such as multimodal systems. These systems leverage multiple data sources—such as text, speech, and visual cues—providing a more holistic understanding of emotions and handling class imbalance. Additionally, exploring emotion detection in diverse languages, including Bengali and other regional languages, could further broaden the reach and impact of these techniques. Leveraging cross-lingual models or domain adaptation techniques could address the challenges posed by linguistic diversity, opening new avenues for more inclusive emotion detection across different cultures and contexts.

References

[1] S. Chaffar and D. Inkpen, "Using a Heterogeneous Dataset for Emotion Analysis in Text," *Advances in Artificial Intelligence*, pp. 62–67, presented at the School of Information Technology and Engineering, University of Ottawa, Ottawa, ON, Canada.

[2] A. Gupta and S. M. Srinivasan, "Constructing a Heterogeneous Training Dataset for Emotion Classification," *Procedia Computer Science*, vol. 171, pp. 1089-1098, May 2020, doi: 10.1016/j.procs.2020.02.259.

[3] P. Nandwani and R. Verma, "A review on sentiment analysis and emotion detection from text," *Social Network Analysis and Mining*, vol. 11, no. 81, pp. 1-19, Aug. 2021, doi: 10.1007/s13278-021-00776-6.

[4] S. Aman and S. Szpakowicz, "Identifying Expressions of Emotion in Text," *Text, Speech and Dialogue: Proceedings of the 10th International Conference, TSD 2007*, Pilsen, Czech Republic, Sep. 3-7, 2007, pp. 196-205, doi: 10.1007/978-3-540-74628-7.

[5] C. O. Alm, D. Roth, and R. Sproat, "Emotions from text: Machine learning for text-based emotion prediction," in *Proceedings of the HLT/EMNLP Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 579-586, Vancouver, British Columbia, Canada, 2005. [Online]. Available: https://doi.org/10.3115/1220575.1220648

[6] 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, no. 1, pp. 181–199, 2018, doi: 10.1007/s10660-017-9257-8.

[7] M. I. Alfarizi, L. Syafaah, and M. Lestandy, "Emotional Text Classification Using TF-IDF (Term Frequency-Inverse Document Frequency) And LSTM (Long Short-Term Memory)," *JUITA: Jurnal Informatika*, vol. 10, no. 2, pp. 225-232, Nov. 2022, doi: 10.30595/juita.v10i2.13262.

[8] A. F. M. N. H. Nahin, J. M. Alam, H. Mahmud, and K. Hasan, "Identifying emotion by keystroke dynamics and text pattern analysis," *Behaviour & Information Technology*, vol. 33, no. 9, pp. 987-997, Aug. 2014, doi: 10.30595/juita.v10i2.13262

[9] U. Gupta, A. Chatterjee, and P. Agrawal, "A Sentiment-and-Semantics-Based Approach for Emotion Detection in Textual Conversations," *arXiv.org*, 21 July 2017. [Online]. Available: https://arxiv.org/abs/1707.06996

[10] B. Desmet and V. Hoste, "Emotion detection in suicide notes," *Expert Systems with Applications*, vol. 40, no. 16, pp. 6347-6354, 2013, doi: 10.1016/j.eswa.2013.05.050.

[11] R. Bensoltane and T. Zaki, "Neural multi-task learning for end-to-end Arabic aspect-based sentiment analysis," Computer Speech & Language, vol. 81, p. 101683, June 2024. [Online]. Available: https://doi.org/10.1016/j.csl.2024.101683

[12] L. Ou and Z. Li, "Modeling inter-modal incongruous sentiment expressions for multi-modal sarcasm detection," Neurocomputing, vol. 616, p. 128874, Feb. 2025. [Online]. Available: https://doi.org/10.1016/j.neucom.2024.128874

[13] B. Amirshahi and S. Lahmiri, "Investigating the effectiveness of Twitter sentiment in cryptocurrency close price prediction by using deep learning," Expert Systems, Aug. 2023. [Online]. Available: https://doi.org/10.1111/exsy.13428

[14] M. S. Islam, M. N. Kabir, N. A. Ghani, K. Z. Zamli, N. S. A. Zulkifli, M. M. Rahman, and M. A. Moni, "Challenges and future in deep learning for sentiment analysis: a comprehensive review and a proposed novel hybrid approach," Artificial Intelligence Review, vol. 57, art. no. 62, Mar. 2024. [Online]. Available: https://doi.org/10.1007/s10462-024-10396-6.