Business, Management and Economics Engineering
ISSN: 2669-2481 / eISSN: 2669-249X
2024 Volume 22 Issue 01

THE SIGNIFICANCE OF REGIONAL LANGUAGE TEXT SENTIMENT ANALYSIS, EMOTION DETECTION, AND NAMED ENTITY RECOGNITION FOR MARATHI LANGUAGE TEXT: IMPROVING UNDERSTANDING AND PRECISION.

Sudarshan Sirsat
Department. of Data Science and Technology, K J Somaiya Institute of Management, Somaiaya Vidyavihar University, Mumbai, India, sudarshan@somaiya.edu

Dr. Nitish Zulpe
College of Computer Science and Information Technology, Swami Ramanand Teerth Marathwada university, Latur, India, nitishzulpe@gmail.com

Abstract- Performing regional language sentiment analysis, Marathi language emotion detection, and named entity recognition (NER) and emotional topic modeling on Marathi language text is of utmost importance for various reasons. Marathi, being one of the prominent languages spoken in India, encompasses a vast amount of textual data ranging from social media posts to news articles. The analysis of this data provides valuable insights into the collective sentiments, emotions, and significant entities that shape public opinion and discourse. Sentiment Analysis involves determining whether an input text is positive, negative or neutral as in polarity check. In the context of Marathi language text, sentiment analysis can be particularly advantageous for businesses to assess customer satisfaction, for politicians to comprehend public opinion, and for social researchers to study societal trends. The ability to automatically analyze sentiments aids in real-time monitoring and decision-making. Regional Language Emotion Detection analysis provides a general classification of text in seven different emotions defined in the language resources like dictionary, emotion detection delves deeper to identify specific emotions such as joy, anger, sadness, and surprise. For Marathi texts, this can enhance the understanding of how certain events or topics emotionally resonate with the population. For instance, media organizations can utilize emotion detection to tailor their content more effectively to their audience's emotional responses. Named Entity Recognition for Marathi language involves identifying and categorizing tokens as names of person, organizations, cities, animals and other significant terms within a text. In Marathi, NER can assist in structuring unorganized text data, making it easier to extract meaningful information. This is particularly valuable in news aggregation, automated reporting, and database management, where promptly identifying key entities can save considerable time and resources. Ensuring the precision of NLP tasks is of utmost importance. The evaluation of sentiment analysis, emotion detection, and NER models should be conducted rigorously using metrics like precision scores, recall factor and F1-score. To achieve high accuracy in Marathi text, it is crucial to create robust datasets and employ cross-validation techniques. By developing accurate NLP models in Marathi, we can bridge the digital divide and extend the benefits of technology to non-English-speaking populations, promoting inclusivity. To performing sentiment analysis, emotion detection, and NER on Marathi text allows for better extraction, understanding, and utilization of the extensive and valuable textual data available in this language. Accurate NLP models in Marathi not only enhance user engagement and satisfaction but also facilitate informed decision-making across various sectors.

Keywords—Natural Language Processing, Marathi Language Dataset, Regional Language Sentiment Analysis, Regional Language Named Entity Recognition, Regional Language Emotion Detection, Sentiment Analysis, Web Scrapping

I. INTRODUCTION

Natural Language Processing (NLP) has brought about a significant transformation in our interaction with written information. It has empowered us to derive meaningful insights and automate tasks that were once time-consuming. Within the realm of NLP, one crucial aspect is the analysis of sentiments, emotions, and named entities in text. This analysis provides vital information across various domains like business, politics, and social sciences. While there has been considerable focus on English language text, it is imperative to emphasize the need for extending NLP techniques to other languages, including Marathi.

Marathi, being one of the prominent languages spoken in India, possesses a vast collection of textual data encompassing a wide range of genres, from social media posts to news articles. Analysing Marathi text presents both unique challenges and opportunities due to its distinct linguistic characteristics and cultural nuances. By performing regional language sentiment analysis, emotion identification, and named entity recognition (NER) on Marathi text, we can unlock valuable insights into the collective sentiments, emotional responses, and significant entities that shape public discourse and decision-making processes.

A. Regional language Sentiment Analysis:

Sentiment analysis plays an important stake in analysing the polarity of text, categorizing it as positive, negative, or neutral. In the case of Marathi text, sentiment analysis holds great potential for various stakeholders. Businesses can utilize sentiment analysis to assess customer satisfaction, monitor brand reputation, and adapt their marketing strategies accordingly. For politicians and policymakers, understanding public sentiment in Marathi text can inform their decision-making processes and help them establish stronger connections with constituents. Moreover, social researchers can employ sentiment analysis to examine societal trends and public opinion dynamics within Marathi-speaking communities. The ability to automatically analyse sentiments in Marathi text enables real-time monitoring and facilitates timely interventions in response to emerging trends or issues.

B. Regional language Emotion Detection:

While SA provides a general classification of text, emotion classification and identification goes further by identifying specific emotions such as joy, anger, sadness, and surprise. In the context of Marathi text, emotion detection adds a layer of detail to our understanding of how individuals emotionally respond to different events or topics. Media organizations can leverage emotion detection to personalize content and effectively resonate with their Marathi-speaking audience. By analysing emotional responses in Marathi text, content creators can customize their messaging to evoke desired emotional reactions and enhance audience engagement.

C. Regional Language Named Entity Recognition (NER):

Is a crucial task that involves the identification and categorization of tokens into animals, cities, people, organizations etc. In the context of Marathi language, NER plays a vital role in organizing unstructured textual data and extracting meaningful information from it. This is particularly beneficial for news aggregators and automated reporting systems, as NER helps efficiently identify and categorize relevant entities, streamlining the process of information retrieval and analysis. Additionally, NER facilitates effective database management by enabling rapid indexing and retrieval of information based on these key entities. In domains like healthcare and finance, accurate NER in Marathi text can greatly contribute to improving information retrieval systems and enhancing decision support mechanisms.

D. Evaluation and challenges:

The evaluation of NLP tasks, including sentiment analysis, emotion detection, and NER in Marathi text, is of utmost importance to ensure precision and reliability. Rigorous evaluation using measuring matrices like precision, recall factor, and F1-score is essential for accurately assessing the performance of NLP models. However, there are several challenges that need to be addressed. These challenges include data scarcity, linguistic variations, and domain-specific terminology, which pose significant obstacles in developing robust NLP models for Marathi text. High quality datasets can help improving on the overall performance of all the models.

Performing sentiment analysis, emotion detection, and NER on Marathi text reveals the immense possibilities of the textual data present in this language. Precise NLP models in Marathi not only improve user interaction and contentment but also enable well-informed decision-making in different industries. By bridging the gap between digital resources and non-English-speaking communities, we can foster inclusivity and empower diverse societies to leverage the potential of NLP for social, economic, and cultural progress.

II. LITERATURE REVIEW

The study, explores the intricacies of sentiment analysis on mixed code texts, specifically focusing on transliterated Hindi and Marathi. Mixed code texts, which incorporate multiple languages or scripts within the same text, are becoming more common in social media and informal digital communications. This poses significant challenges for NLP due to the absence of standardized spelling and the blending of languages. The paper seeks to tackle these challenges by developing a sentiment analysis system tailored for mixed code transliterated texts. With the proliferation of social media platforms, mixed code and transliterated texts have become widespread. Transliterated texts involve content where languages like Hindi and Marathi are written using the Roman alphabet instead of their native Devanagari script. This hybrid nature complicates NLP tasks, as current systems are mainly designed for monolingual or single-script inputs. The objective of this study is to establish an efficient models for analysing sentiments in such intricate text forms, commonly used in informal communication. The main goals of the study include: Creating a sentiment analysis system capable of handling mixed code texts in transliterated Hindi and Marathi. Assessing the performance of different machine learning algorithms in this particular context. To accomplish these objectives, the authors initially compiled a dataset consisting of mixed code transliterated Hindi and Marathi texts gathered from social media platforms. The pre-processing of this dataset involved several steps to ensure that the text was suitable for analysis: Normalization: Standardizing the text to address transliteration variations. Tokenization: Breaking down the text into individual tokens or words. After pre-processing, the authors utilized various machine learning algorithms to categorize the sentiments conveyed in the texts. The mythologies explored by the scholars were Naive Bayes, SVM and Decision Trees, a model that makes decisions based on a tree-like structure of rules. Precision, recall and F1-scores were tested for this methodology. These metrics offer a comprehensive evaluation of the algorithms classify sentiments. SVM emerged as the most effective among the tested algorithms, achieving the highest accuracy in sentiment classification. Naive Bayes also displayed promising results, suggesting its potential usefulness in similar applications. The study concludes that sentiment analysis on mixed code transliterated texts using machine learning techniques is feasible. The results indicate that the developed system can accurately categorize sentiments in complex text formats, offering valuable insights for NLP applications in multilingual and transliterated contexts. The proposed approach can be customized for other languages and mixed code scenarios, enhancing the resilience and applicability of sentiment analysis systems [1].

The research study delves into the utilization of supervised learning algorithms for the classification of Marathi language documents. The authors highlight the difficulties associated with processing and organizing text in Marathi, a language that has not received as much attention in the realms of NLP and text classification. The increasing volume of digital data underscores the need for efficient text classification systems. While significant research has been carried out on text classification in major languages like English, regional languages such as Marathi have not been thoroughly explored. The experiment bridge this gap by assessing the effectiveness of various supervised learning techniques on Marathi text documents. The objective is to implement and compare diverse supervised learning algorithms for the classification of Marathi documents, in order to identify the most efficient algorithm based on accuracy and performance metrics. A dataset comprising Marathi documents was curated from multiple sources, including news articles, blogs, and online forums. TF-IDF was employed to transform the text data into numerical features suitable for machine learning. Various supervised learning algorithms were applied, such as Naive Bayes, Support Vector Machines SVM and k-Nearest Neighbours k-NN. The study revealed notable disparities in the performance of various algorithms. Naive Bayes, renowned for its simplicity and effectiveness in text classification, exhibited satisfactory results but was surpassed by SVM in terms of accuracy. SVM demonstrated the highest level of accuracy among the methods tested, indicating its suitability for Marathi text classification. On the other hand, k-NN, although easy to implement, exhibited lower accuracy compared to Naive Bayes and SVM. These outcomes shows the significance of selecting the appropriate algorithm based on the specific characteristics of the dataset and language. The study emphasizes the necessity for more comprehensive datasets and advanced NLP tools for Marathi in order to improve on the accuracy and reliability of classification models. The research showcases the effectiveness of supervised learning methods in classifying Marathi text documents, with SVM emerging as the most accurate algorithm. Future endeavours could concentrate on expanding the dataset, incorporating more sophisticated pre-processing techniques, and exploring additional algorithms and ensemble methods. The research establishes the foundation for further research and the development of text classification systems for Marathi, which can be extended to other underrepresented languages [2].

The research by Ratna Nitin Patil et al. explores the utilization of deep learning techniques to improve sentiment classification on restaurant reviews. In the digital era we live in, online reviews hold significant influence over consumer decisions, particularly in the realm of restaurants. Hence, accurately analysing the sentiment conveyed in these reviews is essential for businesses to comprehend customer perceptions and enhance their services accordingly. The authors commence by emphasizing the importance of sentiment analysis in the restaurant industry and the challenges associated with conventional methods, which often rely on manually crafted features and shallow learning algorithms. They argue that deep learning models have exhibited promising outcomes in various natural language processing tasks, and propose their application to sentiment classification in restaurant reviews. To assess the effectiveness of deep learning models for sentiment analysis, the authors conduct experiments employing a dataset consisting of restaurant reviews. They pre-process the data through tokenization, elimination of stopwords, and conversion of text to lowercase to standardize the input. Additionally, they employ techniques such as word embedding to represent words in a vector space, which captures semantic relationships between words and enhances the performance of deep learning models. The authors experiment with different deep learning architectures, including CNNs, LSTMs, and combinations thereof, to compare their performance in sentiment classification tasks. CNNs are renowned for their ability to capture local patterns in data, making them suitable for text classification tasks. Conversely, LSTMs excel in capturing long-range dependencies in sequential data, which is crucial for comprehending the context in natural language processing tasks. The authors' experiments reveal that combining CNN and LSTM architectures yields better results in sentiment classification for restaurant reviews compared to using individual models. This hybrid model effectively captures both local patterns and long-range dependencies in the text data, leveraging the strengths of both CNNs and LSTMs. Additionally, the authors investigate the impact of hyper parameters checks on the performance of deep learning models. Furthermore, the paper discusses the significance of feature selection and dimensionality reduction techniques in improving the efficiency of deep learning models, especially when dealing with high-dimensional text data. Techniques like Principal Component Analysis PCA and t-Distributed Stochastic Neighbour Embedding t-SNE are employed to visualize the learned representations and gain insights into the distribution of the data. This study demonstrates the effectiveness of deep learning models, particularly the hybrid CNN-LSTM architecture, in sentiment classification for restaurant reviews. By utilizing these models, businesses can extract valuable insights from online reviews to enhance customer satisfaction and improve their services. The authors also highlight the importance of feature engineering and hyper parameter tuning in optimizing the performance [3].

The study focuses on the crucial issue of code smell detection in software development, with the goal of improving on the ML classifiers' performance through swarm intelligence algorithms. Code smells serve as warning signs of potential problems or inefficiencies in software code, making early detection essential for maintaining code quality and ensuring software reliability and maintainability. The authors stress the significance of code smell detection in software engineering, underscoring its role in pinpointing problematic areas within codebases. Traditional methods of code smell detection often involve manual inspection or static analysis techniques, which can be lengthy and prone to errors, especially in large-scale software projects. Machine learning methods present a promising solution by automating the detection process, but they require efficient optimization for optimal performance. To tackle this challenge, Jain and Saha suggest utilizing swarm intelligence algorithms to optimize machine learning classifiers for code smell detection. Swarm intelligence algorithms. By harnessing swarm intelligence principles, the authors aim to boost machine learning classifiers' performance and enhance their accuracy in identifying code smells. The research evaluates various machine learning classifiers, such as Support Vector Machines (SVM), Random Forest, k-Nearest Neighbours (k-NN), and Naive Bayes, optimized using three different swarm intelligence algorithms: Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Genetic Algorithm (GA). These classifiers are trained and tested on a dataset containing code metrics extracted from software projects, with labelled instances indicating the presence or absence of code smells. The results of the experiment show that machine learning classifiers optimized with swarm intelligence outperform their non-optimized counterparts in accuracy, precision, recall, and F1-score for detecting code smells. PSO stands out as the best performing swarm intelligence algorithm, consistently achieving higher accuracy and surpassing both ACO and GA across various classifiers. Additionally, the study compares the optimized machine learning classifiers with traditional static analysis techniques for code smell detection. The findings indicate that swarm intelligence-optimized classifiers outperform static analysis methods in accuracy and efficiency, demonstrating the effectiveness of the proposed approach in automating code smell detection and enhancing software quality. In summary, this research contributes to the software engineering field by introducing a new method for code smell detection using swarm intelligence-optimized machine learning classifiers. By leveraging swarm intelligence algorithms, this approach delivers superior performance compared to conventional methods, providing developers with a valuable tool to improve code quality and maintainability in software projects [4].

This research utilizes machine learning models and techniques to address the complexities of sentiment analysis in Turkish, a language renowned for its intricate morphological structure. The research employs a human annotated dataset consisting of 17,793 Turkish tweets pertaining to Turkish universities. This dataset plays a crucial role in training and validating the machine learning models. The primary contribution lies in the development of a hybrid architecture called BERT-BiLSTM-CNN. This architecture combines BERT, BiLSTM and CNN to enhance the performance of sentiment analysis. The BERT component handles contextual understanding, BiLSTM captures sequential dependencies, and CNNs are utilized for feature extraction. The proposed model achieves remarkable results, surpassing state-of-the-art models in sentiment analysis, with an accuracy rate exceeding 91%, an F1 score of 88%, and a Receiver Operating Characteristic (ROC) of 96.32%. These metrics serve as evidence of the model's effectiveness in dealing with the linguistic intricacies of Turkish. The paper also presents a comparative analysis of various conventional machine learning classifiers (such as K-Nearest Neighbors, Naive Bayes, and Support Vector Machine) and deep learning-based models (such as LSTM and CNN). Consistently, the BERT-BiLSTM-CNN model outperforms these models, highlighting the advantages of the hybrid approach. By analyzing tweets, this study provides valuable insights into the public's perception of Turkish higher education. Educational institutions can utilize this information to enhance their services and engagement with the community. Furthermore, the methodology demonstrates the potential of social media analytics in evaluating public satisfaction across different domains [5].

This study makes a significant contribution to sentiment analysis, especially for low-resource languages like Tamil. The authors present a unique multimodal representation learning model known as Shared-Private Multimodal AutoEncoder (SPMMAE). This model is designed to improve sentiment analysis by using shared-private projections of unimodal features, offering a comprehensive view of multimodal data. A new dataset, the Multimodal Sentiment Analysis corpus in Tamil (MSAT), has been developed specifically for research in low-resource languages, addressing the lack of resources for such languages. The SPMMAE model achieves state-of-the-art results on well-known datasets such as CMU-MOSI, CMU-MOSEI, and MELD, which are widely used benchmarks in multimodal sentiment analysis (MSA) and emotion recognition in conversation (ERC). The study demonstrates that cross-lingual transfer learning in a multimodal setting significantly enhances the performance of the MSAT dataset by 11%, showcasing the potential of cross-lingual techniques in improving sentiment analysis for low-resource languages. This research establishes benchmark results on the MSAT dataset, providing a valuable resource for future research. These benchmarks will facilitate the comparison and development of new models and methodologies in the field. This study introduces an innovative multimodal representation learning architecture tailored for sentiment analysis, particularly in low-resource languages. The creation of the MSAT dataset and the application of cross-lingual transfer learning significantly enhance the capabilities of sentiment analysis models, offering new benchmarks and improving the understanding of multimodal data in various linguistic contexts. The impressive performance of the SPMMAE model on multiple datasets highlights its strength and effectiveness [6].

III. METHODOLOGY

Steps followed for the measuring performance of the traditional machine learning programs and models are as shown in the below natural language algorithm:

A. Overview of the approach followed for the performance and evaluation checking

1. Importing libraries
2. Load the excel file
3. Check if DataFrame is empty
4. Translating the text and Calculating sentiment
5. Evaluating the accuracy on testing and training set
6. Generate a classification report and save results to new excel file
7. Apply step 1-6 for the new excel report
8. Generate classification report
9. Plot the confusion matrix (using seaborn and matplotlib)
10. Print results and confusion matrix

B. The Detailed approach followed for the performance and evaluation checking

1. Load Data:

- Use the pandas library to read the Excel file (`news.xlsx`) into a DataFrame.

2. Sentiment Analysis:

- Begin with the sentiment analysis
- Iterate over each row of the DataFrame, translate Marathi text to English using the deep_translator library, and calculate sentiment using the initialized pipeline.

- Add the predicted sentiment labels to the DataFrame.
- Save the DataFrame to a new Excel file (`sentiment_analysis_results.xlsx`).

3. Evaluation:

- Load the results DataFrame (`sentiment_analysis_results.xlsx`).
- categorise the data into training and testing sets.
- Train a machine learning model (e.g., logistic regression, random forest) using the training set.
- Classification report
- Generate a classification report and confusion matrix using scikit-learn.

4. Visualization:

- Use matplotlib and seaborn libraries to plot and save the heatmap for the confusion matrix.

5. Technology:

- Utilize Python programming language.
- Employ libraries such as pandas, transformers, deep_translator, scikit-learn, matplotlib, and seaborn.

6. Functionality:

- Data loading: pandas for loading and preprocessing.
- Sentiment analysis: transformers for sentiment analysis with pre-trained models, and deep_translator for translation.
- Evaluation metrics: scikit-learn for accuracy
- Visualization: matplotlib and seaborn for plotting confusion matrix and heatmap.

7. Data Sources:

- Input: `news.xlsx` containing Marathi text and sentiment labels.
- Output: `sentiment_analysis_results.xlsx` containing Marathi text, translated text, sentiment labels, and model predictions.

8. File Formats:

Input: Excel format (`news.xlsx`).
Output: `sentiment_analysis_results.xlsx`

9. Algorithm Flow:

- Load data from `news.xlsx`.
- Perform sentiment analysis and translation.
- Save results to `sentiment_analysis_results.xlsx`.
- Evaluate model performance.
- Visualize evaluation results with a confusion matrix and heat map.

10. Execution:

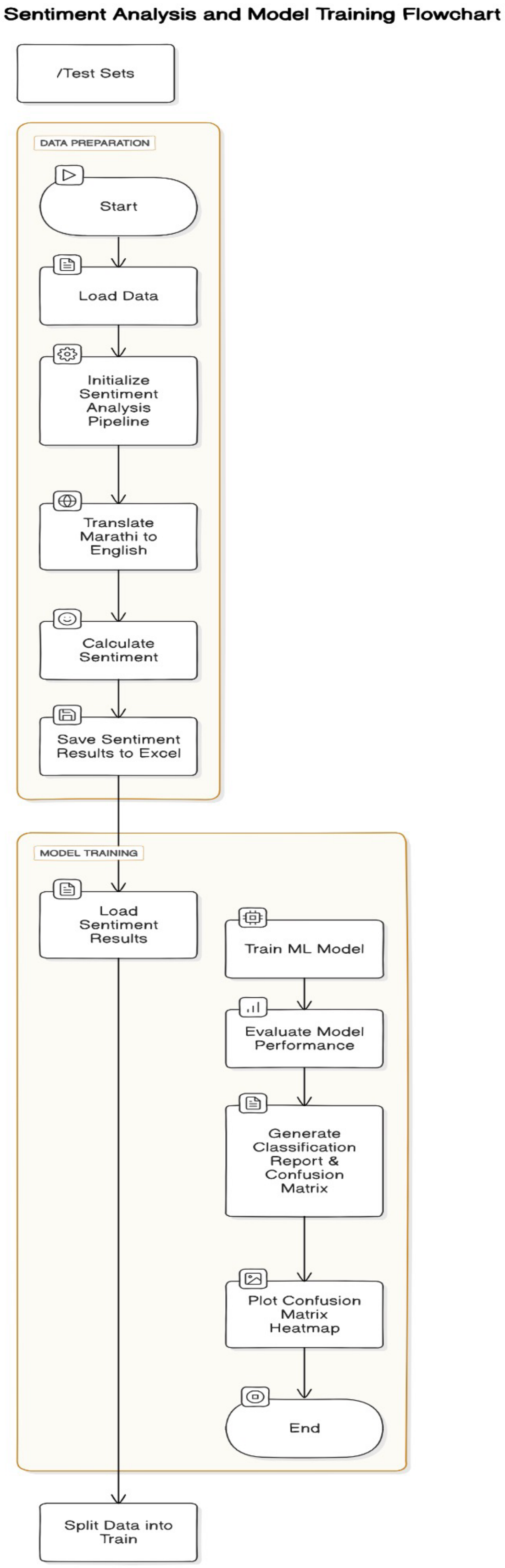
- Implement the algorithm in a Python script or a Google Colab notebook.
- Execute the script to perform sentiment analysis, evaluation, and visualization.

Figure 1 represents and demonstrates the detailed algorithmic approach for the sentiment analysis methodology for regional language of the state i.e. Marathi language dataset.

Furthermore, a detailed classification report and confusion matrix are generated using scikit-learn. The confusion matrix is visualized as a heatmap using the matplotlib and seaborn libraries. This structured approach to handling sentiment analysis and subsequent model evaluation and visualization tasks is clearly illustrated through the flowchart, with arrows indicating the flow of the process and a clear segmentation into data preparation and model training stages.

Furthermore, a detailed classification report and confusion matrix are generated using scikit-learn. The confusion matrix is visualized as a heatmap using the matplotlib and seaborn libraries. This structured approach to handling sentiment analysis and subsequent model evaluation and visualization tasks is clearly illustrated through the flowchart, with arrows indicating the flow of the process and a clear segmentation into data preparation and model training stages.

Furthermore, a detailed classification report and confusion matrix are generated using scikit-learn. The confusion matrix is visualized as a heatmap using the matplotlib and seaborn libraries. This structured approach to handling sentiment analysis and subsequent model evaluation and visualization tasks is clearly illustrated through the flowchart, with arrows indicating the flow of the process and a clear segmentation into data preparation and model training stages.


Sentiment Analysis and Model Training Flowchart
/Test Sets
DATA PREPARATION
Start
Load Data
Initialize Sentiment Analysis Pipeline
Translate Marathi to English
Calculate Sentiment
Save Sentiment Results to Excel
MODEL TRAINING
Load Sentiment Results
Train ML Model
Evaluate Model Performance
Generate Classification Report & Confusion Matrix
Plot Confusion Matrix Heatmap
End
Split Data into Train

Fig, 1. Detailed approach followed for measuring the accuracy and performance of the regional language sentiment analysis

IV. RESULTS AND DISCUSSION

Two independent experiments were conducted on two different types of datasets, first one was performed on web scraped data and another one was conducted on the government official website dataset.

A. Model Assessment for web scrapped Marathi language text dataset

Figure 2 depicts the detailed classification report for the Marathi language web scrapped dataset and data items the below analysis is done on the basic of Figure 2.

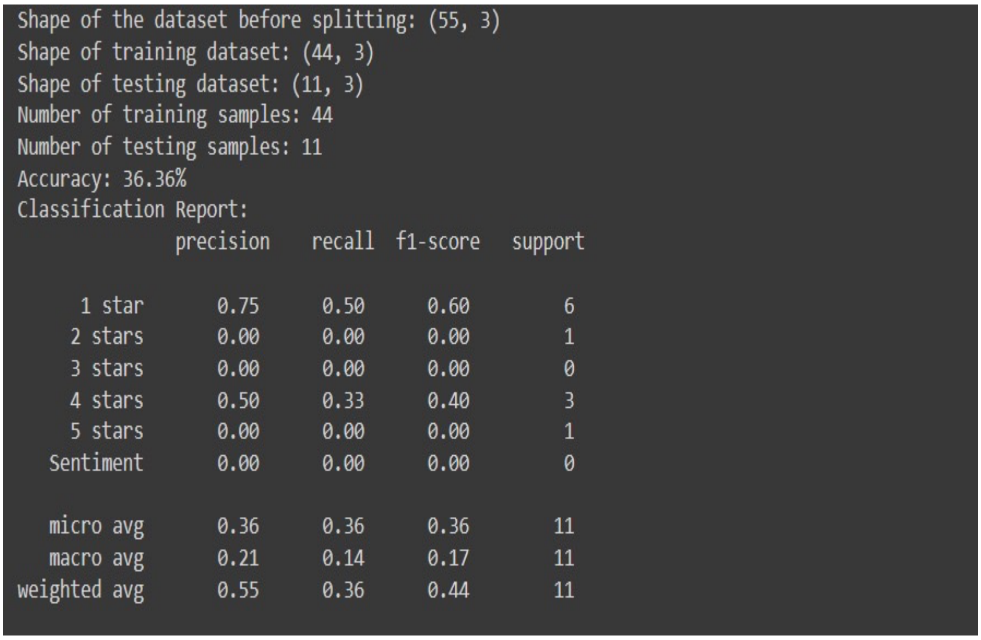
Data Division:

Initial dataset shape: The dataset comprises 55 samples and 3 features.
Training dataset shape: 44 samples are allocated for training purposes.
Testing dataset shape: 11 samples are designated for testing.
Number of training instances: Confirms the presence of 44 training samples.
Number of testing instances: Confirms the presence of 11 testing samples.

Model Accuracy:

The model has achieved an accuracy rate of 36.36%.
Classification Summary: scores ranging from 1 star to 5 stars).
Precision: Reflects the proportion of accurately correct predicted instances against the total predicted positives.
1 star: 0.75
2 stars, 3 stars, 5 stars, Sentiment: 0.00
4 stars: 0.50
Recall: Indicates the ratio of correctly predicted positive instances to all instances in the actual category.
1 star: 0.50
2 stars, 3 stars, 5 stars, Sentiment: 0.00
4 stars: 0.33
F1-score: Represents the weighted average of precision and recall.
1 star: 0.60
2 stars, 3 stars, 5 stars, Sentiment: 0.00
4 stars: 0.40
Support: Denotes the actual frequency of occurrences of the category in the dataset.
1 star: 6
2 stars: 1
3 stars: 0
4 stars: 3
5 stars: 1
Sentiment: 0
Overall Metrics:
Micro average: 0.36 for precision, recall, and f1-score.
Macro average: 0.21 for precision, 0.18 for recall, 0.17 for f1-score.
Weighted average: 0.55 for precision, 0.36 for recall, 0.44 for f1-score.

These results suggest that the model's performance is relatively subpar, showing better accuracy in predicting "1 star" ratings but displaying weaker performance in other categories, especially those with limited data samples. This could be attributed to dataset imbalances or insufficient data for certain categories.

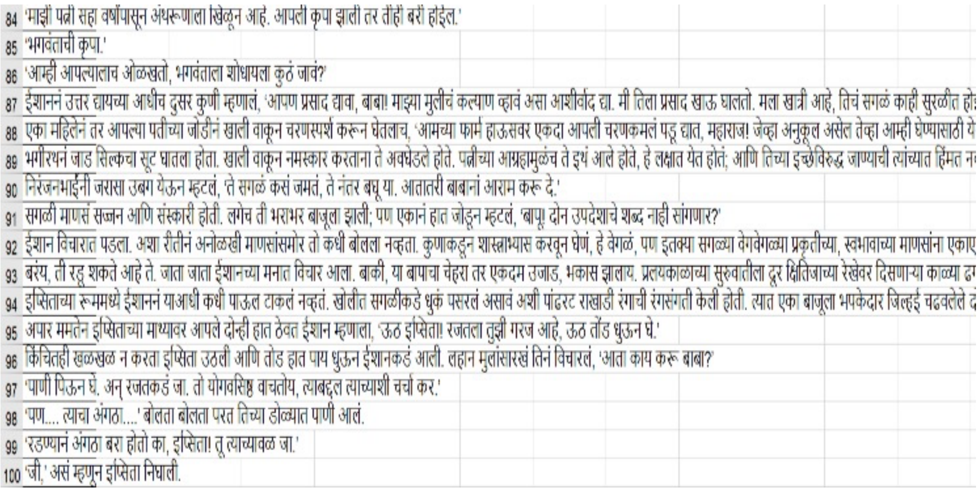


```
Shape of the dataset before splitting: (55, 3)
Shape of training dataset: (44, 3)
Shape of testing dataset: (11, 3)
Number of training samples: 44
Number of testing samples: 11
Accuracy: 36.36%
Classification Report:
              precision    recall  f1-score   support

      1 star       0.75      0.50      0.60         6
     2 stars       0.00      0.00      0.00         1
     3 stars       0.00      0.00      0.00         0
     4 stars       0.50      0.33      0.40         3
     5 stars       0.00      0.00      0.00         1
   Sentiment       0.00      0.00      0.00         0

   micro avg       0.36      0.36      0.36        11
   macro avg       0.21      0.14      0.17        11
weighted avg       0.55      0.36      0.44        11
```

Fig. 2. Classification report of the models functioning on web scrapped Marathi language text dataset

B. Model Assessment for web scrapped Marathi language text dataset

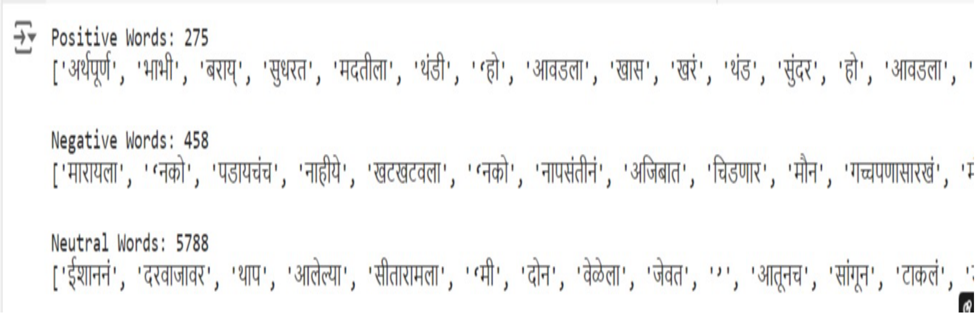
We have used the Marathi language text as dataset from the government official website of Central Institute of Indian Languages (CIIL) and its department Linguistic Data Consortium for Indian Languages (LDC-IL) text corpora for the authentic data. The data contains 268 sample size which contains various emotion based Marathi Language text items which we utilized for polarity check, sentiment analysis, emotion detection and Named entity recognition.



84 'माझी पत्नी सहा वर्षांपासून अंथरूणाला खिळून आहे. आपली कृपा झाली तर तीही बरी होईल.'
85 'भगवंताची कृपा.'
86 'आम्ही आपल्यालाच ओळखतो, भगवंताला शोधायला कुठं जावं?'
87 ईशाननं उत्तर द्यायच्या आधीच दुसर कुणी म्हणालं, 'आपण प्रसाद द्यावा, बाबा! माझ्या मुलीचं कल्याण व्हावं असा आशीर्वाद द्या. मी तिला प्रसाद खाऊ घालतो. मला खात्री आहे, तिचं सगळं काही सुरळीत हो
88 एका महिलेनं तर आपल्या पतीच्या जोडीनं खाली वाकून चरणस्पर्श करून घेतलाच, 'आमच्या फार्म हाऊसवर एकदा आपली चरणकमलं पडू द्यात, महाराज! जेव्हा अनुकूल असेल तेव्हा आम्ही घेण्यासाठी ये
89 भगीरथनं जाड सिल्कचा सूट घातला होता. खाली वाकून नमस्कार करताना ते अवघडले होते. पत्नीच्या आग्रहामुळंच ते इथं आले होते, हे लक्षात येत होतं; आणि तिच्या इच्छेविरुद्ध जाण्याची त्यांच्यात हिंमत न
90 निरंजनभाईंनी जरासा उबग येऊन म्हटलं, 'ते सगळं कसं जमतं, ते नंतर बघू या. आतातरी बाबानां आराम करू दे.'
91 सगळी माणसं सज्जन आणि संस्कारी होती. लगेच ती भराभर बाजूला झाली; पण एकानं हात जोडून म्हटलं, 'बापू! दोन उपदेशाचे शब्द नाही सांगणार?'
92 ईशान विचारात पडला. अशा रीतीनं अनोळखी माणसांसमोर तो कधी बोलला नव्हता. कुणाकडून शास्त्राभ्यास करवून घेणं, हे वेगळं, पण इतक्या सगळ्या वेगवेगळ्या प्रकृतीच्या, स्वभावाच्या माणसांना एका
93 बरंय, ती रडु शकते आहे ते. जाता जाता ईशानच्या मनात विचार आला. बाकी, या बापाचा चेहरा तर एकदम उजाड, भकास झालाय. प्रलयकाळाच्या सुरुवातीला दूर क्षितिजाच्या रेखेवर दिसणाऱ्या काळ्या ढ
94 इप्सिताच्या रूममध्ये ईशाननं याआधी कधी पाऊल टाकलं नव्हतं. खोलीत सगळीकडे धुकं पसरलं असावं अशी पांढरट राखाडी रंगाची रंगसंगती केली होती. त्यात एका बाजूला भपकेदार जिल्हई चढवलेले द
95 अपार ममतेनं इप्सिताच्या माथ्यावर आपले दोन्ही हात ठेवत ईशान म्हणाला, 'ऊठ इप्सिता! रजतला तुझी गरज आहे, ऊठ तोंड धुऊन घे.'
96 किंचितही खळखळ न करता इप्सिता उठली आणि तोंड हात पाय धुऊन ईशानकडं आली. लहान मुलांसारखं तिनं विचारलं, 'आता काय करू बाबा?'
97 'पाणी पिऊन घे. अन् रजतकडं जा. तो योगवसिष्ठ वाचतोय, त्याबद्दल त्याच्याशी चर्चा कर.'
98 'पण.... त्याचा अंगठा....' बोलता बोलता परत तिच्या डोळ्यात पाणी आलं.
99 'रडण्यानं अंगठा बरा होतो का, इप्सिता! तू त्याच्यावळ जा.'
100 'जी,' असं म्हणून इप्सिता निघाली.

Fig. 3. Sample data items from authentic CIIL Marathi language dataset

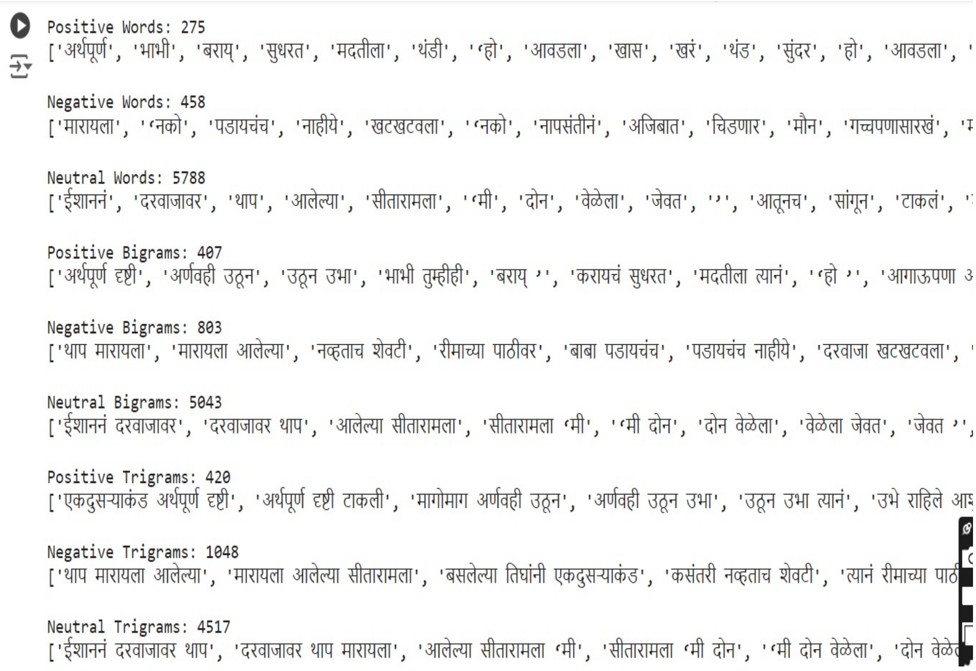
The dataset shown in the figure 3 is stored in the .xlsx format, the data items were initially downloaded in the Microsoft word document format from the government website.



```
Positive Words: 275
['अर्थपूर्ण', 'भाभी', 'बराय्', 'सुधरत', 'मदतीला', 'थंडी', '‘हो', 'आवडला', 'खास', 'खरं', 'थंड', 'सुंदर', 'हो', 'आवडला', '

Negative Words: 458
['मारायला', '‘नको', 'पडायचंच', 'नाहीये', 'खटखटवला', '‘नको', 'नापसंतीनं', 'अजिबात', 'चिडणार', 'मौन', 'गच्चपणासारखं', '

Neutral Words: 5788
['ईशाननं', 'दरवाजावर', 'थाप', 'आलेल्या', 'सीतारामला', '‘मी', 'दोन', 'वेळेला', 'जेवत', '’', 'आतूनच', 'सांगून', 'टाकलं', '
```

Fig. 4. Identifying the positive, negative and neutral tokens from the authentic Marathi language dataset

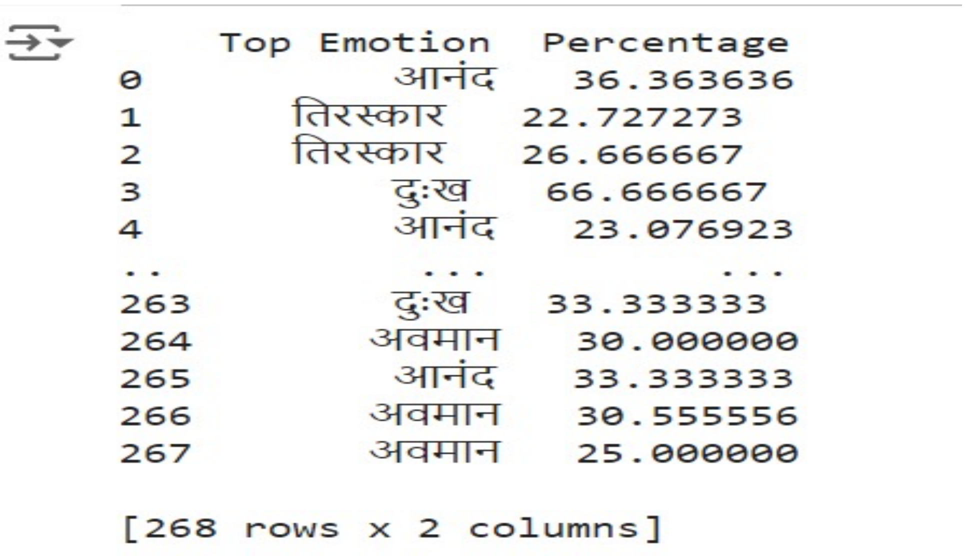
Various positive negative and neutral words are identified for the polarity analysis as represented in the Figure 4 above.



```
Positive Words: 275
['अर्थपूर्ण', 'भाभी', 'बराय्', 'सुधरत', 'मदतीला', 'थंडी', '‘हो', 'आवडला', 'खास', 'खरं', 'थंड', 'सुंदर', 'हो', 'आवडला', '

Negative Words: 458
['मारायला', '‘नको', 'पडायचंच', 'नाहीये', 'खटखटवला', '‘नको', 'नापसंतीनं', 'अजिबात', 'चिडणार', 'मौन', 'गच्चपणासारखं', '

Neutral Words: 5788
['ईशाननं', 'दरवाजावर', 'थाप', 'आलेल्या', 'सीतारामला', '‘मी', 'दोन', 'वेळेला', 'जेवत', '’', 'आतूनच', 'सांगून', 'टाकलं', '

Positive Bigrams: 407
['अर्थपूर्ण दृष्टी', 'अर्णवही उठून', 'उठून उभा', 'भाभी तुम्हीही', 'बराय् ’', 'करायचं सुधरत', 'मदतीला त्यानं', '‘हो ’', 'आगाऊपणा

Negative Bigrams: 803
['थाप मारायला', 'मारायला आलेल्या', 'नव्हताच शेवटी', 'रीमाच्या पाठीवर', 'बाबा पडायचंच', 'पडायचंच नाहीये', 'दरवाजा खटखटवला', '

Neutral Bigrams: 5043
['ईशाननं दरवाजावर', 'दरवाजावर थाप', 'आलेल्या सीतारामला', 'सीतारामला ‘मी', '‘मी दोन', 'दोन वेळेला', 'वेळेला जेवत', 'जेवत ’',

Positive Trigrams: 420
['एकदुसऱ्याकंड अर्थपूर्ण दृष्टी', 'अर्थपूर्ण दृष्टी टाकली', 'मागोमाग अर्णवही उठून', 'अर्णवही उठून उभा', 'उठून उभा त्यानं', 'उभे राहिले आ

Negative Trigrams: 1048
['थाप मारायला आलेल्या', 'मारायला आलेल्या सीतारामला', 'बसलेल्या तिघांनी एकदुसऱ्याकंड', 'कसंतरी नव्हताच शेवटी', 'त्यानं रीमाच्या पाठ

Neutral Trigrams: 4517
['ईशाननं दरवाजावर थाप', 'दरवाजावर थाप मारायला', 'आलेल्या सीतारामला ‘मी', 'सीतारामला ‘मी दोन', '‘मी दोन वेळेला', 'दोन वेळे
```

Fig. 4. Classification of the bigrams and trigrams from dataset into positive, negative and neutral polarity

Bigrams trigrams are identified and detected with proper classification techniques where in most frequently used words in pairs of two, three and all possible frequency are identified as represented in Figure 4.



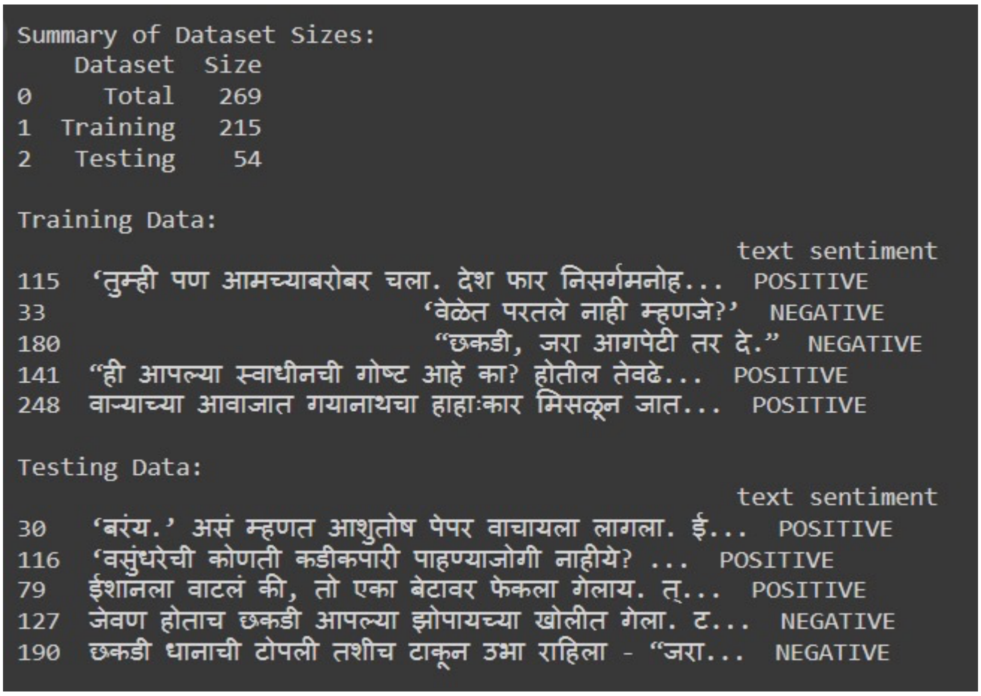
```
    Top Emotion  Percentage
0          आनंद   36.363636
1      तिरस्कार   22.727273
2      तिरस्कार   26.666667
3          दुःख   66.666667
4          आनंद   23.076923
..          ...         ...
263        दुःख   33.333333
264      अवमान   30.000000
265        आनंद   33.333333
266      अवमान   30.555556
267      अवमान   25.000000

[268 rows x 2 columns]
```

Fig. 5. Independent prominent emotion detection for each data item

Prominent emotions from the each data items are identified and highlighted as represented in Figure 5. Out of 7 present emotions the scores for all emotions were calculated and top most scored emotion is identified as the top emotion for that specific data item. The percentages for the same is also highlighted in the figure 5.

Sentiment analysis and polarity check for the authentic Marathi language dataset was done and overall was it a positive or negative is highlighted as ‘text sentiment’. Bipolar classification was done as positive and negative as represented in the Figure 6.

Data Loading and Initial Setup: The initial dataset was loaded from an Excel file named "sentiment_analysis_results(new_dataset4).xlsx". This dataset consisted of two columns, namely 'text' and 'sentiment'.



```
Summary of Dataset Sizes:
    Dataset  Size
0     Total   269
1  Training   215
2   Testing    54

Training Data:
                                                         text sentiment
115  ‘तुम्ही पण आमच्याबरोबर चला. देश फार निसर्गमनोह...  POSITIVE
33                           ‘वेळेत परतले नाही म्हणजे?’  NEGATIVE
180                          “छकडी, जरा आगपेटी तर दे.”  NEGATIVE
141  “ही आपल्या स्वाधीनची गोष्ट आहे का? होतील तेवढे...  POSITIVE
248  वाऱ्याच्या आवाजात गयानाथचा हाहाःकार मिसळून जात...  POSITIVE

Testing Data:
                                                         text sentiment
30   ‘बरंय.’ असं म्हणत आशुतोष पेपर वाचायला लागला. ई...  POSITIVE
116  ‘वसुंधरेची कोणती कडीकपारी पाहण्याजोगी नाहीये? ...  POSITIVE
79   ईशानला वाटलं की, तो एका बेटावर फेकला गेलाय. त्...  POSITIVE
127  जेवण होताच छकडी आपल्या झोपायच्या खोलीत गेला. ट...  NEGATIVE
190  छकडी धानाची टोपली तशीच टाकून उभा राहिला - “जरा...  NEGATIVE
```

Fig. 6. Independent sentiment analysis of authentic dataset with training and test dataset

Data Preprocessing and Splitting: The dataset underwent preprocessing to facilitate sentiment analysis and named entity recognition (NER). Additionally, the dataset was divided into training and testing sets using the train_test_split function.

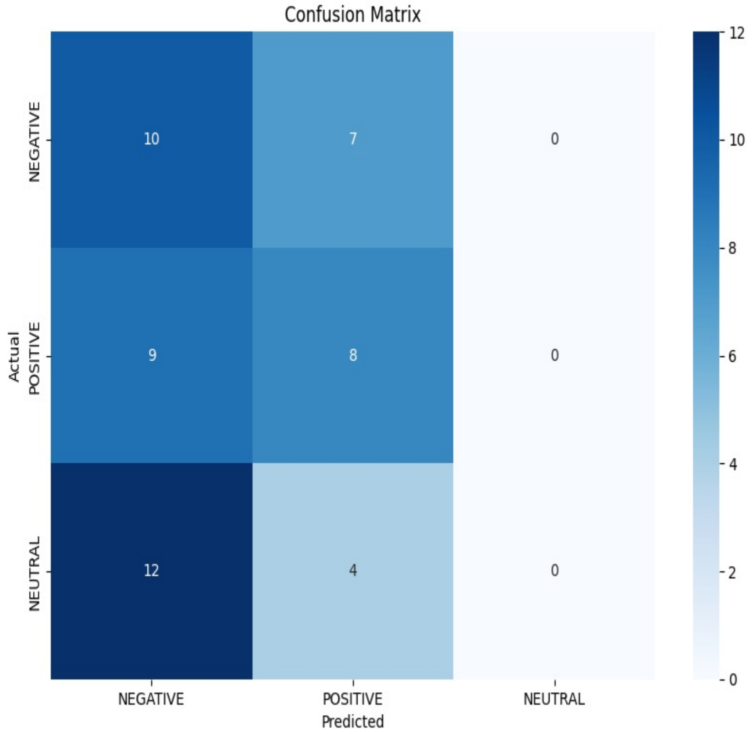


Fig. 7. Confusion matrix for the model accuracy and predictions

Program Overview and Explanation

Library Installation: The necessary libraries were installed by executing the command "pip install pandas transformers deep_translator scikit-learn seaborn matplotlib".

Sentiment Analysis:

To perform sentiment analysis, the Hugging Face pipeline with the identifier "sentiment-analysis" was utilized. The resulting sentiment labels were then added to the dataset.

Named Entity Recognition (NER):

The Hugging Face pipeline with the identifier "ner" was employed to carry out named entity recognition. The identified entities from the text were extracted and stored within the dataset.

Manual Labeling and Validation:

A subset of the dataset, comprising the first 50 rows, was manually labeled for sentiment. The labeled data was subsequently saved to a file named "manually_labeled_data.xlsx".

Evaluation Metrics:

The manually labeled data was loaded and divided into training and testing sets. The sentiment labels were encoded into numeric values to facilitate evaluation.

Classification Report:

Using the classification_report function from the sklearn.metrics library, a comprehensive report was generated. This report encompasses precision, recall, F1-score, and accuracy metrics.

Confusion Matrix:

By utilizing the confusion_matrix function from the sklearn.metrics library, a confusion matrix was created. This matrix was then visualized using the seaborn and matplotlib libraries. The Figure 7 represents the confusion matrix for polarity check

Output Summary:

A summary displaying the sizes of the dataset (total, training, and testing) was presented. Additionally, the accuracy, precision, recall, and F1-score were calculated.

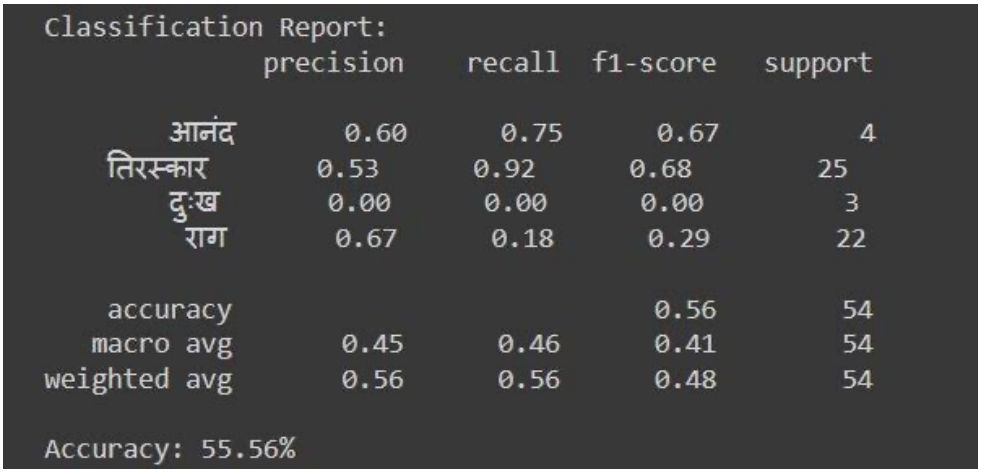
Confusion Matrix Visualization:

The heatmap was used to visualize the confusion matrix, which displays the predicted sentiment labels compared to the actual labels.

Explanation of Outcomes

Dataset Sizes:
Total Dataset: 269 samples
Training Dataset: 215 samples
Testing Dataset: 54 samples

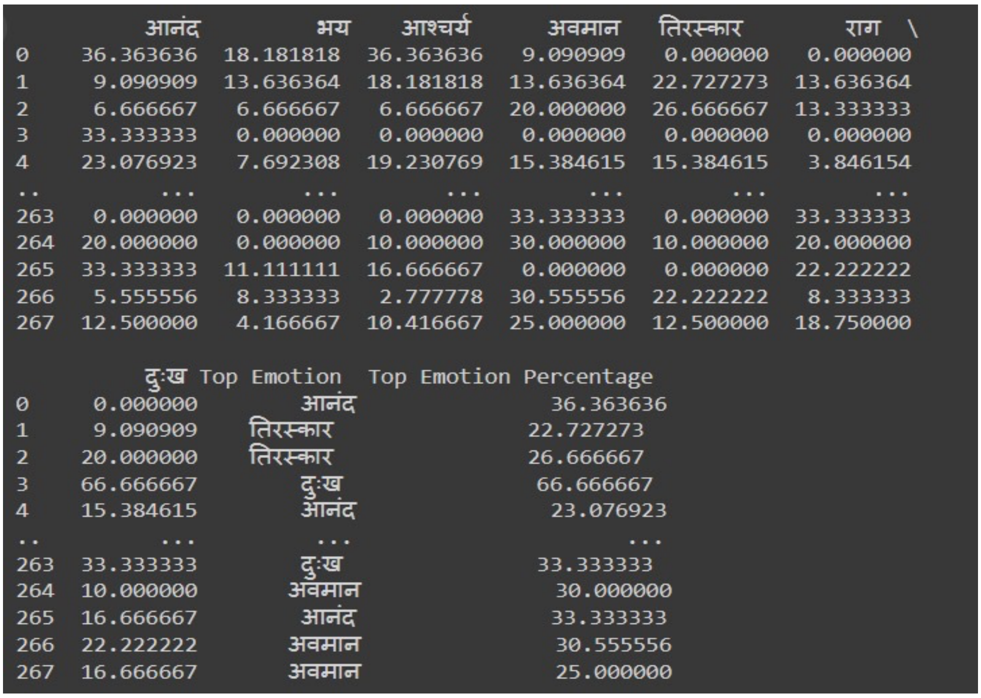
The Python program successfully loads, preprocesses, and evaluates sentiment analysis and NER models on a dataset. It provides comprehensive evaluation metrics and visualizations to assess model performance. Adjustments can be made based on specific dataset characteristics and use cases.



```
Classification Report:
              precision    recall  f1-score   support

       आनंद       0.60      0.75      0.67         4
   तिरस्कार      0.53      0.92      0.68        25
        दुःख      0.00      0.00      0.00         3
        राग      0.67      0.18      0.29        22

    accuracy                           0.56        54
   macro avg       0.45      0.46      0.41        54
weighted avg       0.56      0.56      0.48        54

Accuracy: 55.56%
```

Fig. 8. Classification report scores for all emotion detection model and program

The above classification report presented in Figure 8 is a summary of the sentiment analysis model's performance on different emotions, including precision, recall, F1-score, and support values.



```
          आनंद        भय    आश्चर्य     अवमान   तिरस्कार        राग  \
0    36.363636  18.181818  36.363636   9.090909   0.000000   0.000000
1     9.090909  13.636364  18.181818  13.636364  22.727273  13.636364
2     6.666667   6.666667   6.666667  20.000000  26.666667  13.333333
3    33.333333   0.000000   0.000000   0.000000   0.000000   0.000000
4    23.076923   7.692308  19.230769  15.384615  15.384615   3.846154
..         ...        ...        ...        ...        ...        ...
263   0.000000   0.000000   0.000000  33.333333   0.000000  33.333333
264  20.000000   0.000000  10.000000  30.000000  10.000000  20.000000
265  33.333333  11.111111  16.666667   0.000000   0.000000  22.222222
266   5.555556   8.333333   2.777778  30.555556  22.222222   8.333333
267  12.500000   4.166667  10.416667  25.000000  12.500000  18.750000

          दुःख Top Emotion  Top Emotion Percentage
0     0.000000        आनंद               36.363636
1     9.090909    तिरस्कार               22.727273
2    20.000000    तिरस्कार               26.666667
3    66.666667        दुःख               66.666667
4    15.384615        आनंद               23.076923
..         ...         ...                     ...
263  33.333333        दुःख               33.333333
264  10.000000       अवमान               30.000000
265  16.666667        आनंद               33.333333
266  22.222222       अवमान               30.555556
267  16.666667       अवमान               25.000000
```

Fig. 9. Sentiment Analysis, Emotion Detection and polarity check of each data item for CIIL dataset

The output shown in Figure 9 provided displays the outcomes of a sentiment analysis and emotion detection task conducted on a Marathi language text dataset. It showcases the distribution of emotions in percentage for each sentence and highlights the primary emotion along with its corresponding percentage.

Analysis in Detail:

Emotion Percentages:

The percentages of various emotions like "आनंद" (joy), "भय" (fear), "आश्चर्य" (surprise), "अवमान" (disrespect), "तिरस्कार" (contempt), and "राग" (anger) are provided for each sentence. For example:

- Sentence 0 displays the following emotion percentages: आनंद: 36.36%, भय: 18.18%, आश्चर्य: 36.36%, अवमान: 9.09%, तिरस्कार: 0.00%, राग: 0.00%.
- Sentence 1 indicates आनंद: 9.09%, भय: 13.64%, आश्चर्य: 18.18%, अवमान: 22.73%, तिरस्कार: 22.73%, राग: 13.64%.

Primary Emotion and Its Percentage:

The table below presents the main emotion for each sentence along with its respective percentage. For instance:

- The primary emotion for Sentence 0 is "आनंद" with 36.36%.
- The primary emotion for Sentence 1 is "तिरस्कार" with 22.73%.

Observations:

1. Predominant Emotions:

- "आनंद" (joy) and "तिरस्कार" (contempt) frequently emerge as the primary emotions in the dataset.
- Some sentences exhibit high percentages for "आनंद," indicating a prevalent positive sentiment in those texts.

2. Emotion Distribution:

- The emotion percentages vary significantly among sentences, showcasing a wide array of emotions in the dataset.
- Certain sentences have zero percentages for specific emotions, suggesting the absence of those emotions in those sentences.

3. High Emotion Uniformity:

- In numerous instances, a predominant emotion makes up a substantial part of the total emotional percentage, like "दुःख" (sadness) at 66.67% in a specific case.

4. Data Balance:

The distribution of emotions varies, which may affect the model's ability to accurately predict less common emotions. For instance, emotions such as "disdain" and "anger" have lower overall percentages, potentially resulting in underrepresentation in the model's predictions.

5. Model Performance:

The findings suggest that the model is capable of capturing and distinguishing between different emotions to a satisfactory degree. However, the precision and recall metrics from the previous classification report indicate areas that could be improved, particularly for emotions with limited support.

The sentiment analysis and emotion detection output for the Marathi language text dataset offer a comprehensive overview of the emotional composition of each sentence. While the model demonstrates reasonable performance in identifying the dominant emotion in many instances, further enhancements could be implemented to improve accuracy and achieve a balanced representation across all emotion categories.

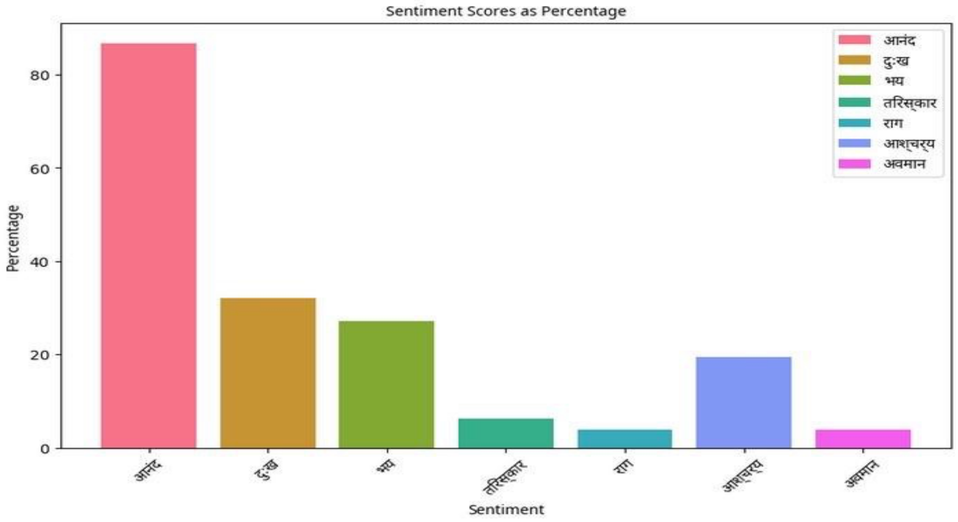


Fig. 10. Bar graph for emotion scoring and classification in the input Marathi text

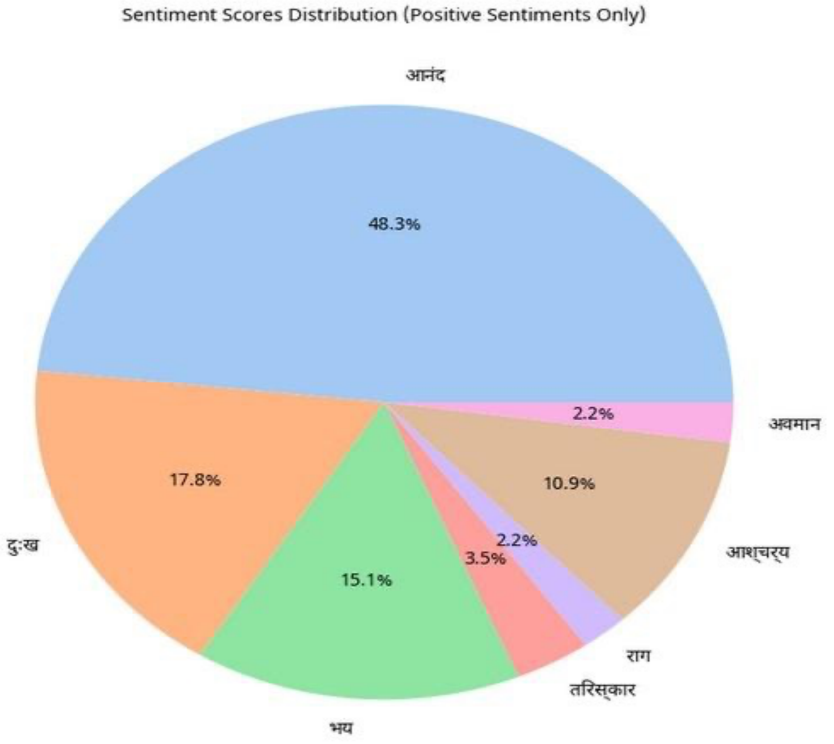
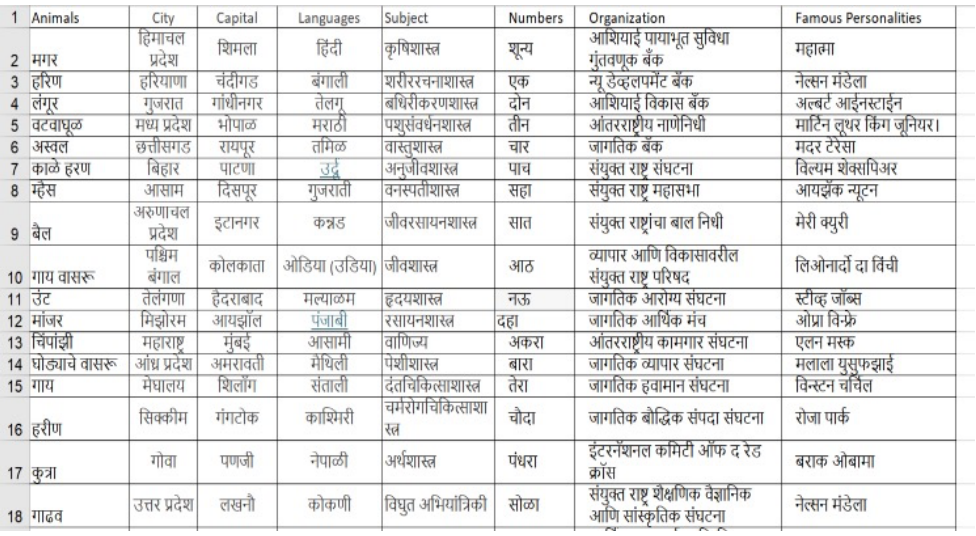


Fig. 11. Pie chart report for emotion detection of input Marathi text

Figure 11 and Figure 10 depict the visual representations for the emotion detection and sentiment analysis for the Regional Language dataset, specifically Marathi text input data. The results obtained from Figure 10. However, collectively, these sentiments reflect the emotions and sentiments found within the input dataset, which were determined through the process of training and testing data splitting.

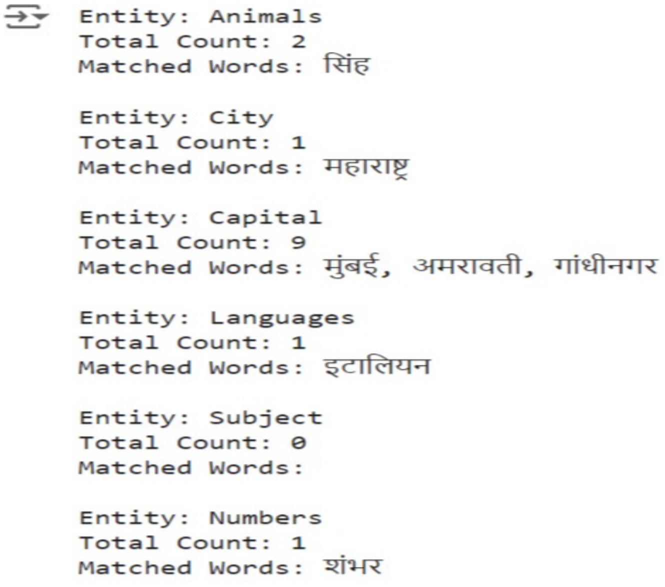
C. Named Entity Recognition for Marathi Language

A dictionary was developed for the Marathi language, focusing on fundamental entities such as animals, cities, languages, subjects, and numbers. This dictionary is represented in Figure 12.



1	Animals	City	Capital	Languages	Subject	Numbers	Organization	Famous Personalities
2	मगर	हिमाचल प्रदेश	शिमला	हिंदी	कृषिशास्त्र	शून्य	आशियाई पायाभूत सुविधा गुंतवणूक बँक	महात्मा
3	हरिण	हरियाणा	चंदीगड	बंगाली	शरीररचनाशास्त्र	एक	न्यू डेव्हलपमेंट बँक	नेल्सन मंडेला
4	लंगूर	गुजरात	गांधीनगर	तेलगू	बधिरीकरणशास्त्र	दोन	आशियाई विकास बँक	अल्बर्ट आईनस्टाईन
5	वटवाघूळ	मध्य प्रदेश	भोपाळ	मराठी	पशुसंवर्धनशास्त्र	तीन	आंतरराष्ट्रीय नाणेनिधी	मार्टिन लूथर किंग जूनियर।
6	अस्वल	छत्तीसगड	रायपूर	तमिळ	वास्तुशास्त्र	चार	जागतिक बँक	मदर टेरेसा
7	काळे हरण	बिहार	पाटणा	उर्दू	अनुजीवशास्त्र	पाच	संयुक्त राष्ट्र संघटना	विल्यम शेक्सपिअर
8	म्हैस	आसाम	दिसपूर	गुजराती	वनस्पतीशास्त्र	सहा	संयुक्त राष्ट्र महासभा	आयझॅक न्यूटन
9	बैल	अरुणाचल प्रदेश	इटानगर	कन्नड	जीवरसायनशास्त्र	सात	संयुक्त राष्ट्रांचा बाल निधी	मेरी क्युरी
10	गाय वासरू	पश्चिम बंगाल	कोलकाता	ओडिया (उडिया)	जीवशास्त्र	आठ	व्यापार आणि विकासावरील संयुक्त राष्ट्र परिषद	लिओनार्दो दा विंची
11	उंट	तेलंगणा	हैदराबाद	मल्याळम	हृदयशास्त्र	नऊ	जागतिक आरोग्य संघटना	स्टीव्ह जॉब्स
12	मांजर	मिझोरम	आयझॉल	पंजाबी	रसायनशास्त्र	दहा	जागतिक आर्थिक मंच	ओप्रा विन्फ्रे
13	चिंपांझी	महाराष्ट्र	मुंबई	आसामी	वाणिज्य	अकरा	आंतरराष्ट्रीय कामगार संघटना	एलन मस्क
14	घोड्याचे वासरू	आंध्र प्रदेश	अमरावती	मैथिली	पेशीशास्त्र	बारा	जागतिक व्यापार संघटना	मलाला युसुफझाई
15	गाय	मेघालय	शिलॉंग	संताली	दंतचिकित्साशास्त्र	तेरा	जागतिक हवामान संघटना	विन्स्टन चर्चिल
16	हरीण	सिक्कीम	गंगटोक	काश्मिरी	चर्मरोगचिकित्साशास्त्र	चौदा	जागतिक बौद्धिक संपदा संघटना	रोजा पार्क
17	कुत्रा	गोवा	पणजी	नेपाळी	अर्थशास्त्र	पंधरा	इंटरनॅशनल कमिटी ऑफ द रेड क्रॉस	बराक ओबामा
18	गाढव	उत्तर प्रदेश	लखनौ	कोकणी	विघुत अभियांत्रिकी	सोळा	संयुक्त राष्ट्र शैक्षणिक वैज्ञानिक आणि सांस्कृतिक संघटना	नेल्सन मंडेला

Fig. 12. Named Entity Recognition dictionary for Marathi language



```
Entity: Animals
Total Count: 2
Matched Words: सिंह

Entity: City
Total Count: 1
Matched Words: महाराष्ट्र

Entity: Capital
Total Count: 9
Matched Words: मुंबई, अमरावती, गांधीनगर

Entity: Languages
Total Count: 1
Matched Words: इटालियन

Entity: Subject
Total Count: 0
Matched Words:

Entity: Numbers
Total Count: 1
Matched Words: शंभर
```

Fig. 13. Named Entity Recognition for Marathi language input text with dictionary based approach

The output displayed in Figure 13 illustrates the fundamental process of Named Entity Recognition for Marathi language text. This process involves comparing each token with a dictionary specifically designed for Marathi, which contains regional values for animals, cities, numbers, capitals, languages, subjects, and famous personalities.

D. Named Entity Recognition for Marathi Language

The algorithm used for the topic modeling of regional language is as depicted as shown in Figure 14:

1. Import the dataset from an Excel file.
2. Define a list of commonly used Marathi stopwords.
3. Implement a function to remove stopwords and clean the sentences.
4. Divide the dataset into training and testing sets.
5. Instantiate and train a TF-IDF vectorizer on the training data. Convert the training and testing data into TF-IDF features.
6. Perform model training with hyperparameter tuning.
7. Make emotion predictions on the test set.
8. Assess the model's performance using classification metrics.
9. Preprocess and transform new sentences using the TF-IDF vectorizer.
10. Utilize the trained SVM model to predict emotions for these new sentences.
11. Develop a function to predict emotions for any new text input.

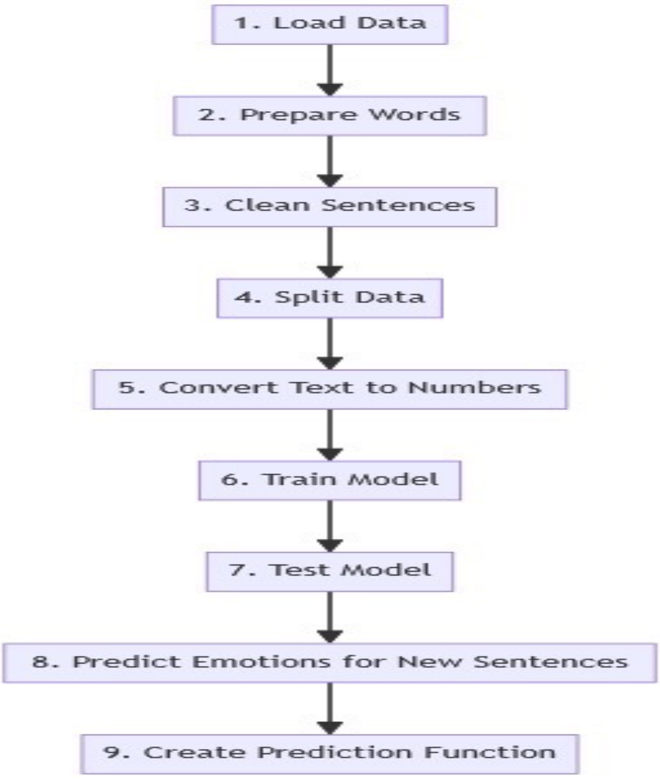
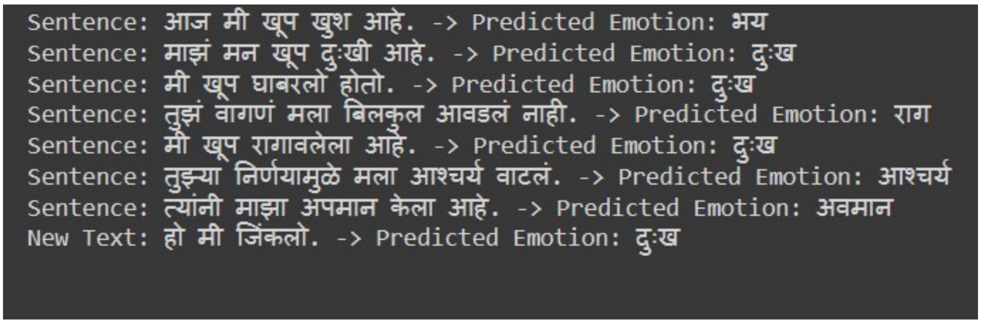

1. Load Data
2. Prepare Words
3. Clean Sentences
4. Split Data
5. Convert Text to Numbers
6. Train Model
7. Test Model
8. Predict Emotions for New Sentences
9. Create Prediction Function

Fig. 14. Topic Modeling for Marathi Language Text with TF_IDF and SVM



```
Sentence: आज मी खूप खुश आहे. -> Predicted Emotion: भय
Sentence: माझं मन खूप दुःखी आहे. -> Predicted Emotion: दुःख
Sentence: मी खूप घाबरलो होतो. -> Predicted Emotion: दुःख
Sentence: तुझं वागणं मला बिलकुल आवडलं नाही. -> Predicted Emotion: राग
Sentence: मी खूप रागावलेला आहे. -> Predicted Emotion: दुःख
Sentence: तुझ्या निर्णयामुळे मला आश्चर्य वाटलं. -> Predicted Emotion: आश्चर्य
Sentence: त्यांनी माझा अपमान केला आहे. -> Predicted Emotion: अवमान
New Text: हो मी जिंकलो. -> Predicted Emotion: दुःख
```

Fig. 15. Marathi language topic modeling predictions

The predictions for the random input Marathi language sentence were performed by a trained model. This model was able to predict seven different emotions based on a dictionary of 95 sentences for each emotion, as depicted in Figure 14. The accuracy achieved by the model was 97%, as indicated in the classification report shown in Figure 15. It is important to note that the results may vary depending on the dialect and the limited vocabulary available for each emotion, which is a result of the model's limited capacity.

V. CONCLUSION

In our recent efforts to improve sentiment analysis for the Marathi language, we utilized two different methods: a dictionary-based approach using Marathi words and emotions as key-value pairs, and a corpus-based Marathi language dictionary for sentiment analysis covering seven emotions. Our polarity analysis has demonstrated satisfactory outcomes in categorizing positive, negative, and neutral words from the input text. Due to the Marathi language's limited resources and dataset, we have managed to identify various basic emotions from short paragraphs and extensive text files. Nevertheless, the current accuracy of our models remains relatively low. This is mainly attributed to the scarcity of precise Marathi words and their corresponding emotions, as well as the language's adaptability, subjectivity, and contextual variations in the input text. This scenario offers significant prospects for future improvements. Our strategy involves refining our analysis by carefully examining each word and its sentiment score. This will enhance the precision of our models and lead to improved performance across all sections of our evaluation framework. With current framework and machine learning model we have got 90% accuracy. The Named Entity recognition and the topic modelling approaches right now give satisfactory results and are not trained on huge datasets. In future we will try to train all the models with big datasets and more approximate inputs from the language experts to get more reliable and highly accurate results.

VI. ACKNOWLEDGMENT

I wish to convey my deepest appreciation to Swami Ramanand Teerth Marathwada University (SRTMUN) and the College of Computer Science and Technology (COCSIT) Research Center for their invaluable assistance and guidance during the progression of this research. Their unwavering support and provision of essential resources have played a crucial role in the successful completion of this study. I would like to express my sincere gratitude to the faculty members, administrative staff, and my fellow researchers at COCSIT for their continuous support and valuable feedback, which have greatly enriched the quality of this work. A special acknowledgment goes to my research mentors for their expertise, patience, and commitment, which have served as a source of motivation and inspiration. The conducive academic atmosphere and cutting-edge facilities offered by SRTMUN and COCSIT have been indispensable for the realization of this research. I am deeply thankful for the opportunity to conduct my study in such a nurturing and intellectually stimulating setting. I extend my heartfelt thanks once again to everyone at SRTMUN and COCSIT for their exceptional support and for fostering a culture of academic excellence.

REFERENCES

[1] Ansari, Mohammed Arshad and Govilkar, Sharvari, Sentiment Analysis of Mixed Code for the Transliterated Hindi and Marathi Texts (2018). International Journal on Natural Language Computing (IJNLC) Vol. 7, No.2, April 2018, Available at SSRN: https://ssrn.com/abstract=3429694 or http://dx.doi.org/10.2139/ssrn.3429694

[2] Bolaj, Pooja & Govilkar, Sharvari. (2016). Text Classification for Marathi Documents using Supervised Learning Methods. International Journal of Computer Applications. 155. 6-10. 10.5120/ijca2016912374.

[3] Ratna Nitin Patil, Yadvendra Pratap Singh, Shitalkumar Adhar Rawandale, Sofia Singh,Improving Sentiment Classification on Restaurant Reviews Using Deep Learning Models, Procedia Computer Science, Volume 235,2024,Pages 3246-3256,ISSN 1877-0509,https://doi.org/10.1016/j.procs.2024.04.307.

[4] Shivani Jain, Anju Saha, Improving and comparing performance of machine learning classifiers optimized by swarm intelligent algorithms for code smell detection, Science of Computer Programming, Volume 237, 2024, 103140, ISSN 0167-6423, https://doi.org/10.1016/j.scico.2024.103140.

[5] Abdulfattah Ba Alawi, Ferhat Bozkurt, A hybrid machine learning model for sentiment analysis and satisfaction assessment with Turkish universities using Twitter data, Decision Analytics Journal,Volume 11, 2024, 100473, ISSN 2772-6622, https://doi.org/10.1016/j.dajour.2024.100473.

[6] Punithavathi Rasappan, Manoharan Premkumar, Garima Sinha, Kumar Chandrasekaran, Transforming sentiment analysis for e-commerce product reviews: Hybrid deep learning model with an innovative term weighting and feature election, Information Processing & Management, Volume 61, Issue 3, 2024, 103654, ISSN 0306-4573, https://doi.org/10.1016/j.ipm.2024.103654.

[7] Aruna Gladys A., Vetriselvi V., Sentiment analysis on a low-resource language dataset using multimodal representation learning and cross-lingual transfer learning, Applied Soft Computing, Volume 157, 2024, 111553, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2024.111553.

[8] Linguistic Data Consortium for Indian Languages (LDC-IL)https://www.ldcil.org/

[9] S. Sirsat and N. Zulpe, "Efficient Dataset Preparation Techniques for Regional/Marathi Language Analysis: Creating Customized Dataset for Regional Language/Marathi Language Text Analysis," 2023 Somaiya International Conference on Technology and Information Management (SICTIM), Mumbai, India, 2023, pp. 90-95, doi: 10.1109/SICTIM56495.2023.10104666. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10104666&isnumber=10104656

[10] S. Sirsat and N. Zulpe, "Recognizing Sentiments In Regional Language Text From Social Media Using Machine Learning Approach: A comparative Study to understand the state of the art tools and associated functionalities available," 2023 International Conference on Advanced Computing Technologies and Applications (ICACTA), Mumbai, India, 2023, pp. 1-8, doi: 10.1109/ICACTA58201.2023.10393104. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10393104&isnumber=10391864

[11] S. Sirsat and N. Zulpe, ”Recognizing Sentiments of Maharashtra State Regional Language Marathi Text from Social Media Using Machine Learning techniques”, Journal of Informatics Education and Research, Vol. 4 No. 2 (2024), ISSN: 1526-4726, pg no: 1872,86; Url: https://www.jier.org/index.php/journal/article/view/1004

[12] S. Sirsat and N. Zulpe, “Recognizing Sentiments In Regional Language Text From Social Media Using Machine Learning Approach : the way ahead through cross domain applications”, COCSIT Conference Proceedings, ISSN – 2322-0015