Enhancing Trust and Interpretability in Malayalam Sentiment Analysis with Explainable AI

Meharuniza Nazeem, Anitha R, Navaneeth S, Rajeev R R, Selvaraj R

International Centre for Free and Open Source Solutions (ICFOSS) Government of Kerala,
Thiruvananthapuram Kerala, India
meharuniza@icfoss.org, anitha@icfoss.org navaneeths@icfoss.org, rajeev@icfoss.in, selvaraj@icfoss.org

***Abstract*—Natural language processing (NLP) has seen a rise in the use of explainable AI, especially for low-resource languages like Malayalam. This study builds on our earlier research on sentiment analysis which uses identified views to classify and understand the context. Support Vector Machine (SVM) and Random Forest (RF) classifiers are two machine learning approaches that we used to do sentiment analysis on the Kerala political opinion corpus. Using Bag-of-Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) features, we construct feature vectors for sentiment analysis. In this, analysis of the Random Forest classifier's performance shows that it outperforms SVM in terms of accuracy and efficiency, with an accuracy of 85.07 %. Using Local Interpretable Model-Agnostic Explanations (LIME) as a foundation, we address the interpretability of text classification and sentiment analysis models. This integration increases user confidence and model use by offering concise and understandable justifications for model predictions. The study lays the groundwork for future developments in the area by demonstrating the significance of explainable AI in NLP for low-resource languages.**

***Keywords*—Natural language processing (NLP), sentiment analysis, explainability in NLP, random forest classifier, LIME.**

I. Introduction

Focusing on the natural language exchange between computers and humans, Natural Language Processing (NLP) is an important area of research. Languages such as English have seen tremendous advancements in NLP, but at the same time low-resource languages have not received the same degree of attention. India's linguistic diversity, with 22 official languages and countless dialects, presents unique challenges, such as rich morphology, and limited annotated datasets. Despite these hurdles, the development of NLP for Indian languages is essential for inclusivity and accessibility, enabling millions to benefit from technology and digital services in their native tongues.

Malayalam is a well-known Dravidian language that is spoken mostly in the Indian state of Kerala. Malayalam is spoken by more than 34 million people and has a strong literary heritage. However, its under-representation in the NLP arena hinders the advancement of the language in the computational space and its use in digital platforms and applications. Recent advancement in NLP technology has seen a shift from traditional machine learning and deep learning approaches to large language models (LLMs). One of the main advantages of deep learning and LLMs is that they do not require extensive knowledge of linguistic rules or grammatical concepts to perform NLP tasks. Instead, they learn patterns and representations directly from large amounts of data, making them highly effective for tasks like text classification, translation, and sentiment analysis.

Large volumes of data and processing power are frequently needed for these models, which can be problematic for low-resource languages like Malayalam. Furthermore, although they are excellent at identifying patterns, they can lack interpretability, which makes it challenging to comprehend the decision-making process. This is where the topic of explainable AI comes into play; it aims to increase the transparency and understandability of these intricate models' inner workings, so as to guarantee the dependability and trustworthiness of their use in vital applications.

The objective of Explainable AI (XAI) is to provide light on machine learning models' decision-making processes in order to overcome their opacity. This is particularly important for NLP jobs where it's necessary to understand the reasoning behind model predictions, such as text classification. When it comes to sentiment analysis for Malayalam text data, XAI techniques can clarify for consumers the reasons behind a given text's classification as positive, negative, or neutral. To reveal the underlying workings of these models, methods like feature importance and visualization tools like LIME (Local Interpretable Model-agnostic Explanations) are used.

Explainable AI techniques in NLP can help practitioners and researchers dealing with low-resource languages better understand how language models understand and analyze text. We may improve the transparency and dependability of these models by using XAI approaches. This would eventually improve user trust and encourage a wider adoption of NLP technology in regional languages.

This paper's primary contributions are: Explainable AI (XAI) for Sentiment Analysis in Malayalam: Emphasizes how significant XAI is to improving interpretability and trust for low-resource languages like Malayalam.

- Sentiment Analysis Model and Performance: Using a Random Forest classifier, a model was developed with 22,519 phrases and achieved 85.07% accuracy, surpassing SVM.
- LIME Integration: LIME was integrated to improve model openness by providing word/phrase contributions that explain predictions.

This paper is structured to provide a comprehensive understanding of the application of explainable AI in text classification, specifically focusing on sentiment analysis of Malayalam text data. In Section 2, we review related works, examining previous research in explainable AI and text classification, with an emphasis on low-resource languages. This is followed by a discussion of the challenges faced in this domain, including data scarcity, computational limitations, and the complexity of Malayalam language processing. In the methodology section, we outline our approach, detailing the data collection process, preprocessing techniques, and the models used. The implementation section delves into the technical aspects of our solution, describing the tools and frameworks employed to develop the model. Subsequently, the experiments and results section presents the findings of our research, including the evaluation metrics and the performance of our models. By following this structured approach, we aim to provide valuable insights into the potential of explainable AI for enhancing NLP applications in Malayalam and other low-resource languages.

II. Related works

This section focuses on relevant research in explainable AI and interpretable machine learning within Natural Language Processing models.

The work, Aman Piyanshu et.al. [1] underscores the pivotal role of interpretability in text classification and sentiment analysis tasks, employing advanced model-agnostic explanation techniques such as LIME and SHAP. LIME elucidates local explanations by simplifying complex models, while SHAP leverages Shapley values to provide feature influence scores for the entire model. These methodologies enable a deeper understanding of AI model decisions. In sentiment analysis, a range of techniques including random forests, Convolutional Neural Networks (CNNs), Long Short Term Memory (LSTM) networks, and Bidirectional Encoder Representations from Transformers (BERT) are explored, emphasizing structured models and contextual learning. By harnessing these explanation methods, this study aims to enhance transparency and accountability in model architectures, fostering trust and wider adoption of NLP technologies.

Asrita Venkata Mandalam et. al. [2] describes methods for leveraging Tamil and Malayalam datasets to classify the polarity of Dravidian code-mixed comments. It suggests three methods: an architecture based on machine learning, a model at the sub-word level, and a model based on word embedding. While the machine learning model makes use of TF-IDF vectorization and Logistic Regression, the sub-word and word embedding models make use of Long Short Term Memory (LSTM) networks with language-specific preprocessing. When the sub-word level model was first presented, it placed fifth for Tamil and twelve places for Malayalam in the ”Sentiment Analysis for Dravidian Languages in Code-Mixed Text” track at FIRE 2020. With weighted F1-scores of 0.65 for Tamil and 0.68 for Malayalam, this paper improves on the previous findings. The Tamil score is comparable to the highest in the FIRE 2020 track.

Ramisa Anan et. al. [3] introduces a BERT-based model for Bangla sarcasm detection, achieving 99.60% accuracy, surpassing traditional methods. It utilizes BanglaSarc, a new dataset, and employs LIME for explainability, enabling insights into model decisions. By leveraging BERT's deep features and conducting thorough comparisons, the research enhances understanding of sarcasm detection. Addressing the scarcity of Bangla sarcasm research, this work aims to bridge language gaps and improve model interpretability for nuanced linguistic analysis.

Building on these studies, we developed a sentiment analysis model for Malayalam using machine learning techniques and LIME for interpretability [4]. This approach allows us to classify sentiment accurately while providing clear, understandable explanations for each prediction. By leveraging LIME, our model not only predicts sentiment but also offers insights into the reasoning behind its decisions, enhancing transparency and trust in the system.

Support Vector Machine (SVM) and Random Forest (RF) classifiers are examples of machine learning techniques that were used in Soumya, S. et al. [6]'s work to classify the sentiment of the dataset. Sentiwordnet feature matrices, Bag-of-Words (BoW), and Term-Frequency against Inverse Document Frequency (TF-IDF) were used to vectorize the input dataset. The accuracy achieved by the lexicon-based method was 84.8%. The accuracy of machine learning techniques using the Sentiwordnet feature vector was 92.6%, 92.9%, and 93.4% for SVM (kernel = linear), SVM (kernel = RBF), and RF, respectively. According to Thulasi, P.K., and Usha, K. (2016) [8], consumers don’t benefit much from polarity categorization performed by standard sentiment analysis tasks alone. Sentiment analysis tasks are enhanced in utility if the system also detects the element on which the user is commenting.

Varun Sundaram, Saad Ahmed, et al [7]. and colleagues have developed a novel approach that involves text preparation and feature selection. This research highlights the use of high-quality labor in order to provide insight into emotion analysis. The general objective is to preprocess the text in the dataset, extract and represent word contextual usage using our method, and then run TF-IDF to comprehend word contextual usage and decide weights assigned to each word. According to Thulasi, P.K., and Usha, K. (2016) [8], users don’t benefit much from polarity categorization performed by standard sentiment analysis tasks alone. Sentiment analysis tasks are improved in utility if the system also detects the element on which the user is commenting. This is where aspect-based sentiment analysis becomes important. With 84.7% accuracy, this system performs sentence-level aspect-based sentiment analysis for Malayalam movie and product reviews.

III. Methodology

The sentiment analysis component of this study involves classifying Malayalam text data as positive, negative, or neutral sentiments. The dataset comprises 18,574 sentences, sourced from survey responses and questionnaires from various constituencies in Kerala, aimed at understanding political viewpoints [5]. To enhance the corpus, an additional 3,945 comments relevant to the same subject were scraped from social media platforms, bringing the total to 22,519 sentences. This dataset is split into 60% for training, 20% for validation, and 20% for testing. The architecture we followed for this work is shown below.

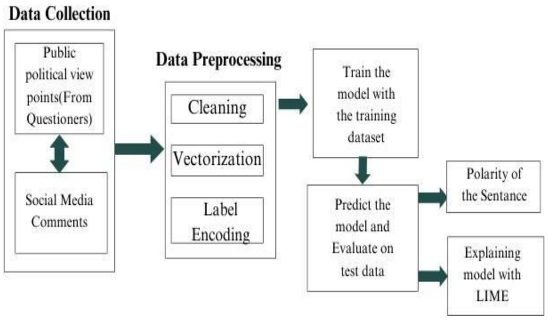
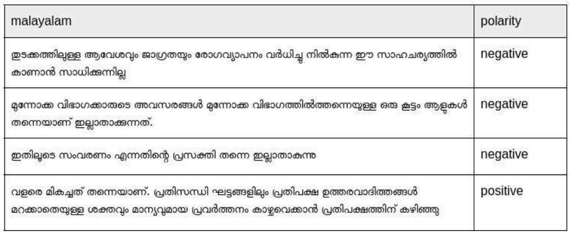

Data Collection
Public political view points(From Questioners)
Social Media Comments
Data Preprocessing
Cleaning
Vectorization
Label Encoding
Train the model with the training dataset
Predict the model and Evaluate on test data
Polarity of the Sentance
Explaining model with LIME

Figure 1. Architecture

In the data preprocessing phase, several steps are undertaken to clean the text data. Hyperlinks, punctuation, and special characters are removed using regular expressions in Python. Further cleaning involves eliminating stop words, unused words, punctuation, and extra white spaces. Each sentence is then manually annotated with sentiment labels—positive, negative, or neutral. This results in a dataset with 7,475 sentences labeled as positive, 7,315 as negative, and 7,729 as neutral.



malayalam	polarity
തുടക്കത്തിലുള്ള ആവേശവും ജാഗ്രതയും രോഗവ്യാപനം വർധിച്ചു നിൽകുന്ന ഈ സാഹചര്യത്തിൽ കാണാൻ സാധിക്കുന്നില്ല	negative
മുന്നോക്ക വിഭാഗക്കാരുടെ അവസരങ്ങൾ മുന്നോക്ക വിഭാഗത്തിൽത്തന്നെയുള്ള ഒരു കൂട്ടം ആളുകൾ തന്നെയാണ് ഇല്ലാതാക്കുന്നത്.	negative
ഇതിലൂടെ സംവരണം എന്നതിന്റെ പ്രസക്തി തന്നെ ഇല്ലാതാകുന്നു	negative
വളരെ മികച്ചത് തന്നെയാണ്. പ്രതിസന്ധി ഘട്ടങ്ങളിലും പ്രതിപക്ഷ ഉത്തരവാദിത്തങ്ങൾ മറക്കാതെയുള്ള ശക്തവും മാന്യവുമായ പ്രവർത്തനം കാഴ്ചവെക്കാൻ പ്രതിപക്ഷത്തിന് കഴിഞ്ഞു	positive

Figure 2. Sample dataset

Following preprocessing, feature vectors are created using Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) representations [12]. BoW represents the text data by focusing on word occurrences without considering their order, making it straightforward but limited. In contrast, TF-IDF weights words based on their importance in a document relative to the entire corpus [19], providing a more refined representation. These vectors are used as inputs for machine learning classifiers, transforming the text data into a numerical format suitable for analysis.

A Random Forest (RF) classifier is used to train the model. By dividing opinions into positive, negative, and neutral categories, this classifier efficiently manages non-linear relationships within the data [13] [20]. By building numerous decision trees and training them on distinct feature sets and subsets of the data, the RF classifier makes use of an ensemble learning technique. The overall classification accuracy is enhanced by the RF classifier, which integrates the predictions from all decision trees. With its ensemble learning method, RF kernel, decision tree aggregation, and feature selection capabilities, the Random Forest classifier shows to be an effective tool for sentiment categorization [21]. This all-encompassing strategy guarantees that the model can manage a wide range of intricate data patterns, leading to more accurate and dependable sentiment analysis.

The performance of the model is evaluated using metrics such as accuracy, precision, recall, and F1 score. Accuracy measures the proportion of correctly classified instances, while precision and recall provide insights into the model’s ability to identify positive instances and retrieve relevant ones. The F1 score, a harmonic mean of precision and recall, offers a single comprehensive measure of performance. Both manual and system accuracy are computed to ensure robust evaluation, with the model predicting sentiment labels for the test dataset based on the training data.

In this study, we leverage the LIME framework for enhancing the transparency and interpretability of a text classification model. The process unfolds with the definition of class names, followed by the creation of a pipeline encapsulating the vectorizer and classifier components. Subsequently, we select instances from the dataset for explanation and predict their probabilities using the constructed pipeline. To unravel the model's decision-making, we employ the LIME Text Explainer, emphasizing the local fidelity of explanations around specific instances.

The explanation generation involves initializing the explainer, choosing a representative text instance, and obtaining a detailed explanation, including feature contributions and weights. Additionally, the predicted probability for the chosen instance is displayed.

The LIME Framework, an acronym for Local Interpretable Model-Agnostic Explanations, focuses on local fidelity, meaning that the explanation should accurately represent the classifier's behavior in the vicinity of the predicted instance [1]. LIME possesses the capability to elucidate any model without requiring access to its internal workings, making it model-agnostic. Since many classifiers employ non-intuitive representations (e.g., word embeddings for text classification), LIME interprets such classifiers in terms of understandable representations (words), even if they differ from the classifier's original representation.

The explanation of the terms mentioned above:

- Local Fidelity: LIME focuses on explaining the local behavior of the complex model around the specific instance being predicted. This ensures that the explanation accurately reflects the model's behavior in the vicinity of the given data point.
- Model-Agnostic: LIME can explain any black-box model without accessing its internal parameters. This makes it versatile and applicable to a wide range of machine learning models.
- Interpretable Representations: In cases where the original model uses non-intuitive representations, such as word embeddings, LIME interprets the model's predictions in terms of more understandable representations, like individual words.
- Feature Selection for Simplicity: The algorithm selects a minimal set of features that are most relevant to the model's prediction, aiming for a simple and interpretable explanation.
- Simple Model Fitting: A simple model is trained on the permuted data, incorporating the selected features and similarity scores as weights. This simple model serves as a local approximation of the black-box model, providing an interpretable explanation for the prediction.

IV. Implementation

A. Sentiment Analysis

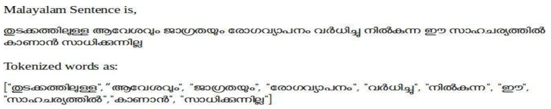
In this section, we outline the implementation of sentiment analysis for Malayalam text data. The goal is to classify sentences into positive, negative, or neutral sentiments using machine learning techniques [14].

1) Data Preprocessing: The preprocessing phase is essential to prepare the raw text data for analysis. The following steps are performed:

- Removal of Hyperlinks, Punctuation, and Special Characters: All hyperlinks, punctuation marks, and special characters are eliminated from the text to clean the data.
- Tokenization: The text is split into individual words or tokens.
- Stop Words Removal: Commonly used words that do not contribute to sentiment (stop words) are removed.
- Negation Handling: Words that indicate negation are identified and handled appropriately, which helps in better sentiment classification.
- Stemming and Lemmatization: The words are reduced to their root form to standardize the text data.

2) Feature Extraction: : Following preprocessing, feature vectors are created using the following methods:

- Bag-of-Words (BoW): This method represents text data as a collection of word counts or binary indicators.
- Term Frequency-Inverse Document Frequency (TF-IDF) [9]: TF-IDF reflects the importance of a word in a document [17][18] relative to the entire dataset.



Malayalam Sentence is,

തുടക്കത്തിലുള്ള ആവേശവും ജാഗ്രതയും രോഗവ്യാപനം വർധിച്ച നിൽകുന്ന ഈ സാഹചര്യത്തിൽ കാണാൻ സാധിക്കുന്നില്ല

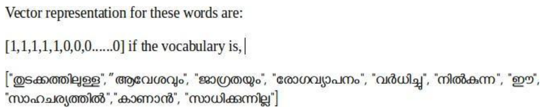
Tokenized words as:

["തുടക്കത്തിലുള്ള","ആവേശവും", "ജാഗ്രതയും", "രോഗവ്യാപനം", "വർധിച്ച", "നിൽകുന്ന", "ഈ", "സാഹചര്യത്തിൽ","കാണാൻ", "സാധിക്കുന്നില്ല"]

Figure 3. Tokenized Sentence

3) Model Training: A machine learning classifier is used to develop the model using the training data:

- Random Forest (RF): The RF classifier is trained using sentence-level approaches and is used to classify data into positive, negative, or neutral sentiments.



Vector representation for these words are:

[1,1,1,1,1,0,0,0......0] if the vocabulary is,

["തുടക്കത്തിലുള്ള","ആവേശവും", "ജാഗ്രതയും", "രോഗവ്യാപനം", "വർധിച്ച", "നിൽകുന്ന", "ഈ", "സാഹചര്യത്തിൽ","കാണാൻ", "സാധിക്കുന്നില്ല"]

Figure 4. Vectorized words

4) Model Evaluation: The performance of the proposed model is evaluated using the precision, recall, and F1-score metrics. Additionally, a confusion matrix [10] has been utilized to display the data and show the predictions made by the model for every class. After training, the model is evaluated using the test dataset to predict accuracy. The performance of the models is assessed using the following metrics:

- **Accuracy:** The proportion of correctly classified sentences in the testing set relative to the total number of sentences.
- **Precision:** The percentage of real positive predictions (positive sentiment correctly

anticipated) among all positive forecasts.

$$Precision = \frac{True\ Positive}{True\ Positive\ + False\ Positive}$$

- **Recall:**The percentage of accurate positive predictions among the testing set's actual positive sentences.

$$Precision = \frac{True\ Positive}{True\ Positive\ + False\ Positive}$$

- **F1-score:** A balance between the two measures is provided by the harmonic mean of precision and recall.

$$F1 = \frac{2 * Precision\ *\ Recall}{Precision\ +\ Recall}$$

The classifiers are used to predict the accuracy of the test dataset following model development. Both manual and system accuracy are computed to ensure robust evaluation.

B. LIME Interpretation

Algorithm: LIME_Framework_Algorithm [1]

Input : Instance to be Explained (x): The data point for which we want to provide an interpretable explanation.

Output: Interpretable Explanation (explanation): A simplified model and associated features that approximate the behavior of the complex model around the given instance.

Steps:

1) Permute Data: Generate perturbed instances by permuting the features of the original instance x.

2) Calculate Distance/Similarity Score Metric: Measure the distance or similarity score between the original instance x and the perturbed instances. This metric quantifies the difference between the instances.

3) Make Predictions on New Data Using Complex Model: Utilize the complex, black-box model to make predictions on the perturbed instances.

4) Select Minimum Features for Maximum Likelihood: Identify the minimum number of features or predictors that maximize the likelihood of the predicted class by the black-box model.This step aims to capture the essential features influencing the prediction.

5) Fit Simple Model to Permuted Data: Fit a simple, interpretable model (e.g., linear regression, decision tree) to the permuted data. Use the selected features and similarity scores as weights in the simple model.

V. Experiments and results

A. Sentiment Analysis

To predict Malayalam data in our study, we employed classifiers based on machine learning. Opinion mining was conducted in Malayalam, and the steps involved in using the RF classifiers . To begin with, we need a corpus of Malayalam statements that have been classified as neutral, negative, or positive depending on the attitude expressed in each sentence. The prediction of the word being actually true positive, true negative, false positive, and false negative. The performance of the proposed model is evaluated using the precision, recall, and F1-score metrics.

The feature vectors for the corpus were generated using radial basis function SVM [11]and RF models, and classifiers were then applied. Sentence-level opinion mining was used to train the TFIDF with RF classifier, whereas the RBF classifier employed the SVM. After switching from sentence-level vectorization to word-level vectorization, the analysis revealed that the TF-IDF in the RF classifier performed better than the SVM model, achieving an astounding accuracy of 85.07%. However, the former's accuracy was only 59.99%.

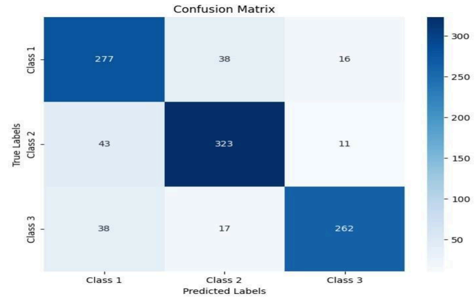
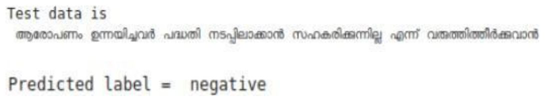


Figure 5. Confusion Matrix

A confusion matrix has been employed to analyze the data and visualize the model's predictions for each class [15] as shown in figure 5. It evaluates the accuracy of the model on a dataset of 22,519 corpora. The first column of the confusion matrix represents true positive values, indicating instances where the model correctly identified the training data and predicted new corpora accurately. Figure 6 below illustrates the sentiment analysis predictions.



```
Test data is
 ആരോപണം ഉന്നയിച്ചവർ പദ്ധതി നടപ്പിലാക്കാൻ സഹകരിക്കുന്നില്ല എന്ന് വരുത്തിത്തീർക്കുവാൻ

Predicted label =  negative
```

Figure 6. Sentiment Analysis output

B. LIME

The explainability tools are intended to clarify how machine learning models behave and make decisions, particularly when applied to classification problems. Let's examine each image and its meaning in more detail. This visualization is probably a bar chart that shows below figure 7 how different aspects of a particular instance affect the prediction to move in the direction of or away from the ”neutral” class.

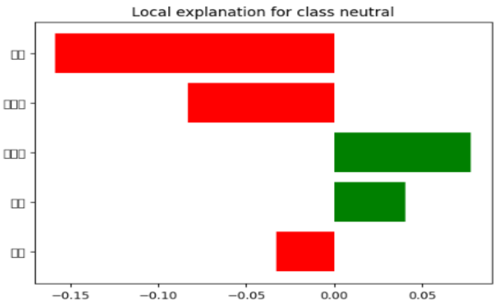


Figure 7. LIME Chart

The model’s predictions for the three classes—”negative” ”neutral” and ”positive” are comprehensively summarised in the figure
8. Prediction probabilities are presented in the leftmost area. The model predicts a 52% chance for the ”negative” class, 7% for ”neutral” and 41% for ”positive”. The remaining portion of the picture is separated into multiple columns, each of which represents a distinct classification aspect:

- NOT negative: Characteristics supporting the hypothesis that the situation is not ”negative”.
- Negative: Characteristics supporting the hypothesis that the situation is ”negative”.
- NOT positive: Characteristics supporting the hypothesis that the case is not ”positive”.
- Positive: Characteristics that support the hypothesis that the situation is ”positive”.
- NOT neutral: Characteristics supporting the hypothesis that the situation is not ”neutral”.
- Neutral: Characteristics that support the hypothesis that the instance is ”neutral”.

The size and direction of each feature’s contribution towards or away from a specific class are shown by the length and direction (positive or negative) of these bars. Once more, features are displayed in what looks to be Malayalam script along the y-axis.

A multifaceted knowledge of how the model distinguishes across classes is made possible by this thorough presentation. These particular predictions can be made by looking at the ”Negative” and ”Positive” columns to determine which traits are important. In a similar vein, the characteristics that discourage the model from predicting those classes are indicated in the ”NOT positive” and ”NOT negative” columns.

The LIME values, which define how much each characteristic contributes to the prediction, are shown on the x-axis. Negative values move the forecast away from the ”neutral” class and towards the positive class Although the character used is unfamiliar, possibly Malayalam the y-axis lists features.

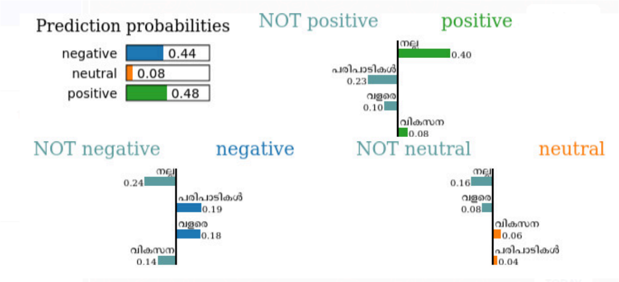


Figure 8. LIME Explanation

It is important to note how the bars are colored: green bars indicate features that have a positive influence on the ”neutral” forecast, while red bars indicate features that have a negative impact. A clear visual depiction of the most influential features is provided by the length of each bar, which represents the contribution magnitude of each feature.

If a characteristic significantly detracts from the ”neutral” classification, for example, a long red bar would suggest a major negative influence. A lengthy green bar, on the other hand, indicates a significant positive influence and strongly supports the ”neutral” designation. When analyzing model behaviour at a detailed level, this kind of chart is quite helpful as it lets the user know exactly which attributes are responsible for a given forecast.

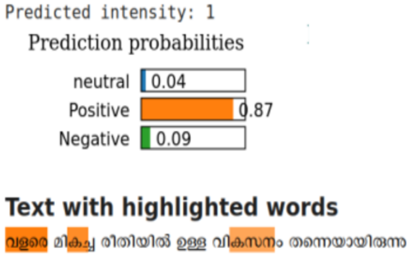


Figure 9. LIME Explanation for given sentence

The sentiment analysis of a Malayalam sentence is shown in the figure 9, where the model predicts a high probability of 87% for the sentiment to be positive. The keywords that contribute to the positive sentiment in the sentence are visualized in the accompanying bar chart. This sentiment has a strong influence on the positive prediction, with a contribution of 0.25. The weights of the various words, indicated in orange, are combined according to their influence on the model’s prediction to determine the overall mood of the statement. The phrase ”വളെര മികച്ച രീതിയിലുള്ള വികസനം തെന്നയായിരുപ്” means ”It was an excellent development towards.” ”വളെര മികച്ച” (extremely excellent) and ”വികസനം” (development) are two phrases that greatly add to the positive atmosphere. The algorithm successfully matches its prediction with the positive tone of the text, demonstrating that it can accurately detect and understand sentiment.

VI. Conclusion

This study concludes by examining the importance of explainable AI in the fields of sentiment analysis and text categorization, specifically for the Malayalam language. By utilizing cutting-edge methods like LIME, we improve our machine learning models' interpretability and make the analysis process more clear and reliable. Using techniques such as Random Forest classifiers and TF-IDF vectorization, our sentiment analysis model achieves strong results on a manually annotated dataset of Malayalam words. We address the important need for model transparency in AI applications by adding LIME, which allows us to deliver understandable explanations for every prediction. This work closes a major gap in sentiment analysis for low-resource languages like Malayalam, while also adding to the expanding corpus of research on explainable AI. In order to further advance the subject of interpretable machine learning in natural language processing, future research will focus on increasing the dataset and increasing computational efficiency.

Acknowledgment

We would like to express our sincere gratitude to Niji G S and Dinesh Lal D for their invaluable coordination and support throughout our study. Special thanks are also extended to Arun A and Arya Sreekumar for their exceptional technical expertise, which was instrumental in the successful completion of this research.

References

[1] Aman Priyanshu* 1 , Sudarshan Sivakumar* 2 , Supriti Vijay* 2 , Nipuna Chhabra* 2 , Aleti Vardhan, Something Something Hota Hai!" An Explainable Approach towards Sentiment Analysis on Indian Code-Mixed Data, Proceedings of the 2021 EMNLP Workshop W-NUT: The Seventh Workshop on Noisy User-generated Text, pages 437–444.

[2] Asrita Venkata Mandalam, Yashvardhan Sharma, Sentiment Analysis of Dravidian Code Mixed Data,Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, pages 46–54April 20, 2021 ©2021.

[3] Ramisa Anan, Tasnim Sakib AponZeba ,Tahsin Hossain,Interpretable Bangla Sarcasm Detection using BERT and Explainable AI, arXiv:2303.12772v1 [cs.CL] 22 Mar 2023.

[4] Vaishak Belle,Ioannis Papantonis, Principles and Practice of Explainable Machine Learning, arXiv:2009.11698v1 [cs.LG] 18 Sep 2020.

[5] Anitha R, Meharuniza Nazeem, Rajeev RR, AnilKumar KS, Comparitive Exploration of Political Opinion Mining Using Machine Learning Techniques. Journal of Information and computational science, Volume 13 Issue, 11 November 2023.

[6] Soumya, S. and Pramod, K.V., 2021, January.Fine-Grained Sentiment Analysis of Malayalam Tweets Using Lexicon Based and Machine Learning Based Approaches. In 2021 4th Biennial International Conference on Nascent Tech nologies in Engineering (ICNTE)(pp. 1-6). IEEE.

[7] Varun Sundaram, Saad Ahmed, Shaik Abdul Muqtadeer, Emotion Analysis in Text using TF-IDF, 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence).

[8] Thulasi, P.K. and Usha, K., 2016, September. Aspect polarity recognition of movie and product reviews in Malayalam. In 2016 International Conference on Next Generation Intelligent Systems (ICNGIS)(pp. 1-5). IEEE.

[9] LSTM, VADER, and TF-IDF-based Hybrid Sentiment Analysis Model, Mohamed Chiny, Marouane Chihab, Younes Chihab, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 7, 2021.

[10] Daniel Jurafsky and James H. Martin, In 2023 January 3.Naive Bayes and Sentiment Classification, Speech and Language Processing. Daniel Jurafsky & James H. Martin. Copyright © 2023.

[11] Nikhil Kumar Singh, Deepak Singh Tomar & Arun Kumar Sangaiah, Sentiment analysis: a review and comparative analysis over social media, Journal of Ambient Intelligence and Humanized Computing volume 11, pages97–117 (2020).

[12] LSTM, VADER, and TF-IDF-based Hybrid Sentiment Analysis Model, Mohamed Chiny, Marouane Chihab, Younes Chihab, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 12, No. 7, 2021.

[13] Soumya S. and Pramod K.V Naive Bayes Support Vector Machine Random Forest, 2020.

[14] Chaudhary Jashubhai Rameshbhai and Joy Paulose, Opinion mining on news paper headlines using SVM and NLP, International Journal of Electrical and Computer Engineering (IJECE) Vol. 9, No. 3, June 2019.

[15] Thulasi P K, Sentiment Analysis in Malayalam, International Journal of Advanced Research in Computer and Communication Engineering IJARCCE, Vol. 5, Special Issue 1, February 2016.

[16] Thulasi, P.K. and Usha, K., September. Aspect polarity recognition of movie and product reviews in Malayalam. In 2016 International Conference on Next Generation Intelligent Systems (ICNGIS)(pp. 1-5), IEEE, 2016.

[17] Soumya, S. and Pramod, K.V., 2021, January.Fine-Grained Sentiment Analysis of Malayalam Tweets Using Lexicon Based and Machine Learning Based Approaches. In 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)(pp. 1-6). IEEE, 2021.

[18] Varun Sundaram, Saad Ahmed, Shaik Abdul Muqtadeer, Emotion Analysis in Text using TF-IDF, 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2021 .

[19] Daniel Jurafsky and James H. Martin, Naive Bayes and Sentiment Classification, Speech and Language Processing. Daniel Jurafsky & James H. Martin. Copyright © 2023.

[20] Sajeetha Thavareesan, Sinnathamby Mahesan, Sentiment Analysis in Tamil Texts: A Study on Machine Learning Techniques and Feature Representation, 2019 IEEE 14th International Conference on Industrial and Information Systems (ICIIS).

[21] Domain-Independent Sentiment Analysis in Malayalam, Computational Intelligence: Theories, Applications, and Future Directions - Volume II (pp.151-160)Edition: 1Chapter: 14Publisher: Springer.

[22] Anitha R, Rajeev R R, K S Anil Kumar and Meharuniza Nazeem, Comparative Exploration of Political Opinion Mining Using Machine Learning Techniques, VOL-13-ISSUE-11-2023, ISSN: 1548-7741.

[23] Deepu S. Nair; Jisha P. Jayan; Rajeev R.R; Elizabeth Sherly, Sentiment Analysis of Malayalam film review using machine learning techniques, 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), **DOI:** 10.1109/ICACCI.2015.7275974

[24] Deepu S. Nair; Jisha P. Jayan; Rajeev R.R; Elizabeth Sherly,Sentiment Analysis of Malayalam film review using machine learning techniques, 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI), **DOI:** 10.1109/ICACCI.2015.7275974

[25] Deepu S. Nair; Jisha P. Jayan; R. R. Rajeev; Elizabeth Sherly, SentiMa - Sentiment extraction for Malayalam, 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), DOI: 10.1109/ICACCI.2014.6968548