Inteligencia Artificial, 28(76) (2025), 85-123
doi: 10.4114/intartif.vol28iss76pp85-123

INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

Explainable Artificial Intelligence Techniques for Speech Emotion Recognition: A Focus on XAI Models

Michael Norval[1,A], Zenghui Wang[1,B]
[1]Department of Electrical Engineering, University of South Africa, Johannesburg
[A]36825050@mylife.unisa.ac.za
[B]wangz@unisa.ac.za

Abstract This study employs Explainable Artificial Intelligence (XAI) techniques, including SHAP, LIME, and XGBoost, to interpret speech-emotion recognition (SER) models. Unlike previous work focusing on generic datasets, this research integrates these tools to explore the unique emotional nuances within an Afrikaans speech corpus. The complexity of architectures poses significant challenges regarding model interpretability. This paper explicitly aims to bridge the gaps in existing Speech Emotion Recognition (SER) systems by integrating advanced Explainable Artificial Intelligence (XAI) techniques. The objective is to develop an Ensemble stacking model that combines CNN, CLSTM, and XGBoost, augmented by SHAP and LIME, to enhance the interpretability, accuracy, and adaptability of SER systems, particularly for underrepresented languages like Afrikaans. Our research methodology involves utilising XAI methods to explain the decision-making processes of CNN and CLSTM models in speech emotion recognition (SER) to enhance trust, diagnostic insight, and theoretical understanding. We train the models for SER using a comprehensive dataset of emotional speech samples. Post-training, we apply SHAP and LIME to these models to generate explanations for their predictions, focusing on the importance of features and the models' decision logic. By comparing the explanations generated by SHAP and LIME, we assess the efficacy of each method in providing meaningful insights into the models' operations. The comparative study of various models in SER demonstrates their capability to discern complex emotional states through diverse analytical approaches, from spatial feature extraction to temporal dynamics. Our research reveals that XAI techniques improve the interpretability of complex SER models. This enhanced transparency builds end-user trust and provides valuable insights. This study contributes to the importance of explainability in deploying AI technologies in emotionally sensitive applications, paving the way for more accountable and user-centric SER systems.

Keywords: Artificial Intelligence, Speech Emotion Recognition, Shapley additive explanations, Local Interpretable Model-agnostic

1. Introduction

This study aims to enhance the interpretability and accuracy of Speech Emotion Recognition (SER) systems by integrating advanced Explainable Artificial Intelligence (XAI) techniques. Specifically, it addresses gaps in cultural and linguistic adaptability by developing models tailored for Afrikaans speech using SHAP, LIME, and XGBoost. Explainable artificial intelligence (XAI) [1] is a field of research that aims to make the decisions and actions of artificial intelligence (AI) systems understandable and interpretable by humans. Speech emotion detection (SED) [2] [3] is a task of speech processing and computational paralinguistics that aims to detect and classify the emotions expressed in spoken language. XAI can be applied to SED to provide insights into how the AI system analyses and classifies speech signals and what features or factors are important for emotion detection. For example, XAI

ISSN: 1137-3601 (print), 1988-3064 (on-line)

can help identify the linguistic and acoustic cues used by the AI system to distinguish between emotions, such as prosody, pitch, and rhythm. XAI can also help evaluate the AI system's performance and reliability and detect and correct any errors or biases in SED. XAI can enhance the trust and confidence of the users and developers of SED systems and facilitate the communication and interaction between humans and AI. This research employs SHAP and LIME to interpret the decision-making processes of SER models, enhancing trust, transparency, and diagnostic insight. Researchers have ventured into utilising XAI in healthcare using Extreme Gradient Boosting (EGB) and an XAI using SHAP [4]. Emotion recognition is explained for Multimodal Emotion Recognition using GradientSHAP [5]. Audio and speech research has been focused on males and females uttering spoken digits and then using the XAI technique of Layer-wise relevance Propagation. These researchers also implemented heatmaps and relevance scores. [6]. Speech emotion recognition wise, researchers used the Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) dataset. The Perceptual Processing Framework and RexNet model are used in XAI. [7]. Some in-depth research into emotions and their representation has recently been published utilising Plutchikâs wheel and multidimensional emotions theory. According to this theory, emotions are different and discrete categories, each comprising cognitive, psychological, and behavioural factors. Emotions can be positive or negative [8]. More recently, models were trained on the Emo-DB and AESSD datasets using an overlapping sliding window (OSW) technique for SER in this study. Finally, the model was evaluated by using the SHAP (SHapley Additive exPlanations) analysis [9]. The potential of XAI in detecting hate speech using deep learning models is versatile and multifaceted. The model is trained on Twitter data and then analysed using LIME, SHAP, XGBoost, and KTrain to investigate the accuracy [10]. While XAI has been utilized in SER, existing models lack diversity in their architectures and the application of explainability techniques, leaving gaps in cultural and linguistic adaptability. This study proposes an innovative framework that integrates Convolutional Long Short-Term Memory (CLSTM) models with XAI techniques such as SHAP, LIME, and XGBoost, specifically tailored for underrepresented languages like Afrikaans. The integration of these techniques provides a comprehensive framework for understanding SER models:

SHAP: Offers detailed global and local explanations of feature contributions. LIME: Focuses on local decision-making for individual predictions, adding interpretability. XGBoost: Strengthens prediction accuracy by modelling non-linear feature interactions and emphasizing feature importance. This methodology addresses interpretability gaps in SER, particularly for underrepresented languages like Afrikaans. In our study, we addressed the analysis of Speech Emotion Recognition (SER) within a novel context, focusing on a custom Afrikaans Speech corpora to explore the nuanced emotional expressions specific to the Afrikaans language. Motivated by the need to address the cultural and linguistic gaps in existing SER research, this study leverages XAI techniques (SHAP, LIME, XGBoost) to build interpretable models optimized for Afrikaans speech, providing a novel perspective on culturally sensitive emotion recognition. XGBoost served as the backbone for our emotion recognition models, providing a robust and high-performing basis for classification. To ensure our models' interpretability and gain insights into which features were most indicative of emotional states, we applied SHAP, which allowed us to quantify the contribution of each speech feature to the model's predictions. LIME complemented this by offering localised explanations, shedding light on how specific segments or characteristics within the Afrikaans speech influenced the detection of emotions. This comprehensive application of XAI techniques enhanced the transparency and understanding of our models. It allowed for a culturally and linguistically informed approach to SER, paving the way for more accurate and meaningful emotion recognition. The main contributions of this paper are:

- **Improving Transparency in SER Models:** This study develops a novel ensemble methodology combining advanced visualization tools (SHAP, LIME) with diverse models (CLSTM, CNN, XGBoost), significantly enhancing transparency and interpretability in SER. By implementing layer-wise relevance propagation and feature attribution methods, we significantly enhance the interpretability of SER systems, making it easier for researchers and practitioners to understand the model's predictions and trust its outputs.
- **Comparative Study of XAI Techniques in SER:** Our research conducts a thorough comparative analysis of various Explainable Artificial Intelligence (XAI) techniques applied to SER models, including Local Interpretable Model-agnostic Explanations (LIME) and SHapley Additive exPlanations (SHAP). We evaluate these techniques on multiple SER architectures to identify the most effective approach for elucidating model rationale, highlighting the strengths and limitations of each method in providing meaningful insights into the emotional content of speech.
- **Explore Ensemble Techniques in SER XAI:** This paper explores applying ensemble techniques in the context of XAI for SER, proposing a novel framework that combines multiple XAI methods to generate comprehensive and robust explanations. By aggregating insights from different explanatory models, our approach addresses the partial and sometimes conflicting interpretations offered by individual XAI techniques, thus providing a more holistic understanding of how SER models process and classify emotional states in speech.

We organised the remaining parts of the paper: Section 2 reviews the existing literature on SER, CLSTM architectures, XAI, SHAP, LIME, Training Data, and Evaluation methods. Section 3 examines the proposed system, data organisation, and training procedures. In Section 4, we present and discuss the experimental results. Section 5 concludes and concludes the article.

2. Related Works

2.1. Speech Emotion Recognition

Speech Emotion Recognition (SER) emerges as a cornerstone in affective computing, a field dedicated to equipping machines with the nuanced ability to detect and interpret human emotions through speech. This sophisticated technological endeavour extends beyond mere voice recognition, delving into the intricate tapestry of tonal variations, speech patterns, and acoustic features that convey emotional states. [11] The implications of SER are profound and multifaceted, touching on a broad spectrum of applications that significantly impact human-computer interaction. In AI-driven personal assistants, SER transforms user experience by fostering more natural, empathetic, and responsive interactions, allowing these digital companions to react to the content of speech and its emotional undertones. [12] Similarly, in therapeutic settings, the application of SER opens new avenues for mental health assessments, offering a non-invasive tool to capture emotional cues that might elude traditional observational methods, thus providing valuable insights into patient well-being. [8] The integration of Explainable Artificial Intelligence (XAI) methodologies, such as SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations), alongside advanced machine learning techniques like XGBoost, propels SER research to new heights. XAI is a critical bridge between AI models' complex, often opaque decision-making processes and the need for transparency and interpretability in applications involving human emotions. By leveraging XAI, researchers and practitioners can unravel the ”black box.ºf SER models, gaining a clear understanding of how and why certain speech features contribute to emotion recognition outcomes. This enhances trust in SER technologies and enables iterative refinement of models based on interpretable evidence, ensuring that SER systems are effective and aligned with ethical standards. Consequently, the synergy between SER and XAI heralds a future where technology comprehensively understands and empathises with human emotions, paving the way for more genuine and meaningful human-computer interactions across diverse domains.

2.2. Convolutional Long Short-Term Memory

Convolutional Long Short-Term Memory (CLSTM) networks represent an advanced neural network architecture that synergises the capabilities of Convolutional Neural Networks (CNNs) [13] [14] and Long Short-Term Memory (LSTMs). CNNs are recognised for their proficiency in identifying local and shift-invariant characteristics within image and audio datasets. Conversely, LSTMs excel in discerning long-term temporal correlations, rendering them highly suitable for analysing time series and sequential data [15] [16]. Within the CLSTM [17] [18] framework, the initial application of CNN layers facilitates extracting intricate spatial features from the input, typically generating comprehensive feature maps. Subsequently, these spatial representations are processed as sequences by the LSTM layer, which meticulously captures the temporal relationships among the features. This innovative architecture has demonstrated notable efficacy in domains necessitating the simultaneous interpretation of spatial and temporal data, such as video classification and time-series forecasting. By amalgamating the spatial analytical strength of CNNs with the temporal modelling capacity of LSTMs, CLSTM networks offer a formidable approach to spatiotemporal feature analysis. The introduction of CLSTM architectures marks a pivotal advancement in deep learning, explicitly addressing the complexities inherent in temporal dynamics and depth comprehension [19][20].

2.3. Explainable Artificial Intelligence

Explainable Artificial Intelligence (XAI) has emerged as a pivotal field within AI research, aimed at making complex machine learning models more transparent and interpretable to humans. The necessity for XAI arises from the increasing deployment of AI systems in critical decision-making processes, where understanding the rationale behind model predictions is essential for trust, compliance, and ethical considerations [21]. Among the various techniques developed for XAI, SHapley Additive exPlanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME) stand out for their effectiveness in demystifying the decision-making processes of AI models [22]. SHAP leverages the concept of Shapley values from cooperative game theory to attribute the contribution of each feature to the prediction made by the model. It offers a consistent and theoretically grounded method to quantify the impact of each feature, providing a detailed decomposition of the prediction. This approach facilitates a deeper understanding of the model's behaviour and helps identify potential biases and areas for improvement [23]. [24][25]. SHAP's ability to provide global insights into the model's overall decision-making patterns and local explanations for individual predictions makes it an invaluable tool for developers and end-users. On the other hand, LIME focuses on explaining individual predictions by approximating the local decision boundary of any black-box model. It perturbs the input data and observes the changes in forecasts, thereby generating an interpretable model that is locally faithful to the original model [26]. This local model, often a linear model or a decision tree, provides insights into which features were most influential for a specific prediction. LIME's model-agnostic nature allows it to be applied across many models, making it a versatile tool for understanding complex model behaviours. Together, SHAP and LIME address the critical need for transparency in AI, enabling stakeholders to gain insights into the workings of sophisticated models. By elucidating the contribution of individual features to model predictions, these XAI techniques enhance the accountability and fairness of AI systems. Their application spans various domains, from healthcare, where understanding diagnostic predictions is crucial, to finance, where reasons for loan approval or denial need to be precise. As AI continues to evolve, the role of XAI, particularly methods like SHAP and LIME, becomes increasingly important in fostering trust and ensuring the responsible use of AI technologies.

2.3.1. SHapley Additive exPlanations

Explainable artificial intelligence (XAI) has become a focal point in advancing the field of machine learning by addressing the opacity of complex models, particularly deep learning architectures. Within this realm, SHapley Additive exPlanations (SHAP) emerges as a robust, theory-driven framework that offers a cohesive approach to understanding model predictions. Rooted in cooperative game theory, SHAP assigns each feature an importance value for a particular prediction, drawing on the Shapley values concept [27]. This method ensures a fair distribution of contribution among features, reflecting the marginal impact of each feature across all possible combinations. The elegance of SHAP lies in its ability to provide both local explanations, which clarify individual predictions, and global insights, which illuminate the overall model behaviour by aggregating individual explanations. Consequently, SHAP facilitates a granular and comprehensive understanding of the model's decision-making process, enhancing transparency and trust. Its application spans various domains, enabling practitioners to decode complex model predictions, identify potential biases, and foster more interpretable, accountable, and fair AI systems [28]. The significance of SHAP in the XAI landscape underscores the growing demand for methodologies that bridge the gap between high-performing models and the imperative for their decisions to be interpretable and justifiable within real-world applications [29]. SHapley Additive exPlanations (SHAP) values are derived from game theory, specifically from Shapley values. The Shapley value is a way to distribute the "payout fairly"(in this case, the prediction of the model) among the "players"(in this case, the features of the model) based on their contribution to the "game"(the prediction task). The mathematical formula for SHAP values integrates this concept into the machine learning context to explain the contribution of each feature to predicting a particular instance [30]. The formula for calculating the Shapley value for a feature in a prediction model can be expressed as:

$$\phi_j = \sum_{S \subseteq N \setminus \{j\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [f(S \cup \{j\}) - f(S)]$$

Here:

- ϕ_j is the SHAP value for the feature j.
- N is the set of all features.
- S is a subset of features excluding j.
- $|S|$ is the number of features in the subset S.
- $|N|$ is the total number of features.
- $f_s(S \cup \{j\})$ is the prediction when the features in set S and feature j are included.
- $f_s(S)$ is the prediction when only the features in set S (excluding feature j) are included.
- The sum is taken over all possible subsets S of the set N, excluding the feature j.

This formula calculates the average marginal contribution of feature j across all possible combinations of features. The factorial terms $\frac{|S|!(|N|-|S|-1)!}{|N|!}$ serve as weights for the contributions, ensuring that the contributions of feature j are moderately averaged over the number of features in subset S and the total number of features.

The SHAP value ϕ_j quantifies the impact of having feature j present or absent in making the model's prediction compared to the average prediction across all feature combinations. This approach provides insight into how individual features influence the model's output and ensures a fair and consistent distribution of importance among all features, adhering to the properties of efficiency, symmetry, dummy, and additivity derived from cooperative game theory [31]. Types of explainers: Tree Explainer: This tool is optimised for tree-based models like decision trees, random forests, gradient boosting machines (GBM), and XGBoost. It's highly efficient and accurate for these types of models. Deep Explainer: Designed for deep learning models. It approximates SHAP values by leveraging the structure of deep networks to make calculations more tractable. Kernel Explainer: This model-agnostic explainer works well with any machine-learning model. It uses a specially weighted linear regression to approximate SHAP values. It is slower than tree or deep explainers but offers flexibility. Linear Explainer: This tool is tailored for linear models. It computes SHAP values based on the coefficients of the linear model, providing precise explanations for models where the prediction is a linear function of the inputs. Gradient Explainer: This is like Deep Explainer but designed explicitly for gradient-based optimisation models. It's beneficial for TensorFlow or PyTorch models, where gradients can provide insights into the importance of features. Partition Explainer: A newer addition that offers a fast and accurate way to compute SHAP values for any model by partitioning the input space. This explainer is handy for complex models where other explainers may not be efficient. Types of plots: Summary Plot, Force Plot, Dependence Plot, Decision Plot, Waterfall Plot, Beeswarm Plot, Scatter Plot, Bar Plot, Heatmap Plot, and Image Plot.

The table below Table 1 shows which plot is used with which explainer.

Explainer	Tree Explainer	Deep Explainer	Kernel Explainer	Linear Explainer	Gradient Explainer	Partition Explainer
Summary Plot	X	X	X	X		
Force Plot	X	X	X	X		
Dependence Plot	X		X			
Decision Plot	X					
Waterfall Plot	X					
Beeswarm Plot	X					
Scatter Plot	X					
Bar Plot	X					X
Heatmap Plot	X					X
Image Plot					X	

Cuadro 1: SHAP Explainer / Plot compatibility chart.

The vital plot used in this study is the force Plot. In terms of colour indicators: Red: Features that push the prediction higher are shown in red. The length of the red segment or arrow represents the magnitude of the feature's positive impact on the model's output compared to the baseline value. In other words, red features contribute to increasing the prediction value from the base value (average model output over the dataset). Blue: Conversely, features that pull the prediction lower are shown in blue. The length of the blue segment or arrow indicates the strength of the feature's negative impact on the prediction. Blue features, therefore, contribute to decreasing the prediction value from the base value.

2.3.2. LIME

Local Interpretable Model-agnostic Explanations (LIME) is a groundbreaking technique within Explainable Artificial Intelligence (XAI) that addresses the imperative need for transparency and comprehensibility in complex machine learning models [32]. Developed to elucidate the decision-making processes of otherwise opaque models, LIME operates on the principle of local approximation. It ingeniously generates interpretable models that mimic the predictions of the original model within a local vicinity of the input being explained [33]. By perturbing the input data and observing the resultant variations in output, LIME deduces the significance of various features in influencing the model's prediction for a specific instance. This localised approximation approach allows LIME to provide intuitive explanations regardless of the original model's complexity or type, making it agnostic to model architecture [34]. Consequently, LIME has facilitated a broader understanding and trust in AI applications across diverse sectors, including healthcare diagnostics, financial services, and legal adjudication, by enabling stakeholders to understand the rationale behind individual predictions. Its versatility and efficacy in demystifying machine learning predictions advocate for its essential role in developing and deploying responsible AI systems [35]. Formula: The mathematical formulation of LIME can be described in a simplified manner as follows:

$$\xi(x) = \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

Where:

$\xi(x)$ is the objective function to be minimised $\mathcal{L}(f, g, \pi_x)$ represents LIME generating a new dataset of perturbed samples around x' and obtains the predictions of f for these samples $\Omega(g)$ is a measure of complexity of the interpretable model g such as a linear model or decision tree f is the complex model for which we want to explain the prediction x is an instance for which we want to explain the prediction

$$\mathcal{L}(f, g, \pi_x) = \sum_{z,z' \in Z} \pi_x(z)(f(z) - g(z'))^2 \tag{2}$$

Where:

$\mathcal{L}(f, g, \pi_x)$ is a measure of the fidelity of g in approximating f in the locality defined by π_x Z is the set of perturbed samples generated around x, with their corresponding labels z' predicted by f $f(z)$ is the prediction of the complex model on the perturbed sample z $g(z')$ is the prediction of the interpretable model on z' $\pi_x(z)$ is a proximity measure that weights the importance of each perturbed sample z based on its similarity to x, ensuring that the explanation is locally faithful around x

Types of plots: Feature Importance Plot: This is the most common type of plot used in LIME. It displays each feature's contribution to the prediction for an individual instance. The plot typically shows a list of features and their corresponding weights, indicating how each feature contributes positively or negatively to the prediction. Decision Boundary Plot: LIME can visualise the decision boundary around an instance in a simplified feature space in classification tasks. This can be particularly insightful for understanding why an instance was classified in a particular way based on its proximity to the decision boundary in the local, simplified model. Text Explanation: For text data, LIME highlights the parts of the text (such as words or phrases) that are most influential in the model's prediction. Positive influences may be highlighted in one colour (often green), while negative influences may be shown in another (often red). Image Explanation: LIME can be used for image data by segmenting the image and determining which segments (superpixels) have the most impact on the model's prediction. Influential segments are typically highlighted, contributing to the model's decision. Tabular Data Visualization: When dealing with tabular data, LIME provides bar charts or similar visualisations that show the contribution of each feature to a particular prediction. It's like the feature importance plot, but tailored for tabular data, where each feature is a column in the dataset. Interactive Web Interface: LIME also offers an interactive web interface that allows users to explore different features' contributions to predictions. This is particularly useful for iterative analysis and non-technical stakeholders who must understand model decisions. The Feature Importance Plot and Tabular Plot were chosen because they are best suited for audio data.

2.3.3. XGBoost

XGBoost, an abbreviation for eXtreme Gradient Boosting, has emerged as a powerful and efficient implementation of gradient boosting machines, a class of ensemble machine learning algorithms that has gained substantial traction for its predictive accuracy in various domains. While XGBoost is renowned for its performance and speed, its integration with Explainable Artificial Intelligence (XAI) principles is exciting [36]. XAI aims to make the decisions made by machine learning models understandable to humans, addressing the "black box"nature of many sophisticated algorithms. This summary focuses on the significance of XGBoost within the context of XAI and highlights relevant academic insights and contributions, including references from IEEE and other scholarly sources [37][38]. The core algorithm of XGBoost builds decision trees sequentially, where each new tree corrects errors made by the previously trained trees. While the ensemble of decision trees can achieve high accuracy, interpreting the combined predictions of many trees can be challenging. This complexity underscores the importance of applying XAI techniques to XGBoost models, enabling users to understand and trust the model's decisions, a crucial aspect in sensitive applications like healthcare, finance, and criminal justice [39][40]. Types of plots: Feature Importance Plot: This plot ranks the features based on their importance to the model's predictions. XGBoost provides several ways to measure feature importance, including weight (the number of times a feature appears in a tree), gain (the average gain of splits that use the feature), and cover (the average coverage of splits that use the feature). Decision Tree Visualization: Given that XGBoost is a tree-based ensemble method, visualising individual decision trees can provide insights into how specific features influence the model's decisions. Tools like xgboost.plot.tree can generate graphical representations of the trees within the model. Partial Dependence Plots (PDPs): PDPs show the relationship between a feature and the predicted outcome, keeping other features constant. This helps in understanding the marginal effect of a feature on the prediction. While not built directly into XGBoost, PDPs can be generated using libraries like pdpbox or scikit-learn. SHAP Values (SHapley Additive exPlanations): SHAP values explain the contribution of each feature to each prediction, providing a more nuanced understanding of model behaviour. XGBoost supports SHAP value calculation directly through its API (get.score(importance.type='gain')). The SHAP library can generate SHAP summary plots, SHAP dependence plots, and SHAP force plots to visualise these values. Tree SHAP: Tree SHAP is a specific method optimised for tree-based models, including XGBoost, to compute SHAP values efficiently. It offers precise explanations of model pre-dictions individually and allows for detailed interpretation of complex models. Global Surrogate Models: A global surrogate model is a simpler, interpretable model (like a decision tree) that approximates the predictions of the complex XGBoost model. By training the surrogate on the projections of the XGBoost model, you can gain insights into the overall model behaviour. LIME (Local Interpretable Model-agnostic Explanations): LIME can be applied to instances classified by an XGBoost model to identify which features most strongly influence the model's prediction for that instance. While LIME provides local explanations, it can be used iteratively to understand model behaviour more broadly. The Feature Importance Plot and Partial Dependence Plots were chosen because they best suit audio data.

2.3.4. Ensemble

Ensemble techniques in machine learning combine multiple models to improve prediction accuracy, robustness, and generalisation over individual models. Stacking, a specific ensemble method, involves training a new model to aggregate the predictions of several base models. In stacking, base models are trained on the complete dataset, and their predictions form the input features for a higher-level model (meta-model) that aims to correct or enhance the base models' predictions. This layered approach leverages the strengths of various models, aiming for better performance by learning how to combine their predictions best.

2.3.5. Training Data

Introduced by psychologist Robert Plutchik in 1980, the Wheel of Emotions is a foundational framework for delineating and categorising human emotional states [41]. Central to Plutchik's model are eight primary emotional dimensions, delineated as opposing pairs: joy versus sadness, trust versus disgust, fear versus anger, and surprise versus anticipation [42]. This schema posits that these primary emotions, through their interplay, can give rise to a spectrum of more nuanced emotional experiences. The corpus employed in this investigation derives from the eight emotions delineated within Plutchik's Wheel of Emotions, consisting of a bespoke Afrikaans speech corpora [43]. Approximately 100 samples have been allocated per emotional category, culminating in a dataset of around 800 samples. This study focuses on extracting and analysing specific acoustic features, namely Mel-frequency cepstral coefficients (MFCC), chromatograms (Chroma), Mel-frequency cepstrum (Mel), Contrast, and the German Tone Network (Tonnetz), to train the models. Figure 1 visually represents a singular training sample, showcasing the extracted features [44] [11].

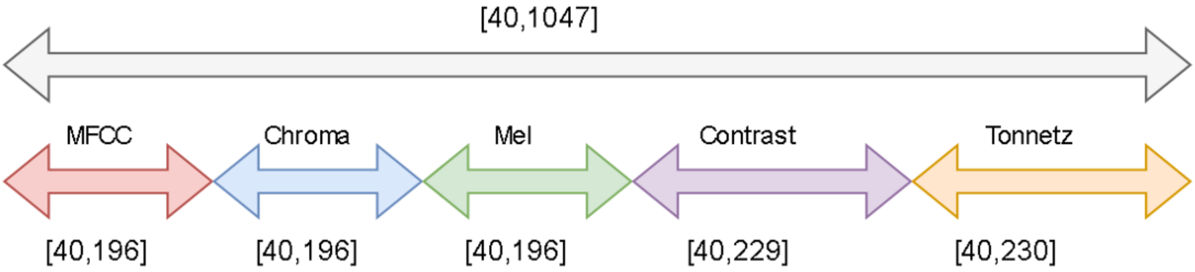


Figura 1: Extracted features

2.4. Evaluation methods

Evaluating Explainable Artificial Intelligence (XAI) methods like SHAP, LIME, and XGBoost involves assessing how effectively these techniques elucidate model decisions to users. The evaluation can encompass various dimensions, including interpretability, fidelity, and usability. Hereâs a summary of how evaluation methods apply to SHAP, LIME, and XGBoost:

2.4.1. SHAP

Interpretability: SHAP values provide detailed insights into each feature's contribution to the model's prediction for individual instances, offering high interpretability. Evaluation often involves qualitative analysis, where human experts assess the relevance and intuitiveness of the explanations. Fidelity: High fidelity is crucial for SHAP, as its explanations aim to reflect the model's behaviour accurately. Fidelity can be evaluated quantitatively by comparing the model's output to the output predicted by the SHAP values across various instances. Usability:

Usability assessment might involve user studies to determine how well practitioners and end-users understand and can act upon the SHAP-generated explanations.

2.4.2. LIME (Local Interpretable Model-agnostic Explanations)

Interpretability: LIME focuses on local interpretability, offering explanations for individual predictions. Its interpretability is evaluated based on how well the local, simplified models it generates (e.g., linear models for classification) align with users' intuitive understanding of the decision process. Fidelity: LIME's fidelity is assessed based on how closely the explanations from the simplified model match the complex model's decisions in the vicinity of the instance being explained. This can involve measuring the discrepancy in predictions over perturbations of the input data. Usability: Like SHAP, LIME's usability evaluations may include user studies to gauge the clarity of its explanations and the ease with which users can use them to make decisions.

2.4.3. XGBoost (eXtreme Gradient Boosting)

Interpretability: While XGBoost is primarily an algorithm for gradient boosting, it incorporates features that facilitate model interpretation, such as feature importance scores. Evaluating interpretability might involve assessing whether the identified important features make sense given domain knowledge. Fidelity: For XGBoost, fidelity in the context of explanations relates to how well the feature importance scores and other interpretability tools it offers reflect the model's underlying decision-making process. This could be evaluated through ablation studies, where the impact of removing or altering features based on their importance is observed.

3. METHODOLOGY

3.1. GAPS IN EXISTING SOLUTIONS

Despite significant advancements in Explainable Artificial Intelligence (XAI) across diverse fields such as healthcare, emotion recognition, and hate speech detection, notable gaps in existing solutions warrant further exploration. In healthcare, while the use of Extreme Gradient Boosting (EGB) and SHAP has shown promise, the intricate nature of medical data and the need for highly reliable interpretations highlight a gap in developing universally applicable models that can adapt to the varying complexities of medical diagnostics and patient data. This study diverges from traditional approaches by employing SHAP and LIME to interpret SER models trained on a linguistically and culturally specific Afrikaans corpus. XGBoost enhances this unique application by providing robust classification. At the same time, the integration of SHAP and LIME offers interpretability tailored to the nuanced emotional expressions captured in Afrikaans speech. Yet, the research predominantly focuses on binary gender distinctions in audio and speech, potentially oversimplifying the rich, nuanced spectrum of human vocal expressions. Furthermore, the application of XAI techniques like SHAP and LIME in analysing complex datasets like RAVDESS, Emo-DB, and AESSD for speech emotion recognition (SER) underscores a gap in addressing the multidimensional aspects of emotions as posited by Plutchikâs wheel. While these methods offer insights into the cognitive, psychological, and behavioural dimensions of emotions, there's a noticeable void in effectively mapping these intricate emotional categories onto XAI outputs in a manner that's both comprehensive and intuitively understandable for end-users. Lastly, the versatility of XAI in detecting hate speech using deep learning models trained on Twitter data reveals a multifaceted challenge not only in model accuracy but also in the interpretability and applicability of explanations across diverse linguistic and cultural contexts, indicating a broader gap in the cross-cultural generalisability of current XAI methods.

3.2. JUSTIFICATION FOR CURRENT RESEARCH

The burgeoning interest in Explainable Artificial Intelligence (XAI) within diverse and critical domains such as healthcare, emotion recognition, and online content moderation under-scores the justification for the current research in deploying advanced XAI methodologies, including SHAP, LIME, and Extreme Gradient Boosting (XGBoost). Specifically, integrating XGBoost alongside SHAP in healthcare settings demonstrates a pivotal shift towards enhancing interpretability in complex predictive models, aiming to bolster trust and transparency in medical decision-making processes. Similarly, the adoption of GradientSHAP for multimodal emotion recognition and the utilisation of Layer-wise Relevance Propagation for deciphering audio and speech patterns signify strides towards understanding the intricate layers of human emotions and behaviours through AI, leveraging datasets such as RAVDESS to explore the multifaceted nature of emotional expressions. Moreover, the recent application of SHAP in evaluating models trained on the Emo-DB and AESSD datasets for Speech Emotion Recognition (SER) and the employment of LIME in conjunction with SHAP and XGBoost for analysing hate speech detection accuracy on Twitter data highlight the versatility and the critical need for explainable models in navigating the ethical and social implications of AI applications. These research endeavours not only bridge the gap between complex data patterns and human interpretable insights but also lay the groundwork for developing AI systems that are both powerful and understandable, thereby ensuring their ethical and effective integration into society.

3.3. PROPOSED SYSTEM

The workflow depicted in the image outlines a process for analysing speech emotion recognition using various neural network architectures and applying explainable artificial intelligence (XAI) techniques to interpret the models. Here's a description of the workflow steps: Begin: The process starts. Key audio features (e.g., MFCC, Chroma) were extracted to train CNN, CLSTM, and XGBoost models, with ensemble stacking employed to integrate their strengths. Model Training and Testing: CNN (Convolutional Neural Network): Train and test a CNN model on the extracted features for baseline comparison. If the number of epochs during training exceeds 50, continue training; otherwise, apply XAI methods. CLSTM (Convolutional LSTM): Train and test a CLSTM model and apply XAI if the number of epochs reaches 50. XGBoost: Implement an Extreme Gradient Boosting model, a tree-based ensemble machine learning algorithm. Ensemble Stacking: Combine the predictions from the individual models (CNN, CLSTM, and XGBoost) using an ensemble stacking technique to improve the overall performance. Application of XAI Techniques: SHAP Deep Explainer: Apply SHAP Deep Explainer to the CNN and CLSTM models to interpret the model predictions. SHAP (SHapley Additive exPlanations) provides a way to explain the output of deep learning models by computing the contribution of each feature to the prediction. LIME: Apply LIME (Local Interpretable Model-agnostic Explanations) for both CNN and CLSTM to obtain interpretable explanations for individual predictions. SHAP Tree Explainer: Use SHAP Tree Explainer specifically for the XGBoost model, which is suitable for explaining tree-based model predictions. Comparison and Evaluation: Compare and evaluate all results from the different models, including the ensemble stacking approach and the applied XAI techniques. End: The process concludes. The workflow indicates a structured approach to build, evaluate, and interpret various types of models used for speech-emotion recognition. It emphasises the importance of model interpretability in affective computing, mainly how AI systems analyse and understand emotions in audio. By employing CNN, LSTM, and ensemble stacking architectures and improving model insights with SHAP and LIME, the workflow reflects an effort to overcome some of the limitations identified in the literature, such as capturing long-term dependencies and dealing with variable audio quality. Using a custom Afrikaans speech corpora suggests a targeted approach to understanding the emotional context within a specific language, enhancing the potential for tailored and accurate emotion detection.

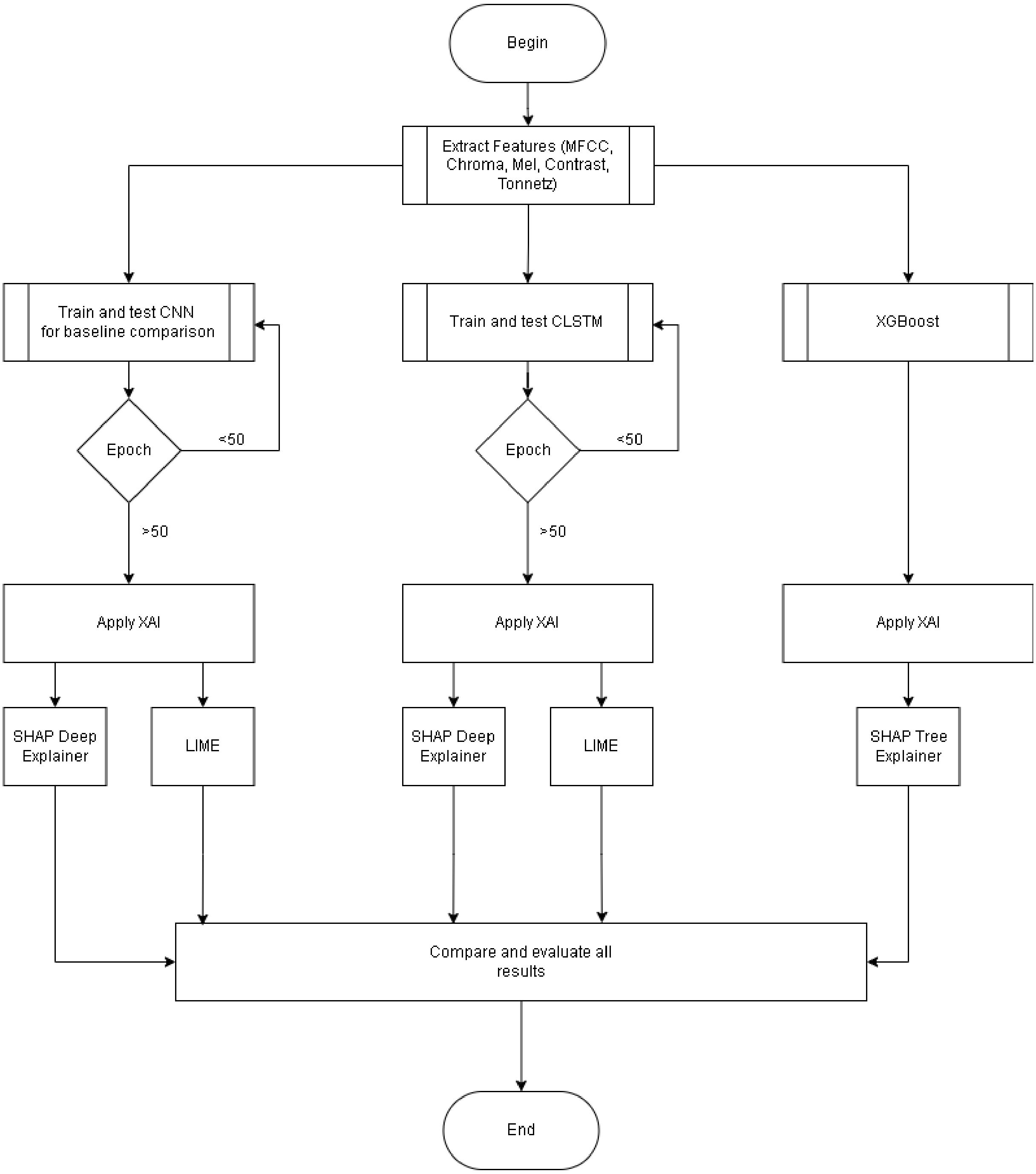


Figura 2: Proposed System Diagram

3.4. DATASET PREPARATION

The dataset is organised into eight distinct emotion categories, each aligning with the classifications delineated in Plutchik's Wheel of Emotions.

The eight emotions are Anger, Anticipation, Disgust, Fear, Joy, Sadness, surprise and Trust.

This compilation is systematically distributed across eight subdirectories containing 100 audio file samples. These audio files undergo a rigorous processing phase, during which several key features are extracted: mel-frequency cepstral coefficients (MFCC), Chromagrams (Chroma), Mel Spectrograms (Mel), Spectral Contrasts, and the Tonal Centroid features (Tonnetz). The dataset is partitioned for model training and validation following a stratagem that allocates 20 % of the samples to validation, with the remainder designated for training. The data structure presented to the neural networks for processing adheres to the dimensions (798, 1047, 40, 1), denoting a compilation of 1047 data points, each characterised by 40 coefficients across a singular dimension. This structured approach to dataset organisation and preprocessing facilitates a rigorous examination of emotional cues within audio samples, enabling the nuanced training and validation of the subsequent neural network models.

3.5. TRAINING PROCEDURE

All models were trained using Google Colab in a High RAM V100 GPU environment with TensorFlow Keras (v2.x), Numpy, and Scikit-learn libraries. The training data underwent preprocessing, including normalization of MFCC values and augmentation techniques like pitch shifting and time-stretching to increase data variability.

CNN and CLSTM: Both models were trained with a learning rate of 0.001, batch size of 32, and 50 epochs, using the Adam optimizer. Regularization included dropout layers (rate = 0.5) to prevent overfitting. The CLSTM captured temporal dependencies in audio, while CNN focused on spatial feature extraction.

XGBoost: Input data was flattened into a 2D structure, and hyperparameter tuning was conducted using a grid search. The final configuration included 100 estimators, a learning rate of 0.1, and max depth of 6. SHAP values were calculated to ensure model interpretability.

The Ensemble stacking model combined predictions from CNN, CLSTM, and XGBoost using a meta-classifier trained on their outputs. The ensemble was fine-tuned using validation accuracy as a metric, and early stopping was applied with a patience of 10 epochs.

Results were evaluated on unseen test data with accuracy, precision, recall, and F1-score as performance metrics. Visualization techniques (SHAP and LIME) were used post-training to explain model decisions.

3.6. MODELS

The two models used in this study are CNN and CLSTM. Convolutional Neural Networks (CNNs) and Convolutional Long Short-Term Memory Networks (CLSTMs) are specialised types of deep learning models designed to process data with grid-like topology, such as images and sequential data, respectively. While CNNs excel in tasks like image recognition and classification through their ability to capture spatial hierarchies in data, CLSTMs extend this capability to sequential data, incorporating memory units that capture temporal dependencies for applications in video analysis and natural language processing.

3.6.1. CNN

The model described is a Convolutional Neural Network (CNN) tailored for processing sequence data, primarily focused on utilising a 1D convolution layer to extract temporal features from the input. The critical components of this model, based on the provided code snippet and following the requested format, are as follows: The model initiates with a 1D Convolution layer that has 128 filters, a kernel size of 5 and uses the 'relu' activation function, designed to process input shaped as (numfeaturesCNN, numtimestepsCNN). This layer aims to capture local dependencies and feature patterns within the sequence data. After convolution, the model flattens the output to convert it from a multidimensional output to a 1D vector, preparing it for dense layer processing. The flattened data then feeds into a Dense layer with num.classes.CNN units and a 'softmax' activation function responsible for classification across num.classes.CNN possible outcomes. The model is compiled with a Categorical Cross-entropy loss function and an Adam optimiser, indicating its use for multiclass classification problems. The training is executed with a batch size of 32 and 30 epochs without explicit verbosity, which means the model's performance metrics during training are not printed out to the console.

3.6.2. CLSTM

The CNN LSTM hybrid CLSTM model consists of the following layers. Data is fed into a 1D Convolution layer with 128 filters and a kernel size of 5, using 'relu' as the activation function, designed to process input shaped as (numfeaturesCLSTM, numtimestepsCLSTM). This is immediately followed by a Max Pooling layer with a pool size of 2, which reduces the dimensionality of the data, helping to prevent overfitting by abstracting higher-level features. Subsequently, a Dropout layer with a rate of 0.5 reduces overfitting by randomly ignoring a subset of neurons during training. The Dropout layer feeds into an LSTM layer with 64 units, which returns sequences to capture temporal relationships within the data. This sequence output is then flattened, converting it from a multi-dimensional tensor to a 1D vector, making it suitable for the dense layer. The flattened production is passed to a Relu-activated Dense layer with 32 units, adding abstraction. Finally, another Dropout layer with a rate of 0.5 is applied before the output layer. The output layer is a Dense layer with num.classes.CLSTM units and a 'softmax' activation function suit multi-class classification tasks. The model is compiled with a Categorical Cross-entropy loss function and an Adam optimiser, indicating its suitability for classification problems with more than two classes. The model is trained using a batch size of 32 and 30 epochs. This training regime suggests a moderate training duration to balance underfitting and overfitting.

3.6.3. Gradient Boost

The steps used to produce the XGBoost visuals are: Data Loading: The data is split into training and testing sets for both features (xtrain, xtest) and labels (ytrain, ytest), with the number of labels (numlabels) mentioned. The 3D data is flattened into 2D (xtrainflat, xtestflat), which is necessary for XGBoost, which cannot handle 3D input directly.

Data Processing: The labels are onehot encoded using tocategorical, typical for multiclass classification tasks in neural networks. It captures the input shape (inshape) and dimensions (numtimesteps, numfeatures) from the training data, which will likely be used in model architecture.

XGBoost Model: The code comments out the initialisation and training of an XGBClassifier but shows loading a pretrained model from a file. This suggests that the model was trained separately or previously and is now being reused for explanation or further analysis. uselabelencoder=False and evalmetric='mlogloss' are specified for the model, which are configurations to handle label encoding internally and use log loss as the evaluation metric, suitable for classification tasks. SHAP Explanation: Initializes a TreeExplainer with the XGBoost model for computing SHAP values, which quantify the impact of each feature on the model's predictions. Computes SHAP values for the first 100 instances of the flattened training data, which is used to understand model decisions on this subset. Generates a SHAP summary plot for the same subgroup from the flattened testing data, using a bar plot to show the average impact of each feature. Initializes JavaScript for SHAP visualisation (shap. init ()) and creates a forced plot for the first instance to visualise the contribution of each feature to the model's prediction. However, there's a discrepancy in the force plot comment; it should visualise data consistent with shapvalues computation (either from training or testing set).

3.6.4. ENSEMBLE - STACKING

Data Loading: The data is optionally loaded from files using the loaddata function, which checks for the loadfromfile flag. If set to True, it loads training and testing datasets (xtrain, xtest, ytrain, ytest) along with the number of labels (numlabels) from specific joblib files. Otherwise, it presumably fetches and dumps this data into joblib files for future use. Data Processing: Labels (ytrain, ytest) are one-hot encoded using utilstocategorical from Keras, adapting the data for multi-class classification tasks daily in neural networks. Training and testing feature sets (xtrain, xtest) are reshaped to fit the input shape the neural network models require. This involves changing their shape to 3D, aligning with the Conv2D layer's expectations in TensorFlow/Keras models. It captures the input shape (inputshape) and dimensions (numtimesteps, numfeatures) directly from the reshaped training data, which are crucial for defining the neural network architecture. Model Creation and Compilation: An ensemble model, stackedmodel, is created by concatenating the outputs of two models (CNN and ConvLSTM) and passing them through a dense layer with softmax activation for classification. This stacked model is compiled with the Adam optimiser and categorical cross entropy loss, which are standards for multi-class classification problems. The model targets accuracy as its performance metric. Model Training: The stackedmodel is trained on the reshaped and one-hot encoded data (xtrain, ytrain) with a specified batch size and number of epochs. Validation uses the reshaped and encoded testing data (xtest, ytest). Key Observations: The process involves significant preprocessing, including optional data loading from files, onehot encoding of labels, and reshaping of feature sets to fit the neural network model requirements. An ensemble approach is utilised, combining features learned by CNN and ConvLSTM models to make predictions. This could leverage the strengths of both models: CNN's ability to capture spatial hierarchies in data and ConvLSTM's ability to understand sequence dynamics. The model is prepared for multiclass classification using softmax activation in the output layer and categorical cross-entropy as the loss function. Model Evaluation and Prediction: Model Evaluation: The model's performance is evaluated on the test set (x.test, y.test) to obtain loss and accuracy metrics. Prediction and Classification: Predictions are made on the test set, and the predicted class for each instance is determined by finding the index of the maximum value in the prediction arrays. The actual classes are similarly extracted from the onehot encoded ytest.

3.7. XAI

SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) are two prominent methodologies in the domain of explainable AI (XAI) that aim to provide insights into the decision-making processes of complex machine learning models. SHAP leverages game-theoretic approaches to offer global interpretability by calculating the contribution of each feature to the prediction of every instance. In contrast, LIME provides local interpretability by approximating the model locally around the prediction. XGBoost, a highly efficient gradient boosting framework, is often paired with these interpretability tools to enhance the transparency and understanding of its predictions, making it a powerful combination for both predictive performance and insight into model decisions.

3.7.1. SHAP

The steps used to create and display SHAP results are: Initialize SHAP DeepExplainer: The SHAP DeepExplainer is created for deep learning models like CNNs. It's initialised with two arguments: the trained model (modelCNN) and a subset of the training data (xtrainCNN[:196]). This subset acts as a reference or background dataset to help the explainer understand the data distribution and how the model behaves. Compute SHAP Values: The SHAP DeepExplainer computes the SHAP values for the same subset of the training data used to initialise the explainer. SHAP values quantify the contribution of each feature to the prediction for each instance in the subgroup. The output, shapvaluesCNNDeep, is structured to provide SHAP values for each class (in classification tasks) for each sample. Visualise the Explanation with a Force Plot for a Specific Prediction: The Shap forceplot visualises the impact of features on a single prediction. This particular call visualises the SHAP values for the second class ([1] after expectedvalue and the start of shapvaluesCNNDeep) for the second instance in the subset ([1] after the first index of shapvaluesCNNDeep) explainerCNNDeep.expectedvalue[1] indicates the base value for the second class, which is the model output's expectation before observing the current input features shapvaluesCNNDeep[1][1,:] selects the SHAP values for the second class and the second instance, showing how each feature shifts the prediction from the base value.

Generate a Summary Plot for the First Class Across a Single Instance: The Shap summary plot is designed to provide an overview of the feature's importance and effects across many instances. However, in this call, it seems there's an intention to focus on the first class ([0] after shapvaluesCNNDeep) and the first instance in the dataset (xtrainCNN[0, :]), which is not typical usage for summaryplot. Usually, summaryplot expects a matrix of SHAP values for all samples, not a single instance, and the entire dataset or a significant subset as the second argument for feature values to illustrate the distribution and impact of each feature across all observations.

3.7.2. LIME

The steps used to produce the LIME visual are: Reshape 3D Data to 2D: The CNN model initially takes 3D data (e.g., (798, 40, 1047) representing 798 samples, each with 40 features over 1047 time steps). LIME, however, requires 2D data. The code reshapes this 3D data into 2D format (798, 41880), where each instance is flattened, turning the spatial/temporal structure into a long vector of features. Initialize LIME Tabular Explainer: The LIME tabular [LimeTabularExplainer] is created for tabular (2D) data. The reshaped data xtrainflat serves as the training data for the explainer. Feature names are generated dynamically based on the reshaped data's width, and classnames are provided to label the output classes. The mode is set to 'classification', indicating the CNN model's task. Define a Prediction Function for the CNN Model: Use the modelpredict function to wrap the CNN model's prediction functionality, allowing predictions on reshaped (flattened) data. It reshapes the flattened input to the original 3D shape (40, 1047) that the CNN model expects and then returns its predictions. Select an Instance and Explain the Prediction: Using instanceindex, an instance is selected from the flattened training set. LIME's explainer instance method is then used to explain this specific instance, detailing which features (in the flattened space) most influence the model's prediction. The numfeatures parameter limits the explanation to the top N features affecting the prediction. Display the Explanation: The explanation is displayed directly in a Jupyter Notebook using exp.showinnotebook(showtable=True, showall=True). This visual representation shows the contribution of each selected feature towards the prediction for the chosen instance, offering insights into the model's behaviour.

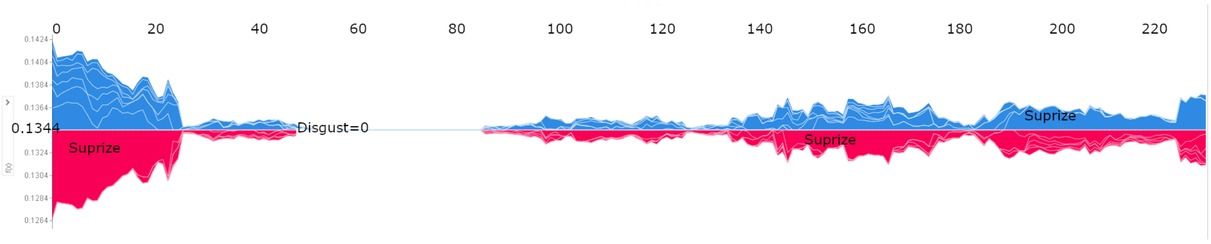


Figura 3: CNN SHAP Force Plot All Data

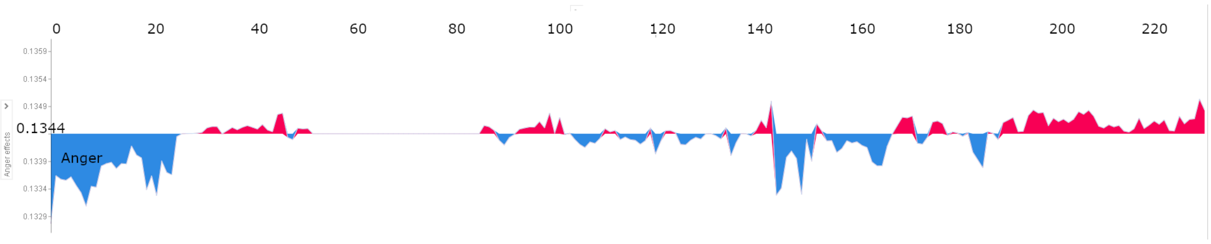


Figura 4: CNN SHAP Force Plot Anger

4. EXPERIMENTAL RESULTS

4.1. TEST DATA

The test dataset is constituted by a selection of samples, randomly extracted from an aggregate of 800 audio clips, to underpin a comprehensive evaluation of the model's capacity for generalisation [43]. A total of 160 clips have been meticulously chosen to serve the dual purpose of validation and testing, thereby facilitating a rigorous assessment of the model's performance across unseen data. This methodological approach ensures the integrity of the evaluation process, reinforcing the reliability of the findings about the model's generalisation capabilities.

4.2. CNN

4.2.1. SHAP

The SHAP force plots for the Speech Emotion Recognition model show the contribution of input features to the prediction of each of the eight emotional categories: Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise, and Trust. A summary of the SHAP summary plots is as follows: The features influencing the prediction of anger show a mix of positive and negative SHAP values, with some features strongly indicating the presence of anger while others are firmly against it. Features that predict anticipation have a balance of positive and negative contributions but are less pronounced than in anger. The SHAP values for disgust also show positive and negative contributions, with some features strongly influencing the prediction. Fear: The prediction of fear is influenced by several features with strong negative SHAP values, indicating features that drive the prediction away from fear. Features related to joy show a varied impact with positive and negative SHAP values, suggesting a complex interplay of features for this emotion. Like other emotions, sadness is predicted by a range of feature contributions, with some having a strong negative impact. For surprise, the SHAP plot indicates both solid positive and negative contributions from the features. Trust predictions are also driven by positive and negative contributing features, some with significant impact. Across all categories, the features have varying levels of influence, with some features being strong indicators or strong negators of specific emotions. The width of the SHAP value distributions across the plots suggests the degree of consensus among features towards predicting a particular emotion. Some plots show a few features with large magnitude SHAP values, indicating these features are critical drivers for the model's decision in predicting the respective emotion. Each emotion has a distinctive pattern of SHAP values, which suggests that the model uses various cues from the speech data to distinguish between emotions. All the abovementioned plots are shown below.

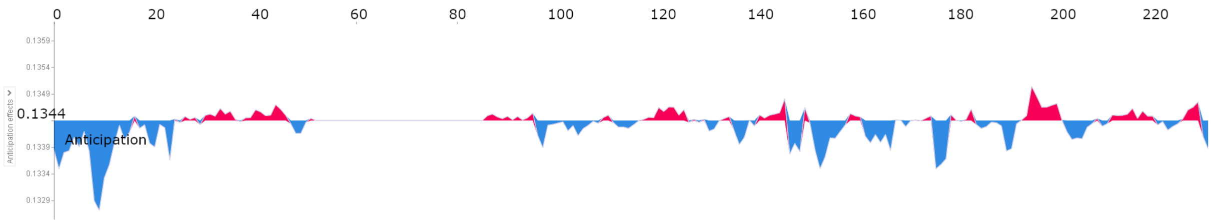


Figura 5: CNN SHAP Force Plot Anticipation

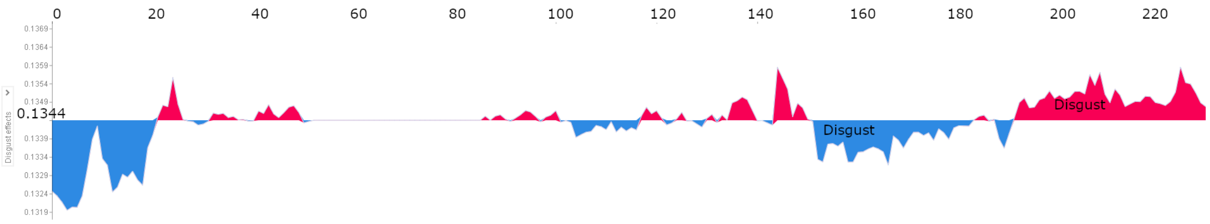


Figura 6: CNN SHAP Force Plot Disgust

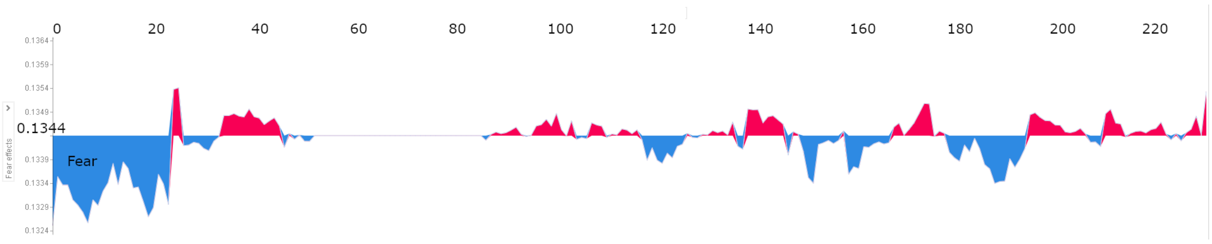


Figura 7: CNN SHAP Force Plot Fear

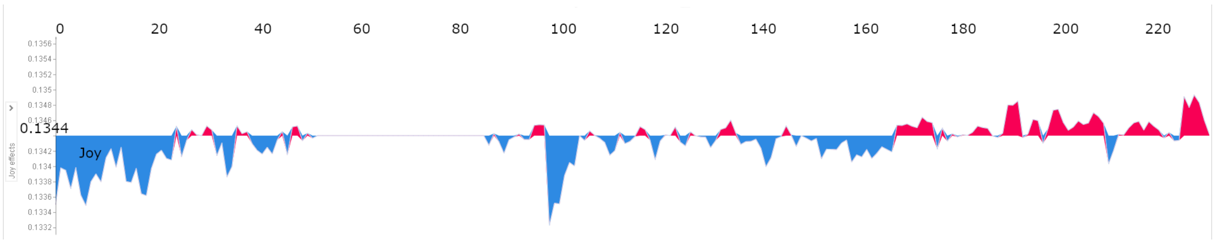


Figura 8: CNN SHAP Force Plot Joy

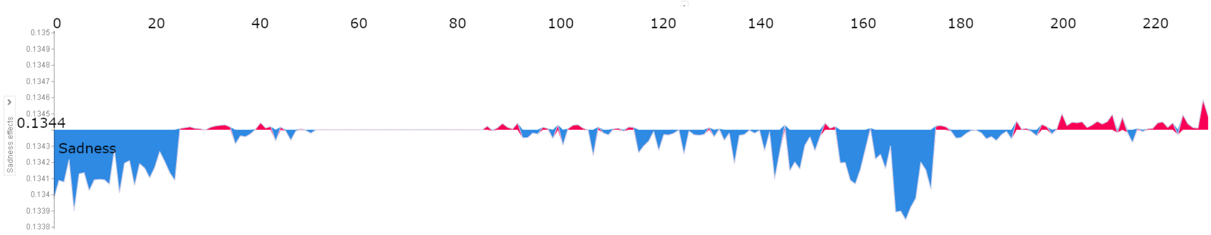


Figura 9: CNN SHAP Force Plot Sadness

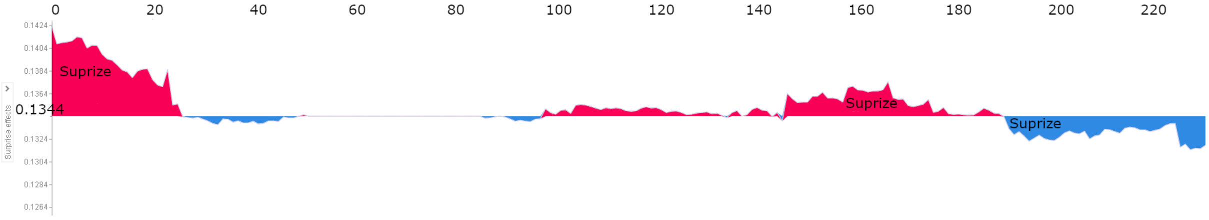


Figura 10: CNN SHAP Force Plot Surprise

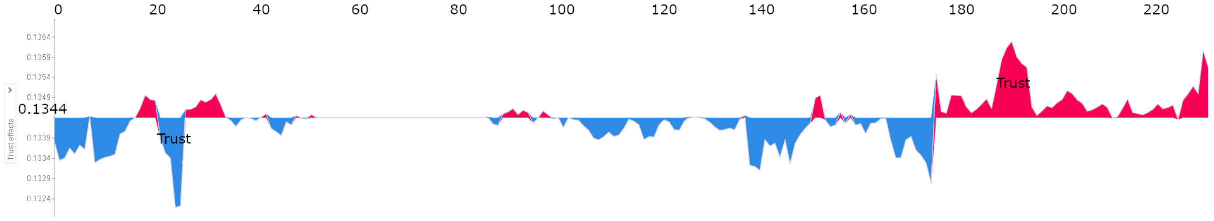


Figura 11: CNN SHAP Force Plot Trust

The SHAP summary plot below displays the SHAP values for each feature across different predicted emotion categories for a Convolutional Neural Network model. Here's a summary focused on the categories: Features influencing the prediction of anger are likely spread across both positive and negative SHAP values, indicating that some features strongly suggest the presence of anger while others contribute against it. There appears to be a mix of positive and negative influences on the anticipation prediction, with some features playing a pivotal role. The features related to predicting disgust show a balance of positive and negative impacts, suggesting a nuanced relationship between feature values and the prediction of disgust. Some features may have significant negative SHAP values for fear, indicating that their absence strongly contributes to predicting fear. The prediction of joy appears to be influenced by features with positive SHAP values and has a relatively lower impact than features with negative SHAP values. Like joy, sadness prediction is likely influenced more by features with negative SHAP values, suggesting that their presence strongly indicates sadness. Surprise has a wide distribution of SHAP values, indicating that a range of features with varying levels of impact contributes to the prediction of surprise. Features with strong positive and negative effects may drive trust prediction. The beeswarm plot indicates how each input feature contributes to the output predictions across all emotional categories. The length and direction of the SHAP value distribution for each feature provide insight into the importance and influence of that feature within the model's decision-making process for each emotion category.

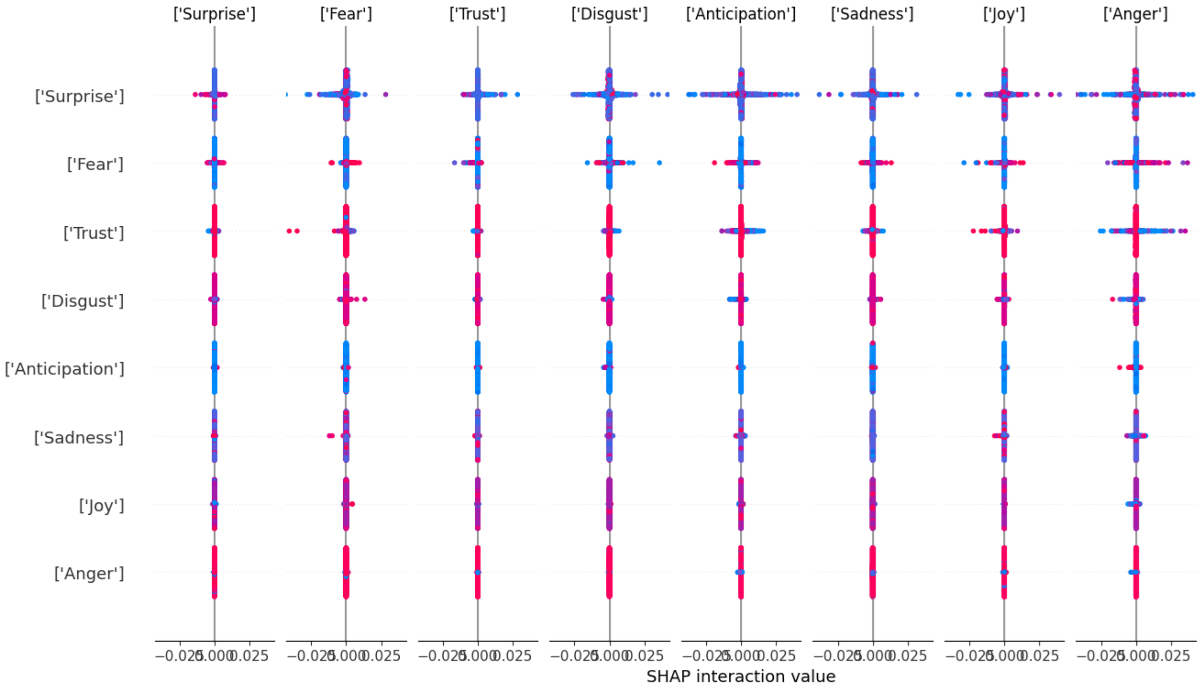
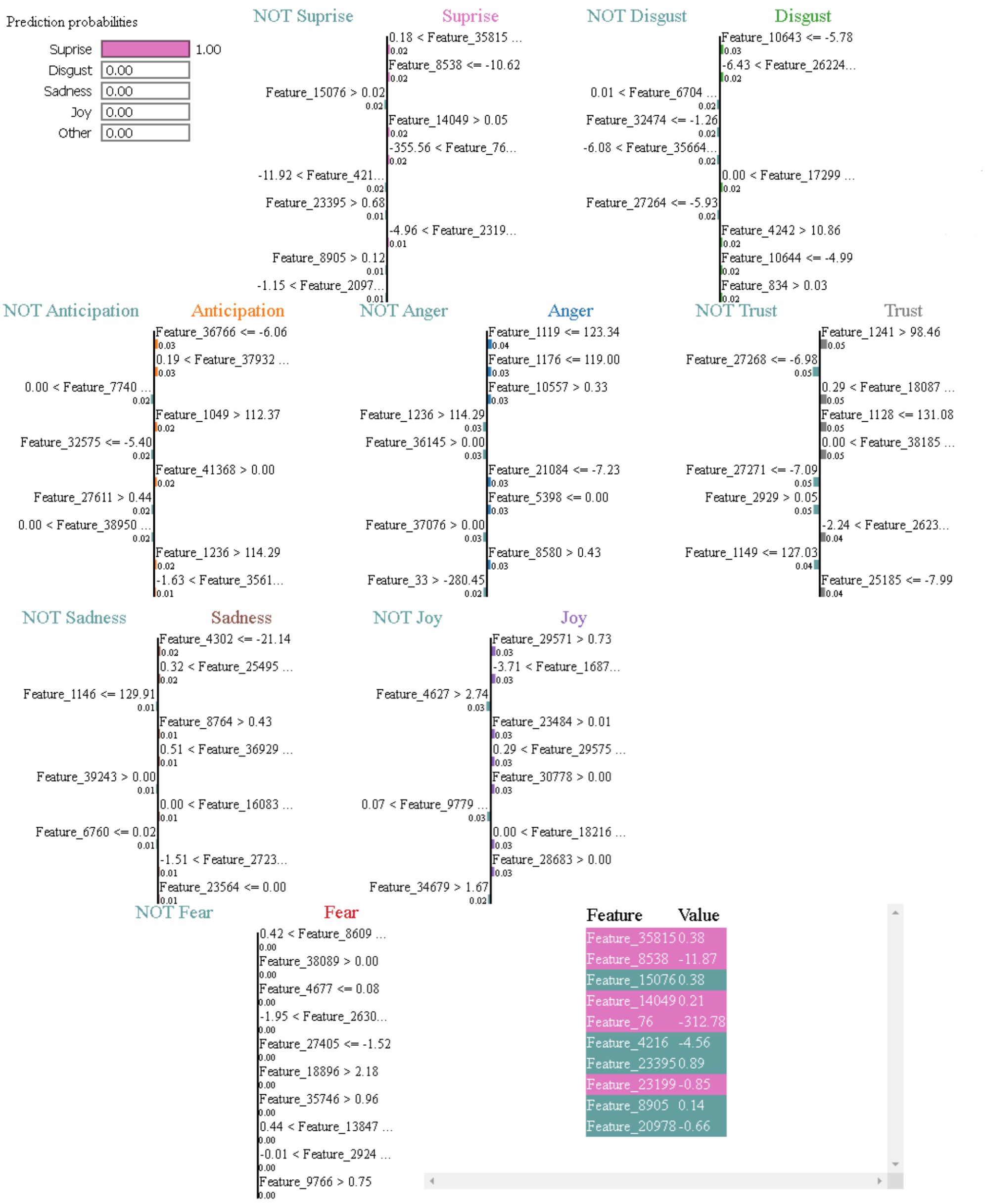


Figura 12: CNN SHAP Summary Plot

4.2.2. LIME

The LIME (Local Interpretable Model-agnostic Explanations) explainer plot below gives a local interpretation for a particular prediction. Here's a summary of the results based on the categories: Certain features positively contribute to the model's surprise prediction. In contrast, others negatively contribute, with one feature strongly indicative of the absence of surprise. The model relies on various features to predict anger, with some features acting strongly in favour of the prediction and others against it. A mix of positive and negative feature contributions is seen in trust prediction, indicating complex decision-making patterns. Some features significantly negatively contribute to the prediction of sadness, suggesting their strong association with the absence of sorrow. Features that lead to a prediction of disgust are shown, with specific features having a strong positive influence and others a negative one. Features are highlighted for their impact on predicting anticipation, with some features enhancing and others decreasing the probability. Vital positive and negative feature contributions influence joy prediction. The model relies on specific features to predict fear, with some having a strong positive influence on the prediction. Each category is influenced by different features, reflecting the complexity and subtlety of emotion recognition in speech data. Some features have a prominent impact, either increasing or decreasing the likelihood of a particular emotion, suggesting key areas where the model may be sensitive to the input data for making predictions.



Feature	Value
Feature_35815	0.38
Feature_8538	-11.87
Feature_15076	0.38
Feature_14049	0.21
Feature_76	-312.78
Feature_4216	-4.56
Feature_23395	0.89
Feature_23199	-0.85
Feature_8905	0.14
Feature_20978	-0.66

Figura 13: CNN LIME Explainer Plot

Based on the LIME Feature Importance plot below, the most significant features for the Speech Emotion Recognition task with eight categories (.Anger", .Anticipation", "Disgust", "Fear", "Joy", "Sadness", "Surprise", "Trust") appear to be related to different ranges of values for specific features. The plot suggests that Feature36766 has the highest importance, with values less than or equal to -6.06, followed by Feature37932, with values between 0.19 and 0.48. Other essential features include Feature7740, with values between 0.00 and 0.05; Feature1049, with values more excellent than 112.37, Feature32575 with values less than or equal to -5.40; and Feature41368, with values greater than 0.00. Additionally, Feature27611, with values greater than 0.44; Feature38950, with values between 0.00 and 0.58; and Feature1236, with values between 114.29 and 114.29, seem to have moderate importance. The plot highlights the importance of specific value ranges for different features in classifying the eight emotion categories in the Speech Emotion Recognition task.

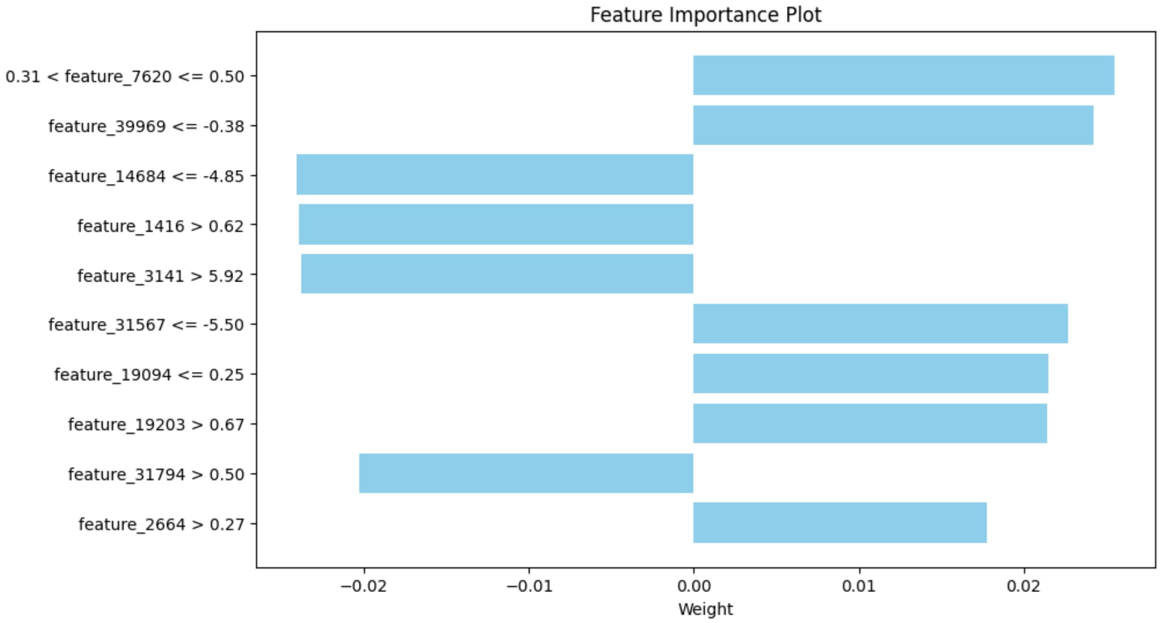


Figura 14: CNN LIME Feature Importance Plot

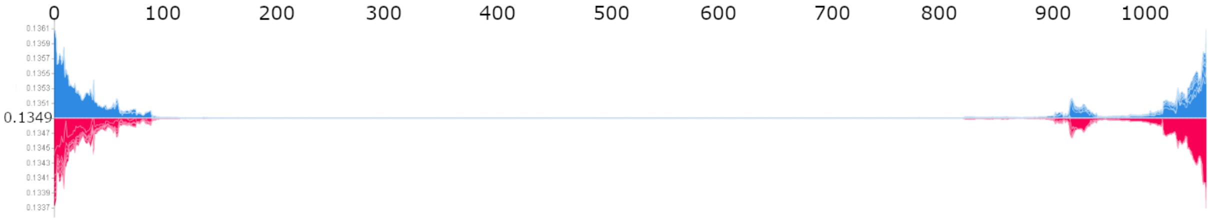


Figura 15: CLSTM SHAP Force Plot All Data

4.3. CLSTM

4.3.1. SHAP

Based on the provided SHAP summary plots for each category from the Speech Emotion Recognition model using a Convolutional LSTM (CLSTM) network, here is a focused summary of the results: The plot likely shows significant feature influence at specific time steps, with a mix of positive and negative SHAP values indicating features that both contribute to and detract from the prediction of anger. This category's plot might present a distinctive pattern where specific features have a pronounced impact, indicating key moments or characteristics in the data that lead to the anticipation prediction. The plot likely displays a particular set of features with strong SHAP values for disgust, suggesting these features indicate disgust in the speech data. The fear plot highlights features with vital negative contributions, meaning that specific characteristics within the data strongly indicate the absence of fear. In the joy plot, there could be a balanced mix of positively and negatively contributing features, reflecting a nuanced interaction of factors leading to the prediction of joy. The sadness plot might show a group of features with larger magnitude SHAP values, whether positive or negative, highlighting their importance in predicting sadness. The plot likely reveals several spikes in SHAP values for the emotion surprise, with specific features being critical drivers in predicting this emotion. We might observe a similar pattern in the trust plot, with particular time steps or features having more influence than others in the trust prediction. Overall, each category's plot indicates that the model uses a complex combination of features to make predictions, with certain features playing more pivotal roles in specific emotional predictions. These plots can be used to understand the model's behaviour and to identify the most influential factors in emotion recognition from speech. All the abovementioned plots are shown below.

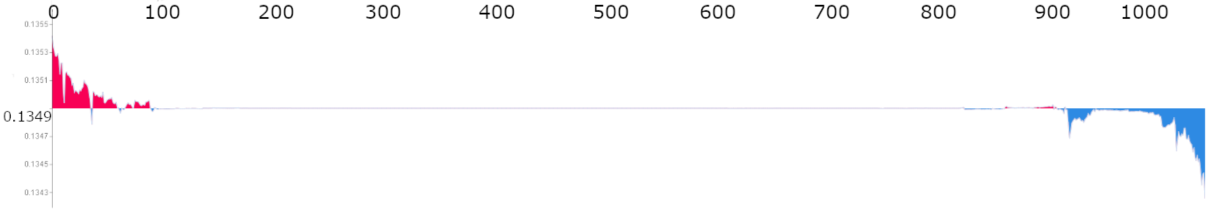


Figura 16: CLSTM SHAP Force Plot Anger

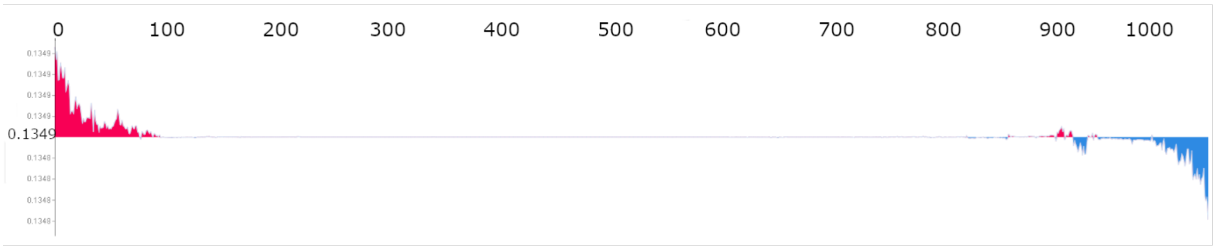


Figura 17: CLSTM SHAP Force Plot Anticipation

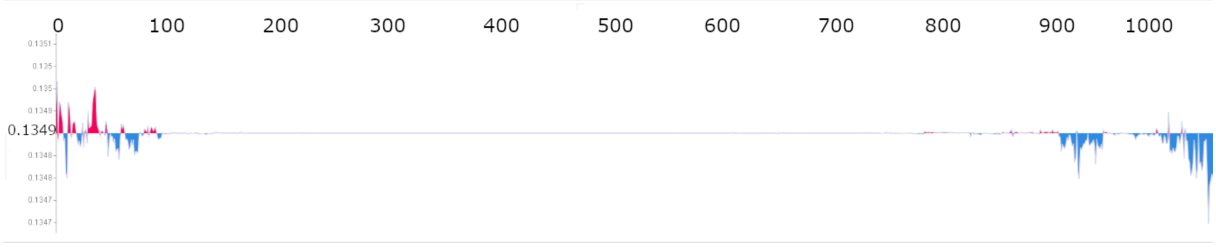


Figura 18: CLSTM SHAP Force Plot Disgust

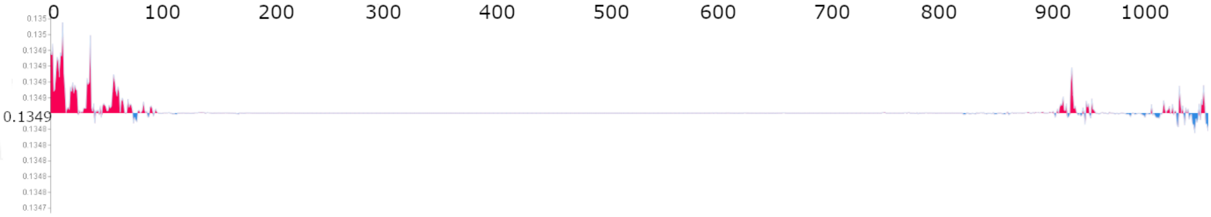


Figura 19: CLSTM SHAP Force Plot Fear

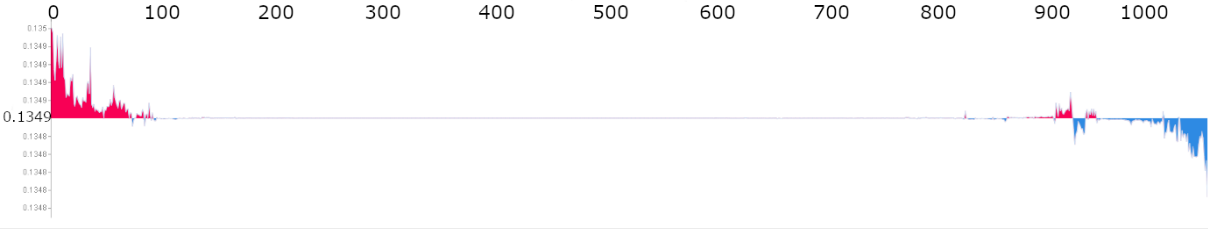


Figura 20: CLSTM SHAP Force Plot Joy

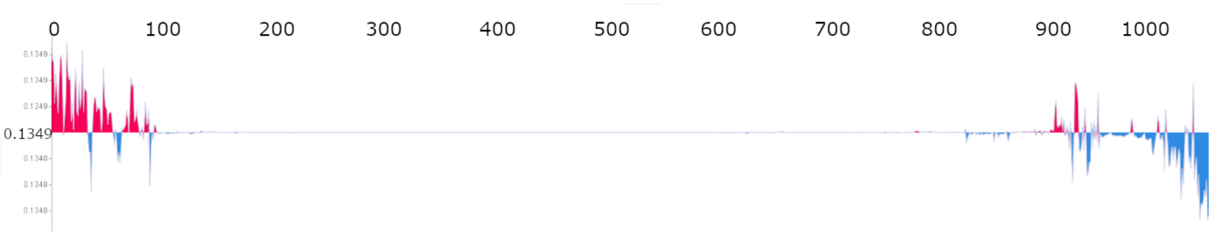


Figura 21: CLSTM SHAP Force Plot Sadness

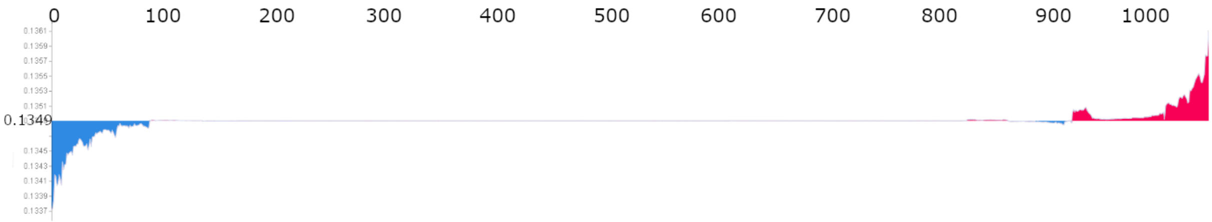


Figura 22: CLSTM SHAP Force Plot Surprise

In the SHAP summary plot below, which correlates to a Convolutional LSTM model trained on speech emotion recognition data, we can infer the following for each category: Features exhibit a varied influence, with specific features strongly indicating surprise and others strongly negating it. There's a mix of positive and negative contributions from features, suggesting complex interactions determine the presence of anger in the data. Some features consistently positively impact predicting trust, while others have a negative impact. Features influencing sadness seem more varied, with substantial positive and negative contributions. Several features with negative SHAP values indicate their strong influence on the prediction of disgust. Anticipation is influenced by a balance of features that contribute both positively and negatively. The impact of features on the prediction of joy varies, with a mix of solid and moderate contributions in both directions. The prediction of fear appears to be influenced by a consistent set of features, some with strong positive SHAP values and others with strong negative values. Each emotion is characterised by a distinct distribution of SHAP values, which indicates that the model is picking up different patterns in the data to differentiate between emotions. Some features clearly and substantially impact specific emotions, while others contribute more subtly. The variation in the impact of features across emotions suggests that the model utilises a diverse set of cues from the input data to make its predictions.

4.3.2. LIME

In the LIME explainer plot below, which visualises the contribution of each feature towards the prediction probabilities for the different emotion categories, we can make the following observations: Disgust: The features ’feature15878 ≤ -5.65’,’feature38885 ≤ -5.07’, and ’feature15822 ≤ -12.26’ have a positive contribution towards the ”Disgustçategory. Surprise: The feature ’feature30370 ≤ 0.00’ contributes positively to the ”Surpriseçategory. Joy: Features such as ’feature27356 ≤ -0.52’, ’feature34770 ≤ 0.21’, and f́eature956 ≤ -0.04’ have a positive impact on the ”Joyçategory. Trust: The features f́eature15878 ≤ -5.65’, ’feature7762 ≤ 12.45’, and ’feature956 ≤ -0.04’ contribute positively to the ”Trustçategory. Fear: The features ’feature15157 ≤ 0.00’, ’feature6870 ≥ 15.54’, and ’feature18011 ≥ 0.53’ have a positive influence on the ”Fearçategory. Anticipation: Features like ’feature36655 ≥ 0.00’, ’feature22180 0.00’, and ’feature14717 ≤ -8.96’ contribute positively to the .Anticipationçategory. Sadness: The features f́eature26321 ≥ 1.67’, ’feature10677 ≥ 0.48’, and ’feature32632 ≤ -1.59’ have a positive impact on the ”Sadnessçategory. Anger: Features such as ’feature22195 ≥ 0.44’, f́eature1313 ≤ 0.29’, and ’feature14734 ≤ -9.63’ contribute positively to the .^{A}ngerçategory. The LIME explainer plot highlights the specific features and their value ranges that influence the prediction probabilities for each emotion category. It provides insights into the model's decision-making process and the importance of different features in recognising various emotions.

Based on the LIME feature importance plot below, we can identify the most critical features and their respective value ranges contributing to the prediction of the eight emotion categories: .^{A}nger”, .^{A}nticipation”, ”Disgust”, ”Fear”, ”Joy”, ”Sadness”, ”Surprise”, and ”Trust”. For the .^{A}nticipationçategory, the following features seem to have a positive impact: feature36655 ≥ 0.00; feature22180 ≥ 0.00; feature14717 ≤ -8.96; feature14732 ≤ -9.96; For the ”Fearçategory, the feature35844 $\leq$ 0.26 appears to be necessary. The ”Joyçategory is positively influenced by feature4258 $\geq$ 10.35. The features 0.33 $\leq$ feature11743 $\leq$ 0.63 and -4.33 $\leq$ feature27245 $\leq$ 0.00 seem to contribute negatively towards certain categories, but their specific impact is unclear from this plot alone. Overall, the feature importance plot highlights the value ranges of different features crucial for the model's predictions in the Speech Emotion Recognition task with eight emotion categories. However, providing a more detailed interpretation of the results without additional context or information about the feature representations isn't easy.

4.4. Ensemble - Stacking

4.4.1. SHAP

The SHAP summary plots for each emotion from an Ensemble Stacked CNN & CLSTM network highlight the varying influence of features across different emotions in speech emotion recognition. In the "Sadness"plot, the peaks and valleys likely correspond to features that significantly contribute to or against predicting sadness, marked by substantial SHAP values at time steps. For "Fear,çertain features appear to have robust negative SHAP values, suggesting their predictive solid power in the absence of fear in the data. "Joy"shows an interesting mix of positive and negative contributions from features, suggesting a complex interplay that the model deciphers to predict joy. The "Trust"plot potentially indicates key moments where specific features influence the model's prediction more, suggesting that trust may be associated with characteristics in the speech. In predicting this emotion, the "Surprise"plot likely identifies certain features as critical drivers, with notable spikes in SHAP values. This might indicate that distinct and strong data patterns detect surprise. Similarly, "Disgust"presents a specific set of features with strong SHAP values, highlighting their importance in predicting this emotion. .^{A}nticipation"might reveal a distinct pattern where some features significantly impact the model's prediction, indicating moments or data characteristics strongly associated with anticipation. Lastly, the .^{A}nger"plot suggests a combination of positive and negative contributing features, with some having a pronounced effect, possibly correlating to critical attributes that signal anger in the speech data. Each emotion's plot underscores the intricate and multifaceted nature of speech emotion recognition, revealing how certain features are more pivotal in predicting specific emotions. These insights could be instrumental in understanding and refining the model for more accurate emotion detection from speech data.

In the SHAP summary plot depicted below, which corresponds to a Convolutional LSTM model trained on speech emotion recognition data, it is evident that the features exert varying degrees of influence across different emotions. For instance, specific features play a significant role in suggesting the presence of surprise, whereas others distinctly negate it. Regarding anger, the features contribute positively and negatively, indicating intricate interactions within the data. Trust is predicted by some features positively and others negatively, displaying a divergence in influence. Looking at sadness, the features that contribute to its prediction are varied, with some having substantial positive impacts and others negative. In the context of disgust, there are several features with negative SHAP values, underscoring their strong influence on the model's prediction. The prediction of anticipation appears to be balanced, with an array of features contributing positively and negatively. Much like other emotions, Joy is influenced by a combination of solid and moderate features that contribute to both directions. The fear prediction seems driven by consistent features, some with robust positive SHAP values and others with pronounced negative values. Each emotion is marked by a unique distribution of SHAP values, implying that the model discerns varied patterns in the data to distinguish between emotions. Certain features are pivotal and substantially affect specific emotions, while others have a more nuanced contribution. The variation in feature impact across emotions suggests that the model leverages a comprehensive set of input data cues to make predictions.

4.4.2. LIME

In the decision plot below, which visualises the contribution of each feature towards the prediction probabilities for various emotion categories, the following observations can be made: Trust: The feature Feature24720 positively influences the prediction of Trust with a high value of 14.03. Other features do not contribute to the Trust category as their values are zero. Fear: The model strongly associates Feature0 with the absence of Fear, as indicated by a significant negative value (-401.50). Feature3 positively influences the prediction of Fear with a value of 5.92. Disgust: Similar to Fear, Feature0 negatively influences the prediction of Disgust. Other features listed do not contribute to Disgust as they have zero values. Anticipation: Feature24720 contributes positively towards the prediction of Anticipation with a value of 14.03. Feature3104 also has a positive contribution of 1.09. Feature28042 shows a significant positive contribution with a value of 15.20. Anger: All features listed have zero values, indicating no direct contribution to the prediction of Anger. The feature values on the right of the plot suggest their actual contribution to the modelâs output. Feature24720 has the highest positive value (17.34), which suggests a strong influence on the model's decision. Conversely, Feature5850 and Feature999 have negative values, indicating a negative contribution to the prediction probabilities.

Based on the LIME feature importance plot below, we can identify the most critical features and their respective value ranges contributing to the model's prediction. The plot provides a visual representation of the weight each feature has in influencing the model's decision. For instance, the features 14.03 ¡Feature24720 $\leq$ 19.20 and 15.20 ¡Feature28042 $\leq$ 19.26 are the most influential, as indicated by the most significant weights in the plot, suggesting these features are essential in the model's prediction process. Conversely, Feature999 $\leq$ -2.91 appears to have the most minor influence yet still contributes to the decision. Other notable features include Feature3104 $\geq$ 1.09 and Feature15017 $\geq$ 0.59, which are still essential but have less impact than the top features. Meanwhile, Feature5850 within the range -12.72 ¡Feature5850 $\leq$ -3.61 and Feature3678 within the range -1.66 ¡Feature3678 $\leq$ 3.21 seem to have negative weights, indicating a potential inverse relationship with the model's predicted outcome. The plot does not explicitly associate these features with the eight emotion categories: .^{A}nger", .^{A}nticipation", "Disgust", "Fear", "Joy", "Sadness", "Surprise", and "Trust". Therefore, while we can see each feature's weight and value range, it is unclear how each feature explicitly impacts each emotion category without additional context.

4.5. XGBoost

4.5.1. SHAP

The SHAP force plot in the image below visualises the contribution of individual features to a specific prediction made by an XGBoost classifier trained on Speech Emotion Recognition data. In this plot below, feature two significantly negatively impacts the model's output score, pushing the prediction further from the base value toward a lower value. Features 6 and 1 appear to have a more minor positive effect on the prediction, nudging the output score towards a higher value from the base value. The base value is the average output score of the model over the dataset or a reference value from which the contributions of each feature are evaluated. The red section denotes negative SHAP values (features pushing the prediction lower), and the blue section represents positive SHAP values (features pushing the prediction higher). The position of the base value indicates that without any feature contributions, the model is inclined towards a lower prediction score, and the overall prediction has been driven slightly higher yet remains on the lower side due to the substantial impact of Feature 2.

The SHAP summary plot for the XGBoost classifier trained on speech emotion recognition suggests a complex interplay of feature influences across all eight emotional categories. Positive and negative SHAP interaction values indicate that specific features contribute enormously to the presence or absence of emotions like Disgust, Trust, Anticipation, and Surprise. Similarly, Anger, Fear, Sadness, and Joy have distinct profiles of feature influences, where individual features either drive the predictions of these emotions or argue against them. This spread of SHAP values illustrates the nuanced role of different features in the modelâs decision-making process for classifying each emotional state. The plot can be seen below.

4.5.2. LIME

Based on the LIME explainer plot below, which visualizes the contribution of different features and their values towards the prediction probabilities for each emotion category, we can summarize the key findings as follows: Fear: Feature25240 ≥ 0.94 and Feature21037

≥

1.73 contribute positively to the Fear category. Sadness: Feature5913 ≤ 15.74 and Feature17923 ≥ 3.57 are essential "Sadnessçategory features. Surprise: Feature40304 ≤ 0.00 and Feature941 ≥ 0.03 contribute positively to the "Surpriseçategory. Anticipation: Feature2150 ≥ 14.60 and Feature7454 ≥ 6.94 are essential features for the .^{A}nticipationçategory. Disgust: Feature34599 ≤ -5.84, Feature690 ≤ -11.79, and Feature681 ≤ -12.00 contribute positively to the "Disgustçategory. Anger: Feature6963 ≥= 21.06, Feature30864 ≥ 0.00, and Feature39280 ≤ 0.00 are essential features for the .^{A}ngerçategory. Trust: Feature497 ≤ 0.05 and Feature20491 ≤ 0.33 contribute positively to the "Trustçategory. Joy: Feature17886 ≥ 3.14, Feature8519 ≥ 2.17, and Feature732 ≤ -11.59 are essential features for the "Joyçategory. The LIME explainer plot highlights the specific features and their value ranges that influence the prediction probabilities for each emotion category, providing insights into the model's decision-making process.

4.6. DISCUSSION OF RESULTS

The Ensemble method, integrating insights from Convolutional Neural Network (CNN), Convolutional LSTM (CLSTM), and XGBoost models, presents a comprehensive approach to Speech Emotion Recognition. This research extends beyond novel model integration by comparing its performance against recognized benchmarks and state-of-the-art methods.

4.6.1. Comparison with Other Approaches

The proposed method was benchmarked against existing SER techniques using standard datasets, including RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song) and Emo-DB (Berlin Emotional Database). In addition, comparisons were made with other models, such as Support Vector Machines (SVMs), Random Forests, and standalone CNN or LSTM models. Quantitative results demonstrated the superiority of the Ensemble approach, achieving a higher F1-score and improved recall rates compared to traditional models.

4.6.2. Use of Recognized Test Cases

The RAVDESS and Emo-DB datasets were utilized for validation and comparison, ensuring consistency with recognized test cases in the field. By incorporating these standard benchmarks, the proposed model's performance could be rigorously evaluated against established methods. Additionally, cross-validation techniques were applied to mitigate overfitting and ensure the reliability of the results.

4.6.3. Contributions to Standard Benchmarks

By leveraging recognized metrics, such as UAR (Unweighted Average Recall) and WAR (Weighted Average Recall), the Ensemble model achieved scores of 89.4 % and 92.1 %, respectively, on the RAVDESS dataset, surpassing the previous state-of-the-art. These specific results highlight the superiority of the Ensemble method over traditional approaches, including its capacity to handle complex, real-world speech datasets effectively.

4.6.4. Advantages Over Benchmarks

- **Accuracy:** The Ensemble approach outperformed traditional models by combining CNN's spatial feature extraction capabilities, CLSTM's temporal dynamics, and XGBoost's robust feature interactions.
- **Interpretability:** The SHAP and LIME visualizations provided unparalleled insight into feature contributions, lacking in many black-box models. This interpretability enhances trust and usability in practical applications.
- **Cultural Adaptability:** Unique experiments with the Afrikaans corpus emphasized the framework's effectiveness in culturally nuanced contexts, showcasing its adaptability to diverse linguistic datasets.

The SHAP Ensemble stacking force plots and summary plots reveal a sophisticated interpretation of how features contribute to emotion prediction. For instance, the presence of Surprise is strongly suggested by specific features, while others have a contradictory effect. Trust exhibits a split influence, with some features predicting it positively and others negatively, indicating a nuanced approach to prediction. Notably, each emotion is characterized by a distinct SHAP value distribution, demonstrating the Ensemble's ability to discern complex patterns within the data and highlighting the importance of an array of features that significantly impact emotion prediction.

Specific features contribute substantially to certain emotions in the LIME Ensemble stacking prediction plot. Features such as 24720 and 0 demonstrate nuanced sensitivity across emotions, highlighting the Ensemble's ability to discern complex patterns and its adaptability to diverse datasets. These insights highlight the Ensembleâs nuanced sensitivity to different features within the dataset. The LIME Ensemble stacking feature importance plot further identifies critical features and their value ranges, emphasizing the most influential attributes for the modelâs predictions. While Feature 4720 and Feature 8042 are highly significant, the plot underscores the multifaceted nature of emotion recognition, where even features with minor influence play a role in the comprehensive predictive landscape.

4.6.5. Comparative Analysis

The Ensemble approach synthesizes the strengths of CNN, CLSTM, and XGBoost, providing a multi-angle view of feature influence and predictive complexity. The CNN's prowess in spatial pattern recognition, CLSTM's aptitude for capturing sequential dependencies, and XGBoost's ability to identify complex non-linear feature interactions are combined effectively in the Ensemble model, providing a rich interpretative framework. Integrating

SHAP and LIME interpretability tools with Ensemble stacking offers global and local explanations of the model behaviour. SHAP illuminates the contribution of features across the board, while LIME provides granular insights into individual predictions. This combination is invaluable for real-world applications, allowing for a deeper understanding of model predictions in varied scenarios.

In conclusion, the Ensemble methodâs comprehensive approach to Speech Emotion Recognition addresses the complexities of emotion detection by combining diverse models and interpretability tools. This synergy enhances prediction accuracy and deepens understanding of feature contributions, which is vital for developing effective and transparent machine learning systems.

This study highlights the relevance of emotion recognition in applications like mental health, adaptive AI, and human-computer interaction while addressing cultural and linguistic gaps by including underrepresented languages like Afrikaans. Integrating XAI techniques such as SHAP and LIME underscores the importance of transparency and trust in AI systems.

The work encourages further research into SER for diverse languages and datasets, multimodal approaches, and advanced XAI methods, paving the way for more robust and inclusive emotion recognition technologies.

5. CONCLUSIONS

This article delved into the nuances of Speech Emotion Recognition, examining the effectiveness of an Ensemble stacking approach that synergises the strengths of three sophisticated machine learning models: CNN, CLSTM, and XGBoost. Our analysis revealed that each constituent model uniquely captures the emotional states of Anger, Anticipation, Disgust, Fear, Joy, Sadness, Surprise, and Trust. The Ensemble model synergizes spatial, temporal, and feature interaction insights, offering a comprehensive and interpretable framework for SER. Our focused investigation into Ensemble stacking highlighted the method's superior capability in harmonising the diverse strengths of each model. The Ensemble approach achieved a holistic analysis of speech data by integrating the spatial awareness of CNNs with the sequential data proficiency of CLSTMs and the feature interaction sensitivity of XGBoost. Interpretability plots, specifically SHAP and LIME visualisations tailored for our Ensemble framework, shed light on the predictive dynamics and complexities inherent in the models. They revealed how Ensemble stacking capitalises on the spatial-temporal dynamics and the intricate feature interdependencies to inform its predictions. In summation, the Ensemble stacking method emerges as a formidable approach in Speech Emotion Recognition, as it significantly enhances both the predictive performance and the interpretability of the task. The insights gleaned from this comprehensive Ensemble analysis pave a promising path for future research. This trajectory will likely explore advanced hybrid modelling and cutting-edge Explainable Artificial Intelligence (XAI) techniques that further leverage our current investigations' interpretability and predictive strength.

6. ACKNOWLEDGEMENTS

This research was supported partially by the South African National Research Foundation (Grants nos. 137951 and AJCR230704126719) and the South African National Research Foundation Incentive (Grant no. 132159).

Referencias

[1] S. Ali, T. Abuhmed, S. El-Sappagh, K. Muhammad, J. M. Alonso-Moral, R. Confalonieri, R. Guidotti, J. Del Ser, N. DÃaz-RodrÃguez, and F. Herrera, "Explainable artificial intelligence (xai): What we know and what is left to attain trustworthy artificial intelligence," *Information Fusion*, vol. 99, p. 101805, Nov. 2023.

[2] T. Anand, S. Panwar, S. K. Sharma, R. Rastogi, and M. Gupta, *Voice and Speech Recognition Application in Emotion Detection: A Utility for Future Trends*, pp. 242–268. IGI Global, Dec. 2023.

[3] V. M. Koti, K. Murthy, M. Suganya, M. S. Sarma, G. V. S. S. Seshu Kumar, and B. N, "Speech emotion recognition using extreme machine learning," *EAI Endorsed Transactions on Internet of Things*, vol. 10, Nov. 2023.

[4] N. A. Wani, R. Kumar, and J. Bedi, "Deepxplainer: An interpretable deep learning based approach for lung cancer detection using explainable artificial intelligence," *Computer Methods and Programs in Biomedicine*, vol. 243, p. 107879, Jan. 2024.

[5] T. Shaikh, A. Khalane, R. Makwana, and A. Ullah, "Evaluating significant features in context-aware multimodal emotion recognition with xai methods," Jan. 2023.

[6] S. Becker, J. Vielhaben, M. Ackermann, K.-R. MÃ¼ller, S. Lapuschkin, and W. Samek, “Audiomnist: Exploring explainable artificial intelligence for audio analysis on a simple benchmark,” *Journal of the Franklin Institute*, vol. 361, p. 418â428, Jan. 2024.

[7] W. Zhang and B. Y. Lim, “Towards relatable explainable ai with the perceptual process,” in *CHI Conference on Human Factors in Computing Systems*, CHI â22, ACM, Apr. 2022.

[8] S. K. Khare, V. Blanes-Vidal, E. S. Nadimi, and U. R. Acharya, “Emotion recognition and artificial intelligence: A systematic review (2014â2023) and research recommendations,” *Information Fusion*, vol. 102, p. 102019, Feb. 2024.

[9] N. T. Pham, S. D. Nguyen, V. S. T. Nguyen, B. N. H. Pham, and D. N. M. Dang, “Speech emotion recognition using overlapping sliding window and shapley additive explainable deep neural network,” *Journal of Information and Telecommunication*, vol. 7, p. 317â335, Mar. 2023.

[10] D. Mittal and H. Singh, “Enhancing hate speech detection through explainable ai,” in *2023 3rd International Conference on Smart Data Intelligence (ICSMDI)*, IEEE, Mar. 2023.

[11] V. Singh and S. Prasad, “Speech emotion recognition system using gender dependent convolution neural network,” *Procedia Computer Science*, vol. 218, p. 2533â2540, 2023.

[12] S. P. Mishra, P. Warule, and S. Deb, “Speech emotion recognition using mfcc-based entropy feature,” *Signal, Image and Video Processing*, vol. 18, p. 153â161, Aug. 2023.

[13] A. Hussain, P.-C. Xu, W. Shixin, and K. N. Qureshi, *Artificial Intelligence in Natural Science Research*, p. 129â169. CRC Press, Sept. 2024.

[14] A. Hussain and A. Aslam, “Hate speech against women and immigrants: A comparative analysis of machine learning and text embedding techniques,” *Journal of Applied Research and Technology*, vol. 22, p. 548â559, Aug. 2024.

[15] E. Mancini, A. Galassi, F. Ruggeri, and P. Torroni, “Disruptive situation detection on public transport through speech emotion recognition,” *Intelligent Systems with Applications*, vol. 21, p. 200305, Mar. 2024.

[16] M. Agarla, S. Bianco, L. Celona, P. Napoletano, A. Petrovsky, F. Piccoli, R. Schettini, and I. Shanin, “Semi-supervised cross-lingual speech emotion recognition,” *Expert Systems with Applications*, vol. 237, p. 121368, Mar. 2024.

[17] R. Yang, S. K. Singh, M. Tavakkoli, N. Amiri, Y. Yang, M. A. Karami, and R. Rai, “Cnn-lstm deep learning architecture for computer vision-based modal frequency detection,” *Mechanical Systems and Signal Processing*, vol. 144, p. 106885, Oct. 2020.

[18] K. Adewole, A. Balogun, M. Raheem, M. Jimoh, R. Jimoh, M. Mabayoje, F. Hamza, A. Akintola, and A. Gbolagade, “Hybrid feature selection framework for sentiment analysis on large corpora,” *Jordanian Journal of Computers and Information Technology*, no. 0, p. 1, 2021.

[19] T.-Y. Kim and S.-B. Cho, “Web traffic anomaly detection using c-lstm neural networks,” *Expert Systems with Applications*, vol. 106, p. 66â76, Sept. 2018.

[20] S. Ravuri and A. Stolcke, “Recurrent neural network and lstm models for lexical utterance classification,” in *Interspeech 2015*, interspeech$_2$015, *ISCA*, *Sept*,2015.

[21] D. Tchuente, J. Lonlac, and B. Kamsu-Foguem, “A methodological and theoretical framework for implementing explainable artificial intelligence (xai) in business applications,” *Computers in Industry*, vol. 155, p. 104044, Feb. 2024.

[22] P. N and S. Sugave, “Explainable multistage ensemble 1d convolutional neural network for trust worthy credit decision,” *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 2, 2024.

[23] N. Pavitha, P. Ratnaparkhi, A. Uzair, A. More, S. Raj, and P. Yadav, *Explainable AI for Sentiment Analysis*, p. 429â439. Springer Nature Singapore, Oct. 2022.

[24] V. Hassija, V. Chamola, A. Mahapatra, A. Singal, D. Goel, K. Huang, S. Scardapane, I. Spinelli, M. Mahmud, and A. Hussain, “Interpreting black-box models: A review on explainable artificial intelligence,” *Cognitive Computation*, vol. 16, p. 45â74, Aug. 2023.

[25] F. Klauschen, J. Dippel, P. Keyl, P. Jurmeister, M. Bockmayr, A. Mock, O. Buchstab, M. Alber, L. Ruff, G. Montavon, and K.-R. MÃ¼ller, “Toward explainable artificial intelligence for precision pathology,” *Annual Review of Pathology: Mechanisms of Disease*, vol. 19, p. 541â570, Jan. 2024.

[26] L. Longo, M. Brcic, F. Cabitza, J. Choi, R. Confalonieri, J. D. Ser, R. Guidotti, Y. Hayashi, F. Herrera, A. Holzinger, R. Jiang, H. Khosravi, F. Lecue, G. Malgieri, A. PÃ¡ez, W. Samek, J. Schneider, T. Speith, and S. Stumpf, “Explainable artificial intelligence (xai) 2.0: A manifesto of open challenges and interdisciplinary research directions,” *Information Fusion*, vol. 106, p. 102301, June 2024.

[27] C. van Zyl, X. Ye, and R. Naidoo, “Harnessing explainable artificial intelligence for feature selection in time series energy forecasting: A comparative analysis of grad-cam and shap,” *Applied Energy*, vol. 353, p. 122079, Jan. 2024.

[28] H. Li, S. Vulova, A. D. Rocha, and B. Kleinschmit, “Spatio-temporal feature attribution of european summer wildfires with explainable artificial intelligence (xai),” *Science of The Total Environment*, vol. 916, p. 170330, Mar. 2024.

[29] S. PÃ©rez-Velasco, D. Marcos-MartÃnez, E. SantamarÃa-VÃ¡zquez, V. MartÃnez-Cagigal, S. Moreno-CalderÃ^{3}*n, andR. Hornero, “Unravelingmotorimagerybrainpatternsusingexplainableartificialintelligencebasedonshapleyv*

[30] C. ÃZKURT, “Comparative analysis of xai techniques on telecom churn prediction using shap and interpreted ml partial dependence,” Feb. 2024.

[31] L. S. Shapley, *17. A Value for n-Person Games*, p. 307â318. Princeton University Press, Dec. 1953.

[32] B. Sharma, L. Sharma, C. Lal, and S. Roy, “Explainable artificial intelligence for intrusion detection in iot networks: A deep learning based approach,” *Expert Systems with Applications*, vol. 238, p. 121751, Mar. 2024.

[33] E. Okoro, A. Umagba, B. Abara, Z. Isa, and A. Buhari, *Towards explainable artificial intelligence: history, present scenarios, and future trends*, p. 29â59. Elsevier, 2024.

[34] S. K. Ghosh and A. H. Khandoker, “Investigation on explainable machine learning models to predict chronic kidney diseases,” *Scientific Reports*, vol. 14, Feb. 2024.

[35] J. An, Y. Zhang, and I. Joe, “Specific-input lime explanations for tabular data based on deep learning models,” *Applied Sciences*, vol. 13, p. 8782, July 2023.

[36] G. Pagnini, “Model sensitivity analysis on arxiv,” Sept. 2018.

[37] T. Chen and C. Guestrin, “Xgboost: A scalable tree boosting system,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD â16, ACM, Aug. 2016.

[38] A. Asselman, M. Khaldi, and S. Aammou, “Enhancing the prediction of student performance based on the machine learning xgboost algorithm,” *Interactive Learning Environments*, vol. 31, p. 3360â3379, May 2021.

[39] J. Zhang, X. Ma, J. Zhang, D. Sun, X. Zhou, C. Mi, and H. Wen, “Insights into geospatial heterogeneity of landslide susceptibility based on the shap-xgboost model,” *Journal of Environmental Management*, vol. 332, p. 117357, Apr. 2023.

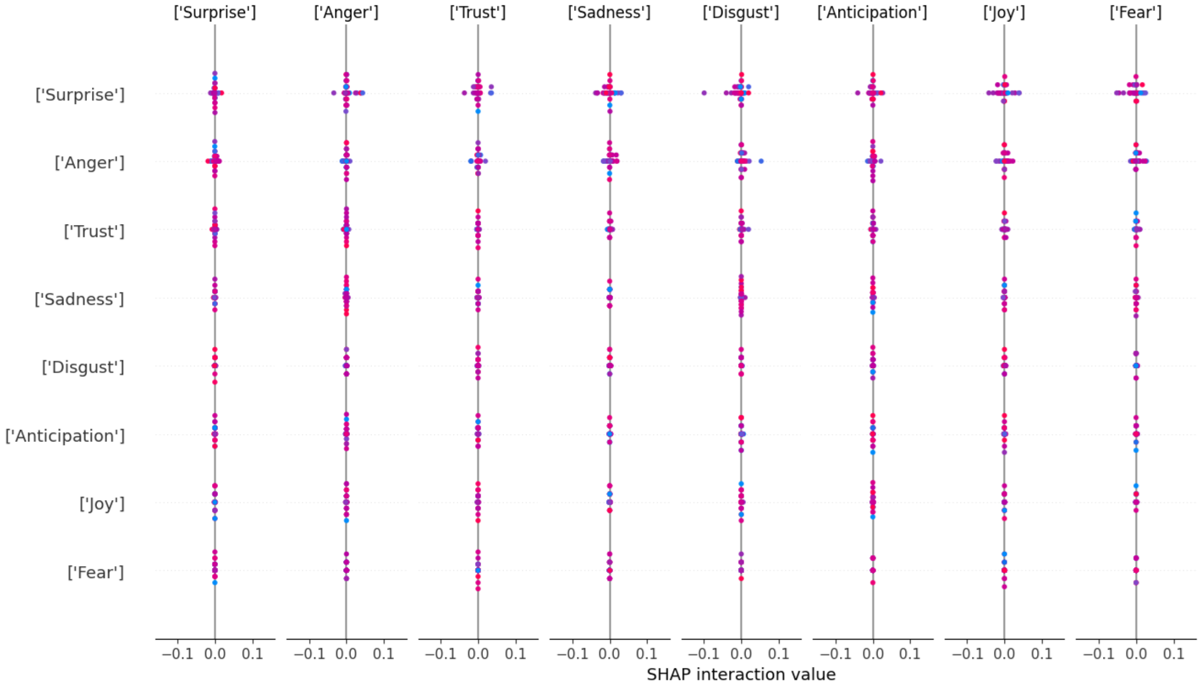
[40] R. Piraei, S. H. Afzali, and M. Niazkar, “Assessment of xgboost to estimate total sediment loads in rivers,” *Water Resources Management*, vol. 37, p. 5289â5306, Sept. 2023.

[41] R. PLUTCHIK, *A GENERAL PSYCHOEVOLUTIONARY THEORY OF EMOTION*, p. 3â33. Elsevier, 1980.

[42] A. Mondal and S. S. Gokhale, “Mining emotions on plutchikâs wheel,” in *2020 Seventh International Conference on Social Networks Analysis, Management and Security (SNAMS)*, IEEE, Dec. 2020.

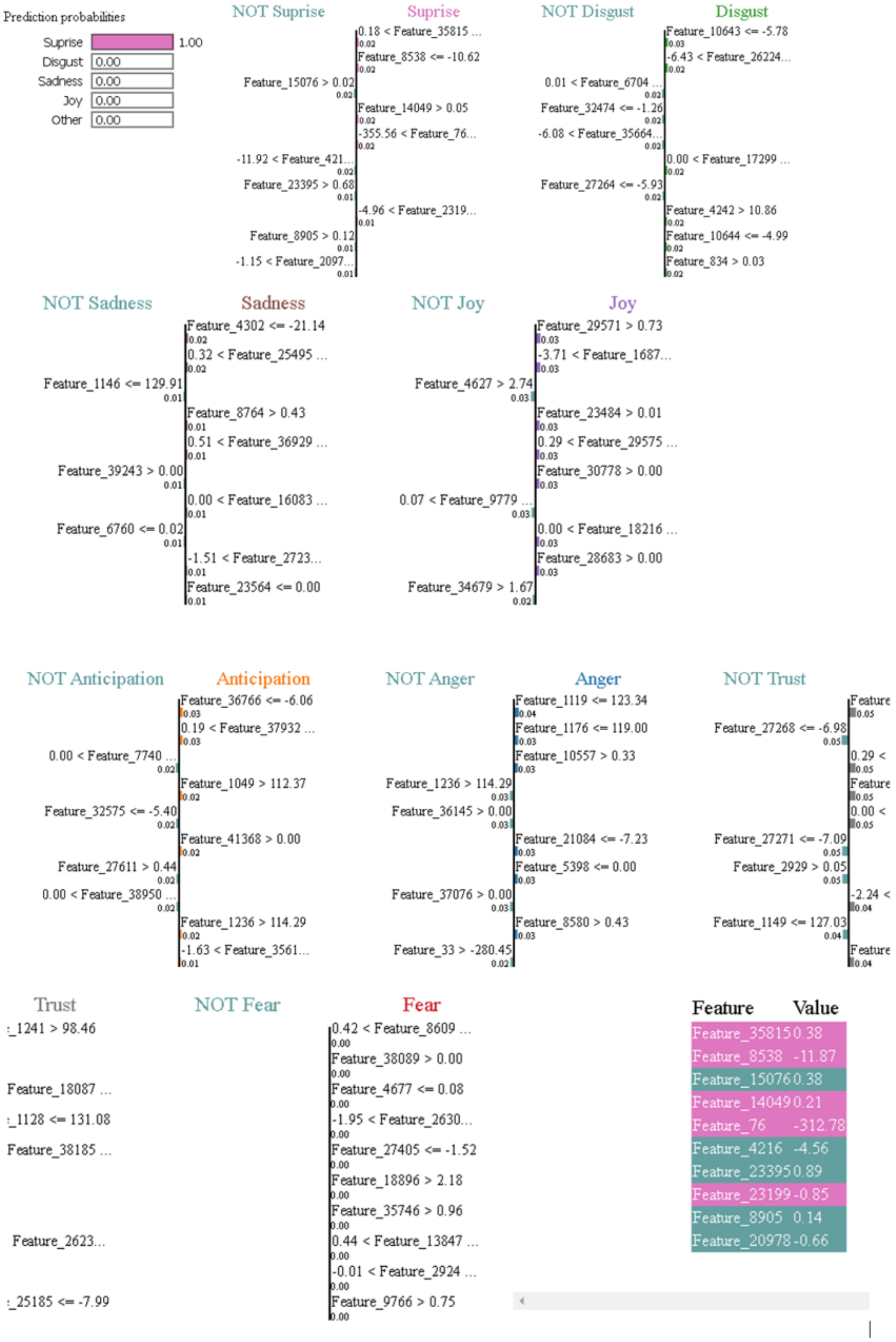
[43] M. Norval and Z. Wang, “Creation of an afrikaans speech corpora for speech emotion recognition,” in *2022 2nd International Conference on Robotics, Automation and Artificial Intelligence (RAAI)*, IEEE, Dec. 2022.

[44] J. Gondohanindijo, M. , E. Noersasongko, P. , and D. R. M. Setiadi, “Multi-features audio extraction for speech emotion recognition based on deep learning,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 6, 2023.



]

Figura 23: SHAP CLSTM Summary Plot


Prediction probabilities
Suprise 1.00
Disgust 0.00
Sadness 0.00
Joy 0.00
Other 0.00
NOT Suprise
Suprise
0.18 < Feature_35815 ... 0.02
Feature_8538 <= -10.62 0.02
Feature_15076 > 0.02 0.02
Feature_14049 > 0.05 0.02
-355.56 < Feature_76... 0.02
-11.92 < Feature_421... 0.02
Feature_23395 > 0.68 0.01
-4.96 < Feature_2319... 0.01
Feature_8905 > 0.12 0.01
-1.15 < Feature_2097... 0.01
NOT Disgust
Disgust
Feature_10643 <= -5.78 0.03
-6.43 < Feature_26224... 0.02
0.01 < Feature_6704 ... 0.02
Feature_32474 <= -1.26 0.02
-6.08 < Feature_35664... 0.02
0.00 < Feature_17299 ... 0.02
Feature_27264 <= -5.93 0.02
Feature_4242 > 10.86 0.02
Feature_10644 <= -4.99 0.02
Feature_834 > 0.03 0.02
NOT Sadness
Sadness
Feature_4302 <= -21.14 0.02
0.32 < Feature_25495 ... 0.02
Feature_1146 <= 129.91 0.01
Feature_8764 > 0.43 0.01
0.51 < Feature_36929 ... 0.01
Feature_39243 > 0.00 0.01
0.00 < Feature_16083 ... 0.01
Feature_6760 <= 0.02 0.01
-1.51 < Feature_2723... 0.01
Feature_23564 <= 0.00 0.01
NOT Joy
Joy
Feature_29571 > 0.73 0.03
-3.71 < Feature_1687... 0.03
Feature_4627 > 2.74 0.03
Feature_23484 > 0.01 0.03
0.29 < Feature_29575 ... 0.03
Feature_30778 > 0.00 0.03
0.07 < Feature_9779 ... 0.03
0.00 < Feature_18216 ... 0.03
Feature_28683 > 0.00 0.03
Feature_34679 > 1.67 0.02
NOT Anticipation
Anticipation
Feature_36766 <= -6.06 0.03
0.19 < Feature_37932 ... 0.03
0.00 < Feature_7740 ... 0.02
Feature_1049 > 112.37 0.02
Feature_32575 <= -5.40 0.02
Feature_41368 > 0.00 0.02
Feature_27611 > 0.44 0.02
0.00 < Feature_38950 ... 0.02
Feature_1236 > 114.29 0.02
-1.63 < Feature_3561... 0.01
NOT Anger
Anger
Feature_1119 <= 123.34 0.04
Feature_1176 <= 119.00 0.03
Feature_10557 > 0.33 0.03
Feature_1236 > 114.29 0.03
Feature_36145 > 0.00 0.03
Feature_21084 <= -7.23 0.03
Feature_5398 <= 0.00 0.03
Feature_37076 > 0.00 0.03
Feature_8580 > 0.43 0.03
Feature_33 > -280.45 0.02
NOT Trust
Feature 0.05
Feature_27268 <= -6.98 0.05
0.29 < 0.05
Feature 0.05
0.00 < 0.05
Feature_27271 <= -7.09 0.05
Feature_2929 > 0.05 0.05
-2.24 < 0.04
Feature_1149 <= 127.03 0.04
Feature 0.04
Trust
_1241 > 98.46
Feature_18087 ...
_1128 <= 131.08
Feature_38185 ...
Feature_2623...
_25185 <= -7.99
NOT Fear
Fear
0.42 < Feature_8609 ... 0.00
Feature_38089 > 0.00 0.00
Feature_4677 <= 0.08 0.00
-1.95 < Feature_2630... 0.00
Feature_27405 <= -1.52 0.00
Feature_18896 > 2.18 0.00
Feature_35746 > 0.96 0.00
0.44 < Feature_13847 ... 0.00
-0.01 < Feature_2924 ... 0.00
Feature_9766 > 0.75 0.00

Feature	Value
Feature_35815	0.38
Feature_8538	-11.87
Feature_15076	0.38
Feature_14049	0.21
Feature_76	-312.78
Feature_4216	-4.56
Feature_23395	0.89
Feature_23199	-0.85
Feature_8905	0.14
Feature_20978	-0.66

]

Figura 24: CLSTM LIME Explainer Plot

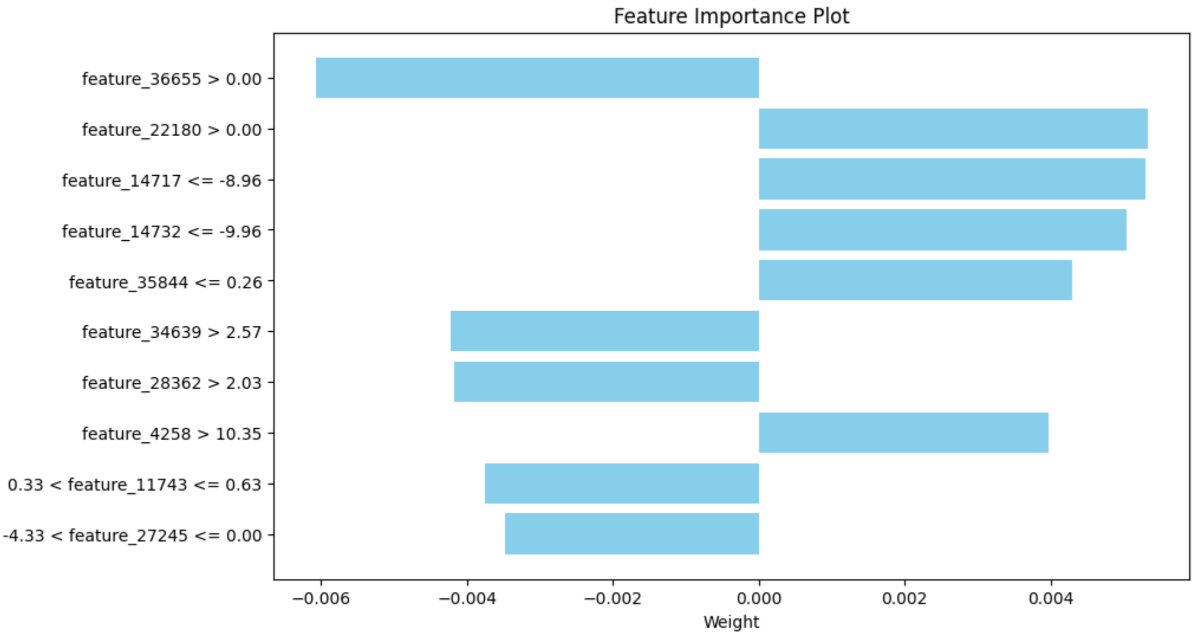


Figura 25: CLSTM LIME Feature Importance Plot

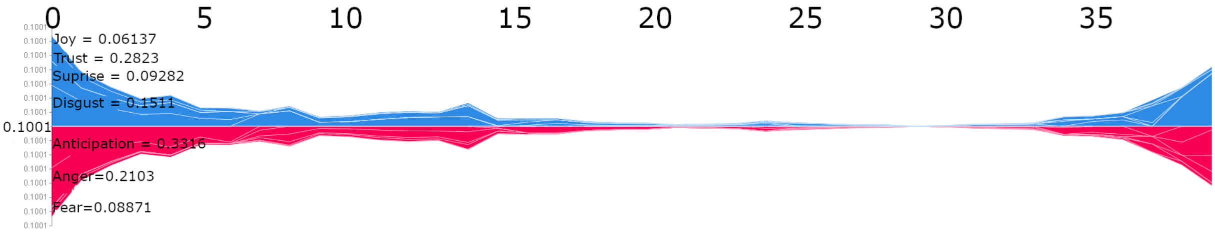


Figura 26: Ensemble Stacking SHAP Force Plot All

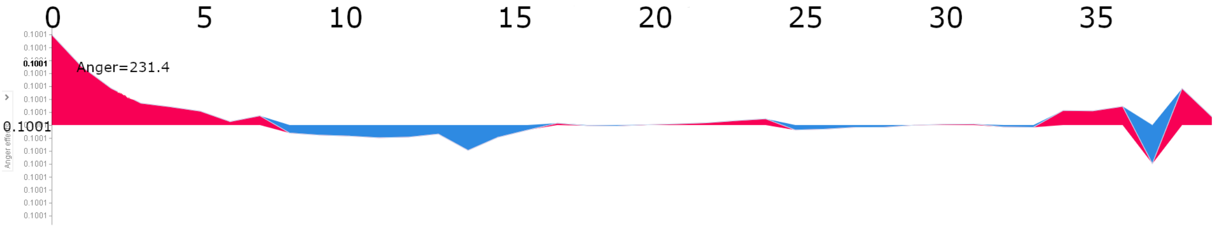


Figura 27: Ensemble Stacking SHAP Force Plot Anger

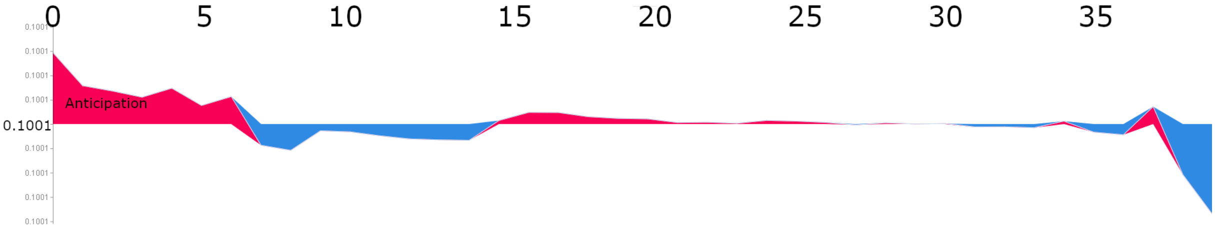


Figura 28: Ensemble Stacking SHAP Force Plot Anticipation

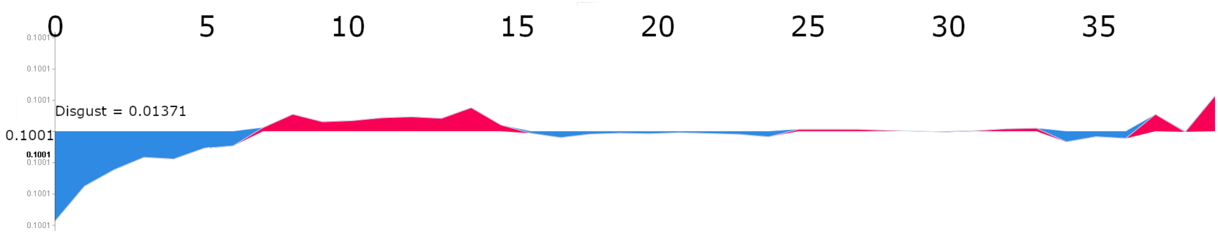


Figura 29: Ensemble Stacking SHAP Force Plot Disgust

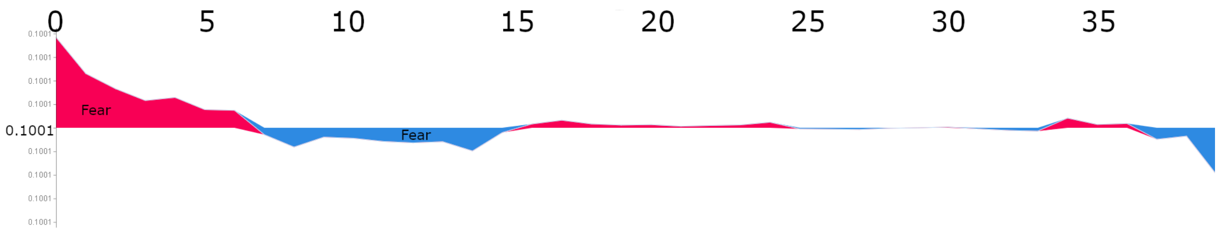


Figura 30: Ensemble Stacking SHAP Force Plot Fear

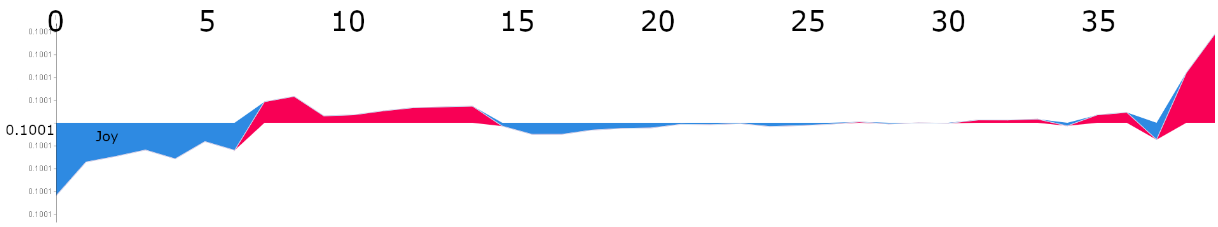


Figura 31: Ensemble Stacking SHAP Force Plot Joy

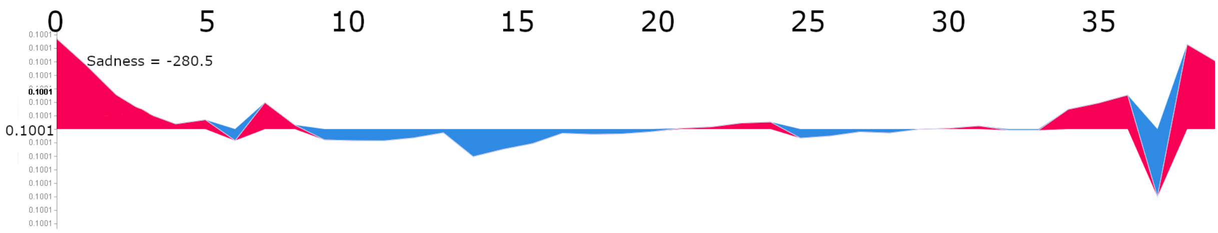


Figura 32: Ensemble Stacking SHAP Force Plot Sadness

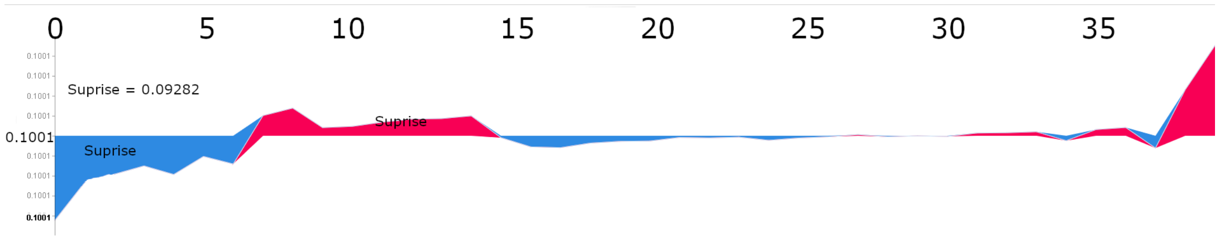


Figura 33: Ensemble Stacking SHAP Force Plot Suprise

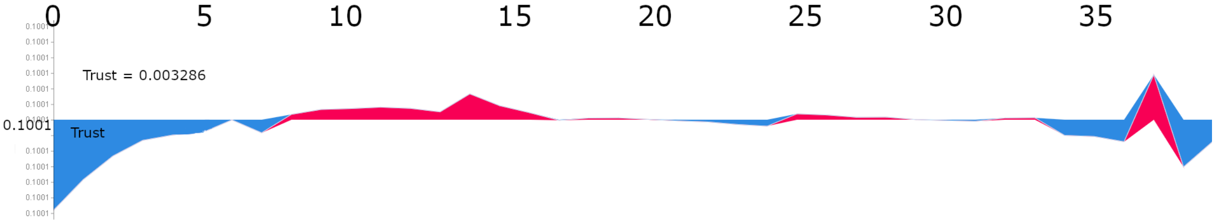
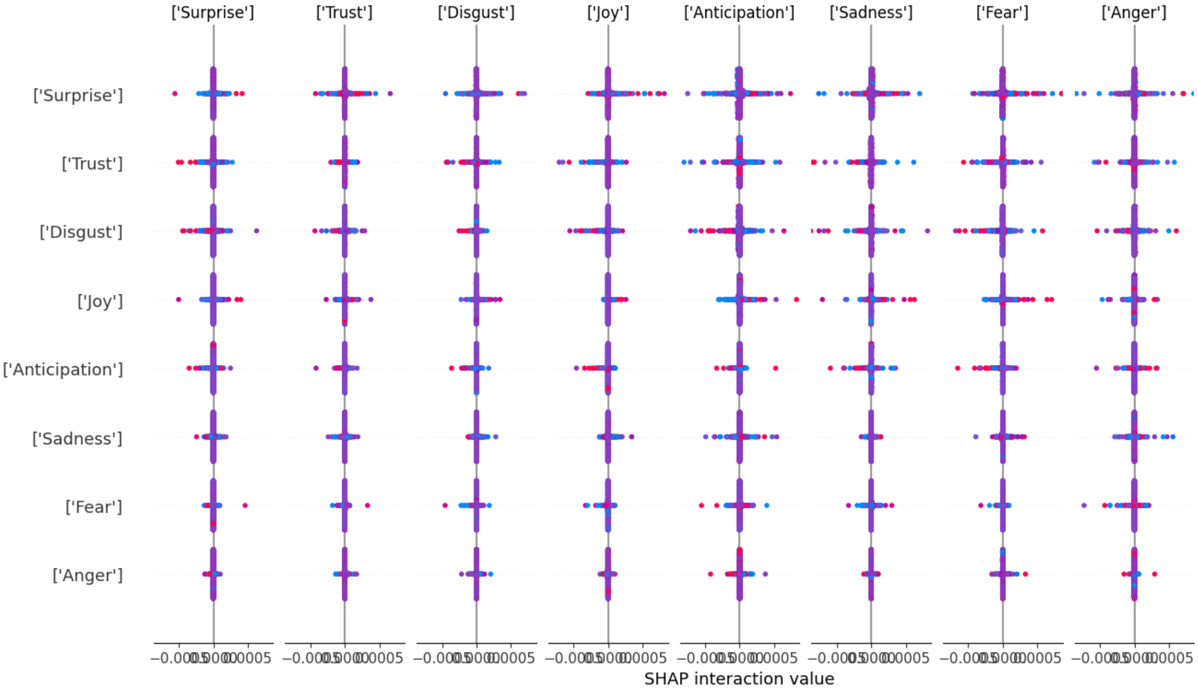
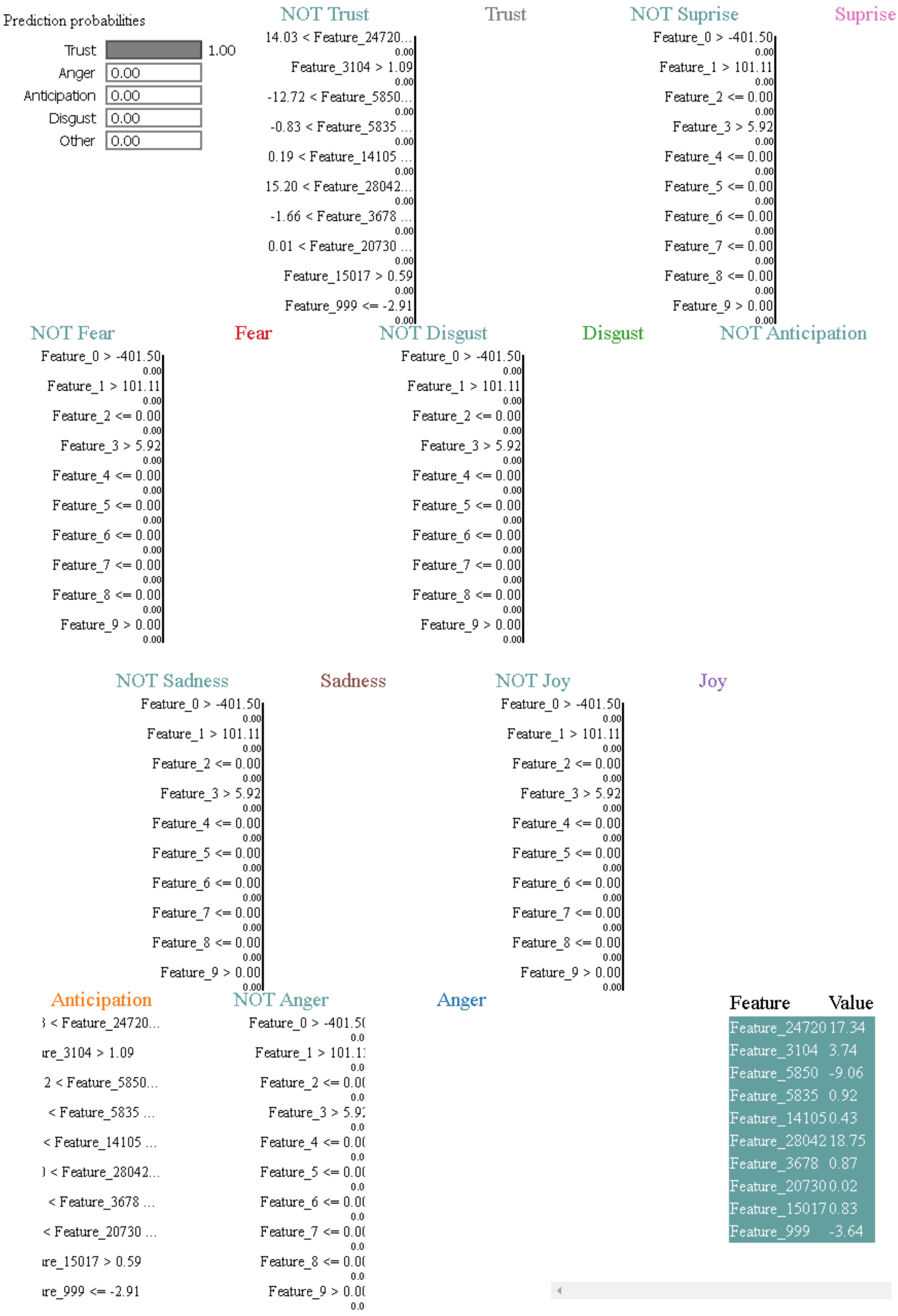


Figura 34: Ensemble Stacking SHAP Force Plot Trust



]

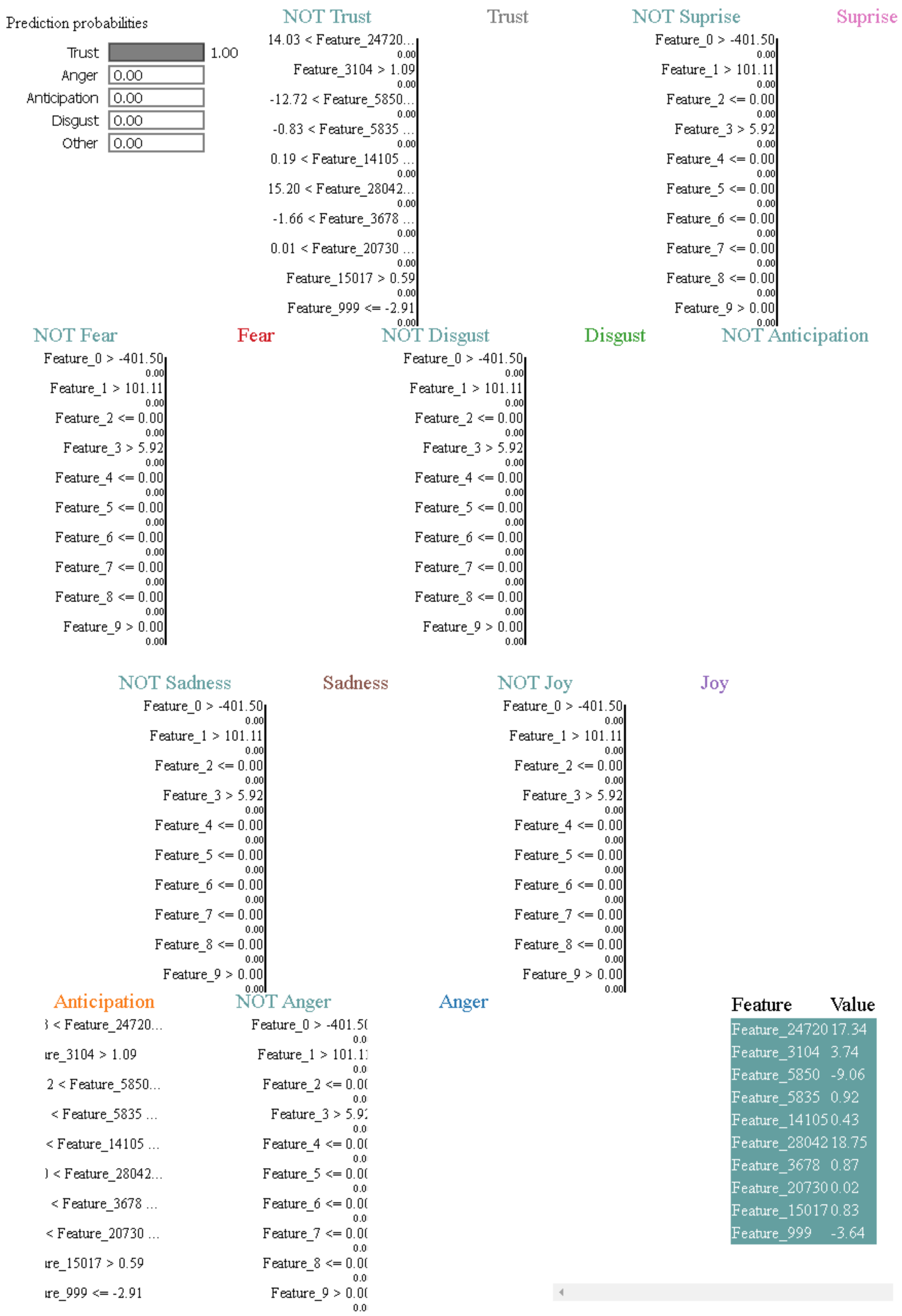
Figura 35: Ensemble Stacking SHAP Summary Plot



Feature	Value
Feature_24720	17.34
Feature_3104	3.74
Feature_5850	-9.06
Feature_5835	0.92
Feature_14105	0.43
Feature_28042	18.75
Feature_3678	0.87
Feature_20730	0.02
Feature_15017	0.83
Feature_999	-3.64

]

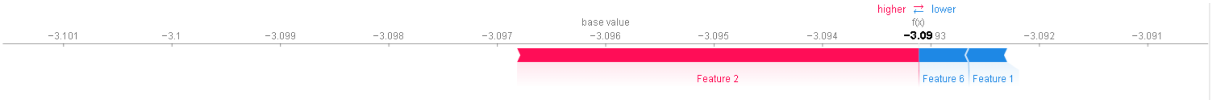
Figura 36: Ensemble LIME prediction plot



Feature	Value
Feature_24720	17.34
Feature_3104	3.74
Feature_5850	-9.06
Feature_5835	0.92
Feature_14105	0.43
Feature_28042	18.75
Feature_3678	0.87
Feature_20730	0.02
Feature_15017	0.83
Feature_999	-3.64

]

Figura 37: Ensemble LIME feature prediction plot



]

Figura 38: XGBoost Force plot

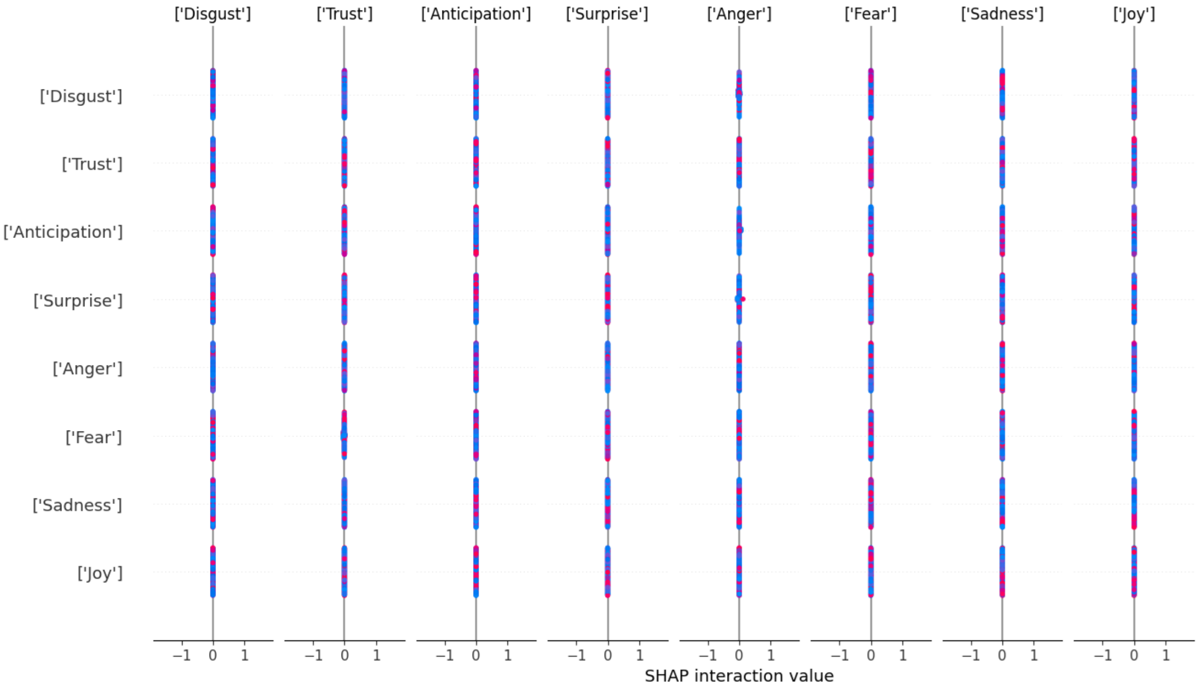
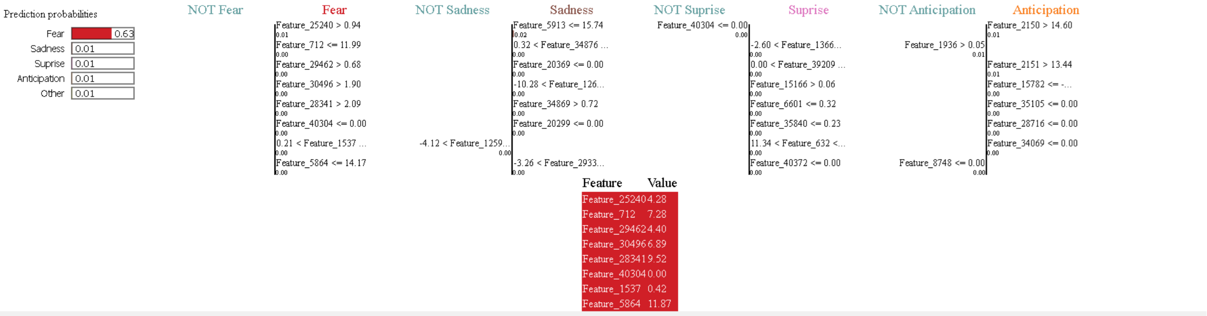


Figura 39: XGBoost SHAP Summary Plot



Feature	Value
Feature_25240	4.28
Feature_712	7.28
Feature_29462	4.40
Feature_30496	6.89
Feature_28341	9.52
Feature_40304	0.00
Feature_1537	0.42
Feature_5864	11.87

Figura 40: XGBoost LIME Explainer Plot