Journal of Soft Computing Exploration
Homepage: htttps://shmpublisher.com/index.php/joscex
p-ISSN: 2746-7686
e-ISSN: 2746-0991

.

Enhancing sarcasm detection via multimodal learning: A BiLSTM-attention approach with text and emojis integration

Nasa Zata Dina[1], Moch. Nafkhan Alzamzami[2]
[1]Department of Engineering, Faculty of Vocational Studies, Universitas Airlangga, Indonesia
[2]Department of Informatics, Faculty of Intelligent Electrical and Informatics Technology, Institut Teknologi Sepuluh Nopember, Indonesia

Article Info

Article history:

Received April 10, 2026
Revised April 13, 2025
Accepted April 28, 2025

Keywords:

Sarcasm detection
Multimodal learning
BiLSTM
Attention mechanism
Natural language processing
Artificial intelligence

ABSTRACT

The detection of sarcasm is a difficult task in Natural Language Processing (NLP) because to the presence of implicit meaning and contextual ambiguity. This is particularly problematic in social media, where emojis are used frequently to indicate tone and intent. The study proposes a multimodal deep learning strategy that combines both textual and emoji features, by utilizing a BiLSTM with attention mechanisms. The goal of this method is to improve the performance of sarcasm detection. The model makes advantage of bidirectional contextual learning and preferentially focuses on informative tokens and emojis in order to do more effective work of capturing complex expressions. According to the findings of the experiments, the Text+Emoji model that was proposed achieves an F1-score of 96.44%, an accuracy of 97.08%, and an area under the curve (AUC) of 99.23%, which is a significant improvement over the unimodal baselines. Future research will focus on enhancing the proposed model by investigating transformer-based architectures to achieve deeper and more contextualized representation learning.

Corresponding Author:

Nasa Zata Dina,
Department of Engineering, Universitas Airlangga,
Kampus B Fakultas Vokasi, Jl. Dharmawangsa Dalam, 28-30, Surabaya, 60286, Indonesia.
Email: nasazatadina@vokasi.unair.ac.id
https://doi.org/10.52465/joscex.v7i2.49

1. INTRODUCTION

Sarcasm detection is still a difficult task in Natural Language Processing (NLP), as sarcasm detection is based on implicit meaning, contextual incongruity, and nuanced linguistic clues. This means that sarcasm, unlike a literal expression, often means the opposite of what is openly expressed, making it challenging for computational models to appropriately interpret. Conventional techniques that solely utilize textual features frequently fail to capture the particulars in real-world applications like sentiment analysis, opinion mining, and social media monitoring, based on their experience [1]. In informal communication contexts, the usage of distinctive and unusual language by users amplifies this issue.

Social media have proliferated rapidly, resulting in communication that now encompasses a variety of multimodal features, including emoticons, pictures, hashtags, and more. Emojis play a vital role in expressing emotions, tone, and purpose. They frequently function as contextual modifiers that change or even invert the meaning of a statement. Recent study indicates that emojis can improve the comprehension of user

intent, especially in sentiment and sarcasm processing tasks [2]–[5]. Numerous existing studies underestimate or regard emojis as simple additional features, hence limiting models' capacity to utilize their emotional and semantic potential. Deep learning models, namely Bidirectional Long Short-Term Memory (BiLSTM) networks, have demonstrated significant capability in modeling sequential relationships in textual data by incorporating both past and future context [6]. The integration of attention mechanisms, including additive attention and self-attention, enhances performance by selectively prioritizing the most informative segments of the input sequence. Nonetheless, research remains restricted about the interaction of attention mechanisms with multimodal inputs, particularly the combination of text and emojis, which increase sarcasm detection capabilities.

The existing literature on sarcasm detection demonstrates a clear progression in both modality and methodology, which directly informs the objectives of this study. Early work by Schifanella et al. [7] explored sarcasm detection in multimodal social platforms by incorporating textual and visual, highlighting the importance of cross-modality, however, the approach primarily relied on traditional feature engineering techniques and did not consider emojis as a distinct modality. In contrast, Poria et al. [8] focused on text-only sarcasm detection and introduced deep convolutional neural networks (CNNs) to capture semantic representations, marking a shift toward neural-based methods but without leveraging multimodal information. Extending this line of research, Hazarika et al. [9] proposed the CASC model, which integrates contextual information from conversation threads, allowing a deeper understanding of sarcasm although it remains limited to textual modality. Recent advancements further reflect a shift from conventional transformer-based models toward large language models (LLMs) with enhanced generalization capabilities, though gaps remain in multimodal integration and evaluation rigor. Kumar et al. [10] utilized a transformer-based architecture that incorporates contextual cues within textual data, demonstrating improved performance over earlier neural approaches while remaining task-specific and unimodal. Extending this direction, Chen et al. [11] introduced an approach that combines LLMs with transfer learning for cross-domain sarcasm detection, focusing on improving robustness and adaptability across different datasets and domains. However, these approaches predominantly emphasize textual understanding and often overlook the role of emojis as complementary signals, as well as the explicit evaluation of attention mechanisms in multimodal settings. Furthermore, limited attention has been given to comprehensive metric-based evaluation to ensure robustness and reliability across diverse scenarios. Therefore, this study addresses these gaps by exploring emojis as a supplementary modality, developing a multimodal deep learning model integrating text and emoji features, evaluating the effectiveness of attention mechanisms, and employing a comprehensive set of evaluation metrics to ensure reliable performance assessment. Considering these challenges, the aims of this study are to (1) explore how emojis can help as an supplementary modality for detecting sarcasm; (2) develop a multimodal deep learning model that incorporates text and emoji modalities; (3) evaluate the effectiveness of attention mechanisms on model's performance for sarcasm detection; and (4) review the evaluation process by performing a comprehensive metric evaluation to ensure robustness and reliability. The following is a discussion of the full research workflow: The suggested approach is described in Section 2. The results of the experiment are discussed in Section 3. The paper's conclusion is discussed in Section 4.

2. METHOD

Data collection, data preprocessing, data labeling, feature extraction, and classification are the five stages of the proposed model. Figure 1 shows the end-to-end workflow. The pre-processing step involves removing numbers, tokenizing sentences, lemmatizing words, converting text to lowercase, and eliminating stop words. The input data is first gathered from open-source datasets like Facebook and Twitter reviews. Data labeling is carried out using valence scores, label encoding, compound scores, and polarity labels following data pre-processing. The review dataset is used in this study. There are 22,290 emoji-filled texts in this dataset, with 9,245 records classified as sarcastic and 13,045 records classified as non-sarcastic. We speculate that a review data intends to express a strong emoji if it uses a single emoji or repeatedly uses an emoji in text. Therefore, the most popular emoji might be able to convey the underlying irony in the text. Therefore, this work uses attention mechanisms and sarcasm-aware emoji embeddings to allow machines to identify sarcasm even when users use the emojis in ways that deviate from their intended meaning.

To investigate how adding emojis affects sarcasm classifier performance, we test and compare three models that we run on BiLSTM attention architectures. Text-only and emoji-only models are contrasted with the proposed model. The following is a list of the models: (1) Model-1 is text-only. There is solely text data in this baseline model. In this model, we remove all emoticons from the text in the dataset, leaving only the words; (2) Model-2 is just emojis. Only emoji data is used in this baseline model. In this model, the dataset's text is removed, leaving only the emoji; (3) Model-3 is emoji and text. The text in the dataset that contains emojis is used by this model.

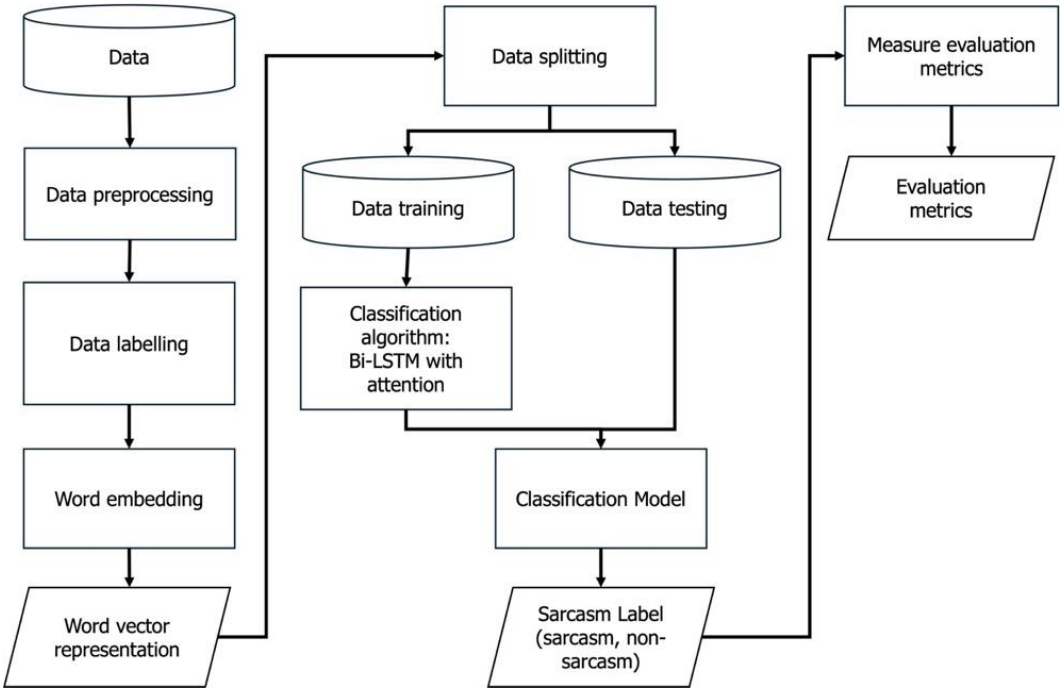

Figure 1. End-to-end workflow of the proposed model

Data Preprocessing

Using the correct pre-processing techniques increases the number of instances that can be successfully identified, making pre-processing the data a crucial step. A range of techniques are used in the preprocessing stage to transform unprocessed textual input into a format appropriate for additional analysis. Preprocessing techniques include tokenization, lemmatization, deleting stop words, removing punctuation and numerical values, and converting lowercase letters [12].

Eliminating numbers and punctuations

To allow the reader to focus solely on textual information, the numerals and punctuations are removed from the text so the text can be more readable and consistent. This process removes all numeric characters (0-9) and punctuation symbols (e.g., commas, periods, exclamation marks) from the text. The objective is to minimize noise while maintaining only significant lexical tokens, hence enhancing consistency and streamlining subsequent text analysis activities such as tokenization and feature extraction [13][14].

Conversion of lowercase

The text consistently use various capitalizations to highlight proper nouns and the beginnings of sentences. This stage transforms all text into lowercase. It reduces redundancy and improves the reliability of textual analysis. Converting text to lowercase enhances output homogeneity, rendering it beneficial for diverse text mining and natural language processing (NLP) applications [15]–[17].

Tokenization

Tokenization is the process of converting text into tokens prior to transforming them into vectors. This stage facilitates systematic analysis by converting unstructured text into manageable parts that can subsequently be handled in later stages, including stop word removal, normalization, and feature extraction [18]–[21]. This essential phase in text data analysis assists in the classification of terms from the text.

Lemmatization

Lemmatization is the process of reducing words to their lemma or root form [22]. The objective is to standardize various inflected forms of a word to facilitate study and comparison. In contrast to stemming, which only shortens words, lemmatization takes into account the morphological and contextual significance of a

word, guaranteeing that the resulting lemma is linguistically accurate. This process improves consistency and reduces redundancy in textual data. This is very useful for NLP and text analysis.

Stop words removal

Words that are usually removed before NLP are known as stop words [23]–[25]. Stop words in English include articles, prepositions, and conjunctions. Because fewer tokens are needed for training, removing stop words undoubtedly decreases the dataset and shortens the training time. The example of preprocessing technique's output can be seen in Figure 2.

Every step in the data preprocessing, as shown in Table 1, helps with the preprocessing of the data. First, the punctuation and number are removed in Stage-1. The second stage involves converting all text to lowercase. Unlike some earlier techniques, Stage-3 involves tokenization. Lemmatization is introduced at Stage-4 to preserve the semantic meaning of words. Stop words are eliminated in Stage-5. The cleaned reviews are then added to the dataset for further processing.

Table 1. Sample output for preprocessing technique

Stage	Preprocessing Technique	Text Output
1	Eliminating numbers and punctuations	Learning from my guru Happy teacher Day
2	Conversion of lowercase	learning from my guru happy teacher day
3	Tokenization	[learning, from, my, guru, happy, teacher, day]
4	Lemmatization	[learn, from, my, guru, happy, teacher, day]
5	Stop words removal	[learn, guru, happy, teacher, day]

Data Labeling

Data labelling in this study is divided into two stages which are sentiment labelling (emoji-based and text-based sentiment) and sarcasm labelling. Data labeling is the process of adding labels in order to produce a labelled dataset for training models. Because machine learning algorithms require ground truth labels for testing and training, data labelling is crucial. First, sentiment polarity is assigned independently to both textual and emoji modalities. For textual content, a lexicon-based approach is employed to classify each instance into positive, negative, or neutral. Similarly, emoji sentiment is derived using an established emoji sentiment lexicon, where each emoji is mapped to a corresponding polarity score and aggregated at the instance level. It is important to emphasize that these sentiment labels are not directly used as sarcasm detection. Instead, they function as auxiliary signals to capture sentiment incongruity, which is a defining characteristic of sarcastic expression.

The sarcasm labeling is then constructed based on the relationship between the two modalities. Specifically, an instance is labeled as "sarcastic" when there is a polarity mismatch between textual sentiment and emoji sentiment, for example, positive text accompanied by negative emojis, or negative text with positive emojis. Conversely, instances exhibiting sentiment consistency across modalities are labeled as non-sarcastic. Neutral label is handled by considering their relative contribution to polarity alignment, instances with ambiguous or weak sentiment signals are excluded or assigned based on dominant polarity to reduce noise.

Emoji-based sentiment labelling

The process for extracting the emojis from the dataset is depicted in Figure 2. We use the data from the current emoji sentiment lexicon developed by [26]–[29] to generate the emoji sentiment lexicons. We only chose a subset of the corpus's emoji-containing data by looking for any emoji-containing data. To create our unlabeled emoji sentiment lexicon, we follow the steps as follows: (a) to extract emoji characters and translate them into a Unicode representation. Characters are converted into bytes using a Unicode string encoding standard using regular expressions.; (b) to prevent duplication in the emoji sentiment lexicons, we compile a list of distinct emojis. This implies that self-repeating emojis only occur once in the vocabulary.; (c) to search the current emoji sentiment lexicon and use word sentiment lexicons to determine the polarities of the words based on their descriptions without the need for human intervention.

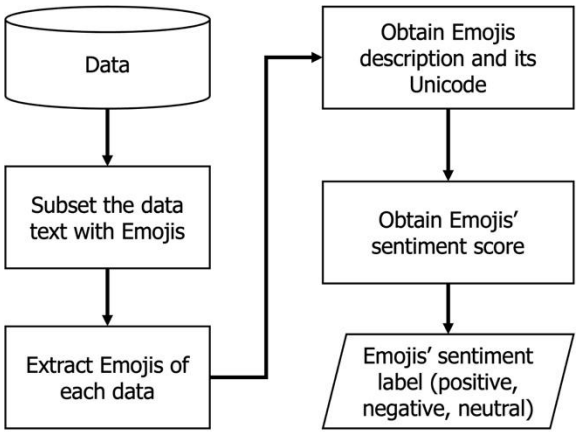

Figure 2. A process to extract Emojis' sentiment label

Text-based sentiment labelling

Many sentiment lexicons have been created in a variety of ways, such as automatically and semi-automatically, as well as manually, which is seen to be a costly and time-consuming operation. Sentiment lexicons are usually collections of words that have sentiment polarity ascribed to them. Numerous sentiment classification has employed sentiment lexicons to ascertain the texts' semantic orientation [30], [31]. These sentiment lexicons have shown that a sentence's polarity values can be combined. These sentiment lexicons have shown that sentiment may be computed on a continuous scale by combining the polarity values from a sentence [32]. Annotators validated our sentiment English lexicons. Additionally, throughout the sentiment annotation process, the annotators marked some of the sentiment-bearing terms.

Emoji and text-based sentiment

We suggest the following algorithm for emoji and text-based sentiment labelling, as shown in Figures 3, 4, 5, and 6, which uses texts and emojis for sentiment ranking [26], [27], [33]. The steps are as follows: (a) find and extract a subset of data containing emojis, use emoji Unicode as seen in Table 2.; (b) categorize data containing sentiment-bearing emojis into three groups: negative, neutral, and positive as depicted in Table 3.; (c) create lists of words that express sentiment and use the translated word lexicon to assign a score as illustrated in Table 4, gather every word from data that are positive, neutral, and negative; and eliminate terms that appear in one or both of the other lists.; (d) based on the highest word coverage with the lists of sentiment-bearing words.; (e) categorize the remaining data without sentiment-bearing emojis, for example, data without sentiment-bearing emojis, into the classifications negative, neutral, and positive using the emoji sentiment lexicon scores from the emoji sentiment lexicon, classify all the data with and without sentiment-bearing emojis into the classes negative, neutral, and positive based on the highest word coverage with the lists of sentiment-bearing words.

Table 2. The example of emoji's unicode and its sentiment.

Emoji	Unicode	Negative	Neutral	Positive	Sentiment score	Unicode name
😍	0x1f60d	0.052	0.219	0.729	0.678	Smiling face with heart-shaped eyes
😭	0x1f62d	0.436	0.220	0.343	-0.093	Loudly crying face
😘	0x1f618	0.053	0.193	0.754	0.701	Face throwing a kiss
😊	0x1f60a	0.060	0.237	0.704	0.644	Smiling face with smiling eyes

Table 3. Classification based on emoji

Review data	Emoji	Negative	Neutral	Positive	Sentiment score	Sentiment label
That's fantastic. 😍	😍	0.052	0.219	0.729	0.678	Positive

😭 all the feels. TAKE MY MONEY	😭	0.436	0.220	0.343	-0.093	Negative
Cuteness overload 😘	😘	0.053	0.193	0.754	0.701	Positive
Almost looks like mine 😊	😊	0.060	0.237	0.704	0.644	Positive

Table 4. Classification based on text

Review data	Negative	Neutral	Positive	Sentiment score	Sentiment label
That's fantastic.	0.0	0.256	0.744	0.5983	Positive
all the feels. TAKE MY MONEY	0.0	1.0	0.0	0.0000	Neutral
Cuteness overload	0.0	0.364	0.636	0.4404	Positive
Almost looks like mine	0.0	1.0	0.000	0.000	Neutral

Sarcasm labelling

The sarcasm labeling procedure begins by converting both textual content and emojis into continuous sentiment scores rather than simple categorical labels. Specifically, each text is assigned a sentiment score within a range from -1 (strongly negative) to +1 (strongly positive), and the emojis are similarly aggregated into a sentiment score within the same range. These continuous scores are then transformed into discrete polarity labels using a small threshold value. If the sentiment score is greater than the threshold, it is considered positive; if it is less than the negative of the threshold, it is considered negative; and if it falls within the threshold range around zero, it is treated as neutral. This process produces three possible polarity values for both text and emoji: positive, negative, or neutral.

Sarcasm is subsequently determined based on the relationship between the polarity of the text and the emoji. An instance is labeled as sarcastic when the text and emoji express opposite sentiments, for instance, one is positive while the other is negative. In contrast, if both modalities share the same sentiment orientation (both positive or both negative), the instance is labeled as non-sarcastic.

Special consideration is required for cases involving neutral sentiment. To avoid ambiguity, a common and recommended approach is to remove instances where either the text or the emoji is neutral. Alternatively, if retaining more data is necessary, the final label can be determined by relying on the modality with the stronger sentiment, or by conservatively labeling all neutral-involved cases as non-sarcastic.

Finally, a confidence filtering step is applied to improve label reliability. Only samples with sufficiently strong sentiment signals are retained, meaning that both text and emoji scores must exceed a predefined minimum magnitude. It helps eliminate weak or ambiguous cases that may introduce noise.

Classification: Deep-layered Architectures and Hyperparameters

To assess the effectiveness of the proposed approach, experiments were conducted using a Bidirectional Long Short-Term Memory (BiLSTM) network enhanced with a self-attention mechanism [34]–[36]. The attention mechanism adopted in this study is additive in nature, employing a hyperbolic tangent (tanh) transformation combined with a learnable context vector to compute attention weights over the BiLSTM hidden states. Model hyperparameters were optimized a grid search strategy. The best-performing configuration includes 16 attention units, a dropout rate of 0.02, the Adam optimizer with a learning rate of 0.01, and a sigmoid activation function in the output layer. Although configurations with 16 and 32 attention units yielded comparable performance, 16 units were selected due to their more favorable balance between predictive performance and computational efficiency, while still outperforming the configuration with 8 units. To reduce overfitting, all models were trained for 25 epochs with a batch size of 32, and early stopping was applied with a patience of 6 epochs based on validation loss. The overall architecture of the proposed BiLSTM with self-attention model is illustrated in Figure 3.

To effectively capture contextual dependencies in sequential data, the BiLSTM processes the input sequence in both forward and backward directions, producing a sequence of hidden representations, where each representation encodes contextual information derived from both preceding and succeeding tokens. To further highlight the most informative parts of the sequence, a self-attention layer with a learnable context vector is applied on top of these BiLSTM outputs. Rather than relying on uniform or fixed pooling strategies, this mechanism assigns adaptive importance weights to each time step.

Specifically, for each hidden state, an attention score is computed by applying a linear transformation followed by a tanh activation function, where the transformation involves a trainable weight vector and a bias term. These scores are then normalized across all time steps using a softmax function, producing attention weights that sum to one and reflect the relative importance of each hidden state. The final context vector is

obtained by computing a weighted sum of all hidden states, where each hidden state is scaled by its corresponding attention weight. This context vector captures the most relevant semantic information in the sequence and is subsequently passed to a dense layer with a sigmoid activation function for binary classification.

It is important to note that the employed attention mechanism is additive, as it relies on a nonlinear tanh transformation with learnable parameters to derive attention scores. However, unlike classical Bahdanau or Luong attention mechanisms, the proposed approach does not explicitly model query key interactions. Instead, it can be interpreted as a simplified self-attention mechanism with a single global context vector, which has been shown to be effective for sequence representation learning.

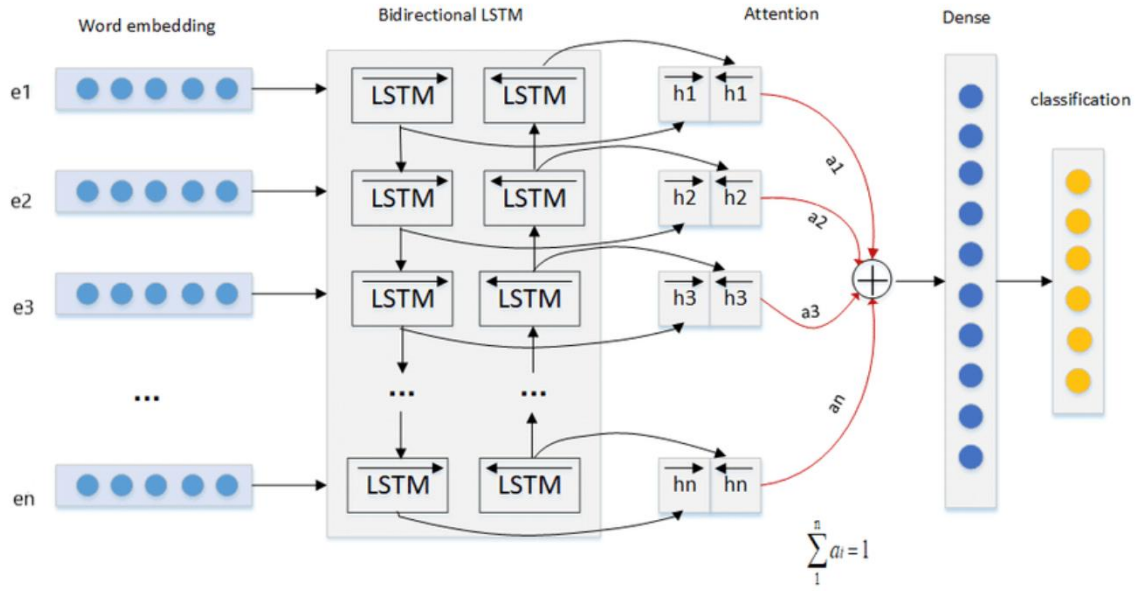

Figure 3. Bi-LSTM with attention [37]

Performance Metrics

A confusion matrix is a table that summarizes the performance of a classification model by comparing its predicted labels to the true labels. Figure 4 shows a confusion matrix table.

		Actual values	
		True	False
Predicted values	True	True Positive (TP)	False Positive (FP)
	False	True Negative (TN)	False Negative (FN)

Figure 4. Confusion matrix

Accuracy measures a model's performance by quantifying how often predictions align with true labels [38], [39]. It is calculated as Eq. (1).

$$\text{Accuracy} = \frac{1}{m}\sum_{i=1}^{m}\frac{TP_i+TN_i}{TP_i+TN_i+FP_i+FN_i} \quad (1)$$

Precision represents the average proportion of ground-truth relevant labels among those predicted as relevant [40]. It is calculated as Eq. (*2*).

$$\text{Precision} = \frac{1}{m}\sum_{i=1}^{m}\frac{TP_i}{TP_i+FP_i} \quad (2)$$

Recall represents the average proportion of ground-truth relevant labels that are correctly identified [41]. It is calculated as Eq. (3).

$$\text{Recall}= \frac{1}{m}\sum_{i=1}^{m}\frac{TP_i}{TP_i+FN_i} \quad (3)$$

The F1-score represents the harmonic mean of precision and recall, providing a balanced measure of both [42]. It is calculated as Eq. (3).

$$\text{F1-score}= \frac{2\ x\ Recall\ x\ Precision}{Recall+Precision} \quad (4)$$

Area Under Curve (AUC) measures the area under the ROC curve [43]. A higher AUC value indicates better model performance as it suggests a greater ability to distinguish between classes. An AUC value of 1.0 indicates perfect performance while 0.5 suggests it is random guessing.

3. RESULTS AND DISCUSSIONS

Description of The Dataset

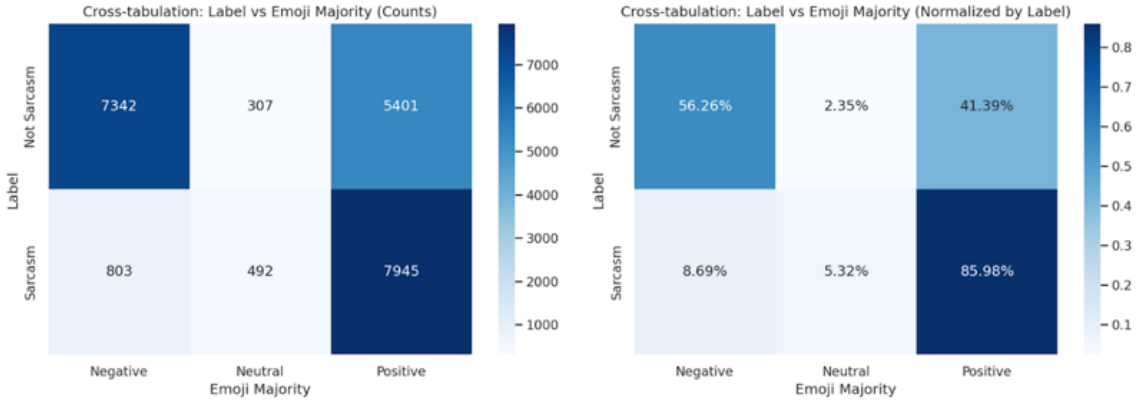

Figure 5. Sarcasm labels vs emoji sentiment distribution on dataset

Figure 9 shows the correlation between sarcastic labels and the predominant emoji sentiment. Non-sarcastic text is mostly linked to negative emojis (7342 instances) and positive emojis (5401), with very few neutral examples (307) in the count-based heatmap. Positive emojis (7945) predominate in sarcastic text, but negative (803) and neutral (492) emojis are far less common. The normalized heatmap makes this tendency much more evident: 85.98% of sarcastic instances are linked to positive emojis, whereas just 8.69% are negative and 5.32% are neutral. Non-sarcastic content, on the other hand, is more evenly divided, with only 2.35% neutral and 56.26% negative and 41.39% positive emojis. Non-sarcastic content displays a more uniform distribution of emojis, however the overall data indicates a substantial inclination for sarcasm to coincide with positive emojis. This undoubtedly underscores the disparity between the expressed sentiment and genuine intent. Figure 5 illustrates the correlation between the primary utilization of emojis and classifications of sarcasm through both count-based and normalized distributions. Results indicate a substantial correlation between the use of positive emojis and caustic discourse. Sarcastic instances are much more associated with positive emojis (7945) in the count-based analysis than with negative (803) and neutral (492) emojis. The statistics, which show that 85.98% of sarcastic data equates to positive emojis, further support this tendency. Non-sarcastic text, on the other hand, exhibits a more evenly distributed distribution, with 56.26% negative and 41.39% positive emojis, and just 2.35% neutral usage. These results imply that sarcasm frequently depends on a polarity contrast, in which ironic or opposing interpretations to the textual content are conveyed through the use of positive emojis. Emoji information is therefore a crucial contextual indicator for sarcasm identification.

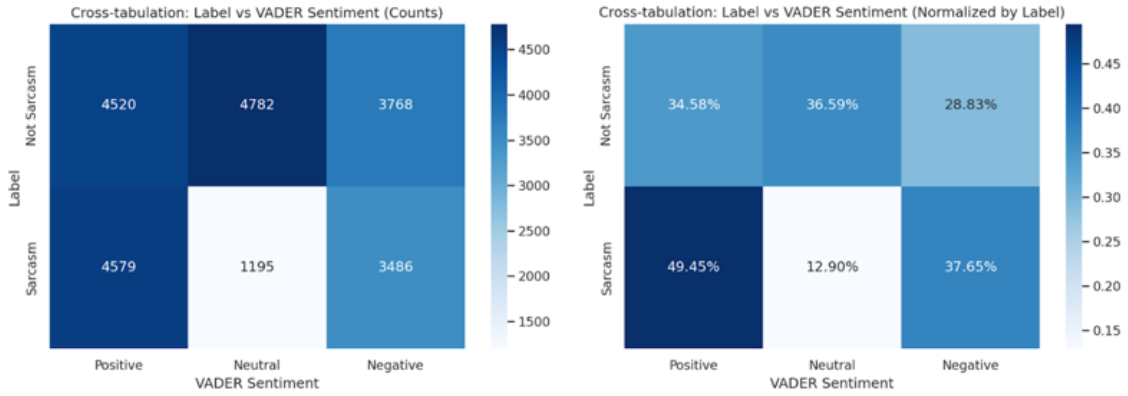

Figure 6. Sarcasm labels vs text sentiment distribution on dataset

Figure 6 illustrates the relation between sentiment polarity and sarcastic labels on text dataset. Non-sarcastic content is very evenly distributed across sentiments in the count-based heatmap, with neutral sentiment being the most common (4782 instances), followed by positive (4520) and negative (3768). Sarcastic material, on the other hand, exhibits a different pattern: neutral sentiment is far less common (1195), while positive sentiment predominates (4579), followed by negative sentiment (3486). The normalized heatmap makes this distribution more evident: sarcastic incidences are mostly linked to positive sentiment (49.45%), negative sentiment (37.65%), and neutral sentiment (12.90%). In contrast, the distribution of non-sarcastic content is more uniform, with 28.83% negative, 34.58% positive, and 36.59% neutral. The results indicate that non-sarcastic text demonstrates an equal sentiment distribution, but sarcasm is more commonly associated with positive sentiment scores. This likely signifies an inconsistency between reported sentiment and the actual intention. Figure 6 illustrates the cross-tabulation of sentiment polarity and sarcasm labels. The sentiment distributions reveal a more complex pattern than those obtained from emoji-based study. Non-sarcastic data is balanced, with neutral sentiment slightly predominating (36.59%), followed by positive (34.58%) and negative (28.83%). Sarcastic content, however, has a stronger correlation with positive sentiment (49.45%), negative sentiment (37.65%), and neutral sentiment (12.90%). This implies that sarcastic expressions tend to show greater sentiment polarity and are less likely to look neutral. The concept of semantic incongruity, where the surface attitude deviates from the intended meaning, is further supported by the predominance of positive sentiment in sarcastic contexts.

Results and Discussion

Performance comparison of the proposed models across three modalities can be seen in Figure 11. There are three modalities: Text-only, Emoji-only, and the Text+Emoji model that are evaluated using accuracy, F1-score, precision, recall, and AUC. The multimodal strategy regularly overcomes the unimodal strategy across all evaluation metrics. The Text+Emoji model demonstrates its resilience and great generalization capacity by achieving the highest accuracy (97.09%), F1-score (96.44%), recall (95.61%), and AUC (99.24%). This suggests that the model's capacity to identify complex patterns in the data is greatly improved by incorporating textual and emotional modalities. Among three models compared in this study, the Text-only model performs the best in terms of precision (98.64%), indicating that it produces accurate positive predictions. Its low recall (70.93%), however, indicates that it is unable to identify true positive instance, leading to a less balanced overall performance. On the other hand, compared to the Text-only model, the Emoji-only model performs mediocrely, with a greater recall (84.27%), but poorer precision (77.74%), F1-score (80.87%), and accuracy (83.54%), indicating limitations when depending only on sentiment. The results indicate that the unimodal model captures only a part of the data. The multimodal model offers a more thorough representation, resulting in enhanced and more consistent performance across all evaluation metrics. These results highlight how necessary it is to combine various modalities in order to successfully handle complex tasks like sarcasm detection. Multimodal approaches are crucial for sarcasm detection, as the integration of data from many modalities enhances the classification model. The Text+Emoji model's enhanced efficacy illustrates that neither textual nor emotional features alone can adequately express the complexity of sarcasm. Emoji and emotions exhibit contrasting polarity patterns that are challenging to identify with a single modality.

The emoji-based model may produce ambiguity [44], whereas the text-based model can achieve great precision but often lacks contextual awareness [45]. The model exhibits superior performance overall due to its enhanced ability to detect implicit clues and anomalies, related to the integration of both modalities. These findings underscore the importance of employing multimodal data necessary for complex language comprehension tasks such as sarcasm detection.

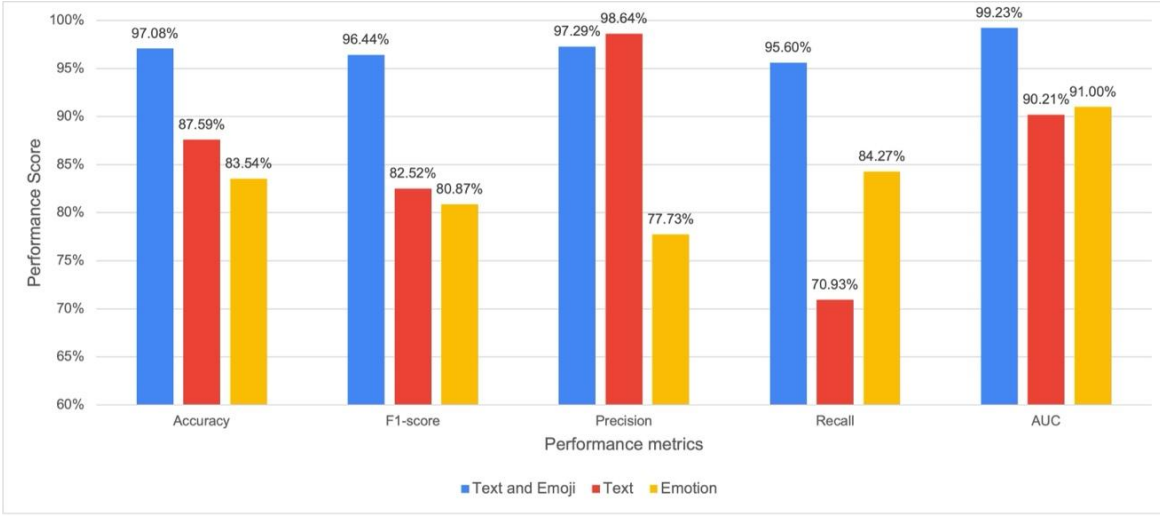

Figure 7. Results of proposed model on the dataset

The features of the BiLSTM with attention mechanisms, designed to capture sequential dependencies and contextual significance within input data, reveal the performance variances between modalities. Figure 7 demonstrates that the Text+Emoji modality surpasses both text-only and emoji-only inputs, attaining the highest overall performance (F1-score: 96.44%, AUC: 99.23%). The combination advantages of the attention mechanism and BiLSTM contribute for this enhancement. The model may capture contextual dependencies from both the past and future, as the BiLSTM component processes the input sequence bidirectionally. This is effective for textual content; however, the inclusion of emojis significantly enhances its efficacy, as emojis frequently function as emotional and semantic modifiers that clarify the meaning of a statement. The attention mechanism augments this BiLSTM by allocating more weight to the most prominent segments of the sequence. In a multimodal context, attention can selectively concentrate on two factors: (a) emotionally significant emojis and (b) contextually relevant words. This allows the model to more effectively address ambiguities like as sarcasm, implicit sentiment, or informal idioms that are challenging to decipher from text alone. The Text+Emoji model has a balanced performance, attaining high precision (97.29%) and high recall (95.60%), resulting in an enhanced F1-score. Conversely, the Text-only model exhibits a high precision of 98.64% but a far lower recall of 70.93%, suggesting it is overly strict and overlooks numerous relevant instances. This indicates that the attention mechanism lacks sufficient signals to identify all pertinent patterns in the absence of emojis.

The emoji-only model, on the other hand, achieves lower precision (77.73%) but balanced recall (84.27%), indicating that while emotional factor help identify relevant instances, they are not sufficiently discriminative when applied alone. This indicates that the optimal performance of attention requires extensive contextual input. Overall, these results demonstrate that the effectiveness of the attention mechanism is significantly impacted by its informative input. The more informative representation created by the combination of text and emojis allows the attention layer to focus on the most relevant semantic and emotional factor. The BiLSTM-attention model's ability to obtain both high accuracy and robustness thus validates the advantage of multimodal learning.

To evaluate the contribution of each input modality, an ablation study was conducted by systematically removing modalities of the multimodal input. An overview of the results is shown in Figure 11. The integrated model (Text+Emoji), which outperforms across all parameters, validates the efficacy of integrating text and emojis. The removal of emojis results in a reduction in the F1-score from 96.44% to 82.52%f, primarily related to a decline in recall. This indicates that emojis are important for identifying factors that may be ambiguous or implicit. Nonetheless, when solely emojis are applied, the model demonstrates balanced recall but diminished accuracy, indicating that emoji features alone are inadequate for precise classification without linguistic context. These results demonstrate that: (a) text provides structural and semantic context; (b) emojis provide emotion; and (c) the combination of Emoji+Text modalities enhances the

effectiveness of the attention mechanism. The ablation results strongly support the performance of the models but rather increase from the complementary interaction across modalities within the BiLSTM-attention.

4. CONCLUSION

The purpose of this study was to examine how emojis may be used to detect sarcasm, create a multimodal deep learning model, and assess how attention mechanisms contribute to sarcasm detection. The results indicate emojis as a useful supplemental modality that enhances textual content by offering emotional and contextual indications that are frequently lacking in plain text. The proposed model performs better than text-only model by applying both modalities to produce a more reliable and context-aware representations. The proposed model achieves an F1-score of 96.44%, an accuracy of 97.08%, and an area under the curve (AUC) of 99.23%. Furthermore, it is demonstrated that the effectiveness of sarcasm detection is improved by the addition of attention processes. It enables the proposed model to interpret complicated relationships between text and emojis by highlighting the importance of the input.

Future research will focus on enhancing the proposed model by investigating transformer-based architectures to achieve deeper and more contextualized representation learning. The integration of additional modalities such as images and contextual metadata will be examined to augment the model's ability to recognize emotions. The study will use cross-domain and multilingual datasets to ensure broader applicability, facilitating a more comprehensive assessment of the model's generalizability. Future study will investigate the application of emoji embeddings to develop context-sensitive sentiment tagging techniques tailored for specialized systems, emphasizing the significance of emojis.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Nasa Zata Dina: Conceptualization, Methodology, Software, Project administration, Supervision, Writing – review & editing. **Moch. Nafkhan Alzamzami**: Software, Validation, Writing – original draft.

DECLARATION OF COMPETING INTERESTS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] R. Dhumpati *et al.*, "Enhancing sarcasm detection in sentiment analysis for cyberspace safety using advanced deep learning techniques," *Sci. Rep.*, vol. 15, p. 22681, 2025.

[2] C. Cui *et al.*, "A survey on multimodal large language models for autonomous driving," in *2024 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2024, pp. 958–979.

[3] D. S. Chauhan, G. V Singh, A. Arora, A. Ekbal, and P. Bhattacharyya, "An emoji-aware multitask framework for multimodal sarcasm detection," *Knowledge-Based Syst.*, vol. 257, p. 109924, 2022.

[4] V. Grover and H. Banati, "An attention approach to emoji-focused sarcasm detection," *Heliyon*, vol. 10, no. 17, p. e36398, 2024.

[5] F. Barbieri, M. Ballesteros, and H. Saggion, "Are emojis predictable?," in *EACL*, 2017, pp. 105–111.

[6] F. Long, K. Zhou, and W. Ou, "Sentiment analysis of text based on bidirectional LSTM with multi-head attention," *IEEE Access*, vol. 7, pp. 141960–141969, 2019.

[7] D. Schifanella, P. de Juan, J. Tetreault, and L. Cao, "Detecting sarcasm in multimodal social platforms," in *ACM Multimedia*, 2016, pp. 1136–1145.

[8] S. Poria, E. Cambria, D. Hazarika, and P. Vij, "A deeper look into sarcastic tweets using deep convolutional neural networks," 2016.

[9] S. Hazarika, S. Poria, E. Cambria, and R. Zimmermann, "CASC: Contextual sarcasm detection in online discussion forums," 2018.

[10] A. Kumar, R. Sharma, and P. Bhattacharyya, "Transformer-based sarcasm detection with contextual cues," *Knowledge-Based Syst.*, vol. 67, no. 9, pp. 7399–7430, 2023.

[11] T. An, P. Yan, J. Zuo, X. Jin, M. Liu, and J. Wang, "Enhancing Cross-Lingual Sarcasm Detection by a Prompt Learning Framework with Data Augmentation and Contrastive Learning," *Electronics*, vol. 13, no. 11, p. 2163, 2024.

[12] M. Siino, I. Tinnirello, and M. La Cascia, "Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on transformers and traditional classifiers," *Inf. Syst.*, vol. 121, p. 102342, 2024.

[13] N. Babanejad, A. Agrawal, A. An, and M. Papagelis, "A comprehensive analysis of preprocessing for word representation learning in affective tasks," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, 2020, pp. 5799–5810.

[14] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of Twitter data," in *Proc. Workshop Language in Social Media (LSM 2011)*, 2011, pp. 30–38.

[15] J. Camacho-Collados and M. T. Pilehvar, "On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis," in *Proc. EMNLP Workshop BlackboxNLP*, 2018, pp. 40–46.

[16] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Inf. Process. Manag.*, vol. 50, no. 1, pp. 104–112, 2014.

[17] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, and N. Bhamidipati, "Hate Speech Detection with Comment Embeddings," in *WWW '15 Companion: Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 29–30.
[18] M. Hassler and G. Fliedl, "Text preparation through extended tokenization," *WIT Trans. Inf. Commun. Technol.*, vol. 37, 2006.
[19] P. McNamee and J. Mayfield, "Character n-gram tokenization for European language text retrieval," *Inf. Retr. Boston.*, vol. 7, no. 1, pp. 73–97, 2004.
[20] S. Vijayarani and R. Janani, "Text mining: Open source tokenization tools-An analysis," *Adv. Comput. Intell. An Int. J.*, vol. 3, no. 1, pp. 37–47, 2016.
[21] L. A. Mullen, K. Benoit, O. Keyes, D. Selivanov, and J. Arnold, "Fast, consistent tokenization of natural language text," *J. Open Source Softw.*, vol. 3, no. 23, p. 655, 2018.
[22] E. Guzman and W. Maalej, "How do users like this feature? A fine-grained sentiment analysis of app reviews," in *Proc. IEEE 22nd Int. Requirements Engineering Conf. (RE)*, 2014, pp. 153–162.
[23] H. P. Luhn, "Key word-in-context index for technical literature (KWIC index)," *Am. Doc.*, vol. 11, no. 4, pp. 288–295, 1960.
[24] H. Saif, M. Fernandez, Y. He, and H. Alani, "On stopwords, filtering and data sparsity for sentiment analysis of Twitter," in *Proc. 9th Int. Conf. Language Resources and Evaluation (LREC)*, 2014, pp. 810–817.
[25] M. Makrehchi and M. S. Kamel, "Automatic extraction of domain-specific stopwords from labeled documents," in *Proc. 30th Eur. Conf. IR Research (ECIR)*, 2008.
[26] P. K. Novak, J. Smailović, B. Sluban, and I. Mozetič, "Sentiment of emojis," *PLoS One*, vol. 10, no. 12, p. e0144296, 2015.
[27] S. A. A. Hakami, R. Hendley, and P. Smith, "Arabic emoji sentiment lexicon (Arab-ESL): A comparison between Arabic and European emoji sentiment lexicons," in *in Proc. 6th Arabic Natural Language Processing Workshop, Kyiv, Ukraine (Virtual)*, 2021, pp. 60–71.
[28] F. Haak, "Emojis in lexicon-based sentiment analysis: Creating emoji sentiment lexicons from unlabeled corpora," in *in Proc. LWDA Workshops*, 2021, pp. 279–286.
[29] M. Fernández-Gavilanes, J. Juncal-Martínez, S. García-Méndez, E. Costa-Montenegro, and F. J. González-Castaño, "Creating emoji lexica from unsupervised sentiment analysis of their descriptions," *Expert Syst. Appl.*, vol. 103, pp. 74–91, 2018.
[30] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," in *Proceedings of the ESWC2011 Workshop on "Making Sense of Microposts": Big Things Come in Small Packages*, 2011, pp. 93–98.
[31] C. J. Hutto and E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. 8th Int. Conf. Weblogs and Social Media (ICWSM)*, 2014, pp. 218–225.
[32] M. Kaity and V. Balakrishnan, "Sentiment lexicons and non-English languages," *Knowl. Inf. Syst.*, pp. 4445–4480, 2020.
[33] J. Kranjc, J. Smailović, V. Podpečan, M. Grčar, M. Žnidaršič, and N. Lavrač, "Active learning for sentiment analysis on data streams: Methodology and workflow implementation in the ClowdFlows platform," *Inf. Process. Manag.*, vol. 51, pp. 187–203, 2015.
[34] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," 2014.
[35] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2015, pp. 1412–1421.
[36] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5–6, pp. 602–610, 2005.
[37] Q. Zhou and H. Wu, "NLP at IEST 2018: BiLSTM-attention and LSTM-attention via soft voting in emotion classification," in *Proc. 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2018, pp. 189–194.
[38] N. Japkowicz, "Assessment metrics for imbalanced learning," in *Imbalanced Learning*, 2013, pp. 187–206.
[39] J. Brownlee, "Failure of classification accuracy for imbalanced class distributions," *Machine Learning Mastery*, 2020. .
[40] S. M. Najem and S. M. Kadeem, "A survey on fraud detection techniques in e-commerce," *Tech-Knowledge*, vol. 1, no. 1, pp. 33–47, 2021.
[41] M. Z. Tazehkandi and M. Nowkarizi, "Three approaches to measuring recall on the web: A systematic review," *Electron. Libr.*, vol. 38, no. 3, pp. 477–492, 2020.
[42] L. A. Jeni, J. F. Cohn, and F. D. La Torre, "Facing imbalanced data: Recommendations for the use of performance metrics," in *Proc. Int. Conf. Affective Computing and Intelligent Interaction Workshops (ACII)*, 2013, pp. 245–251.
[43] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.
[44] E. Mower *et al.*, "Interpreting ambiguous emotional expressions," in *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, Amsterdam, Netherlands*, 2009, pp. 1–8.
[45] P. Naveen and P. Trojovský, "Overview and challenges of machine translation for contextually appropriate translations," *iScience*, vol. 27, no. 10, p. 110878, 2024.