IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 10 July 2024, accepted 5 August 2024, date of publication 9 August 2024, date of current version 20 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3441323

RESEARCH ARTICLE

TransNet: Deep Attentional Hybrid Transformer for Arabic Posts Classification

MD. MITHUN HOSSAIN[1], MD. SHAKIL HOSSAIN[1], MD. SHAKHAWAT HOSSAIN[1], M. F. MRIDHA[2], (Senior Member, IEEE), MEJDL SAFRAN[3], AND SULTAN ALFARHOOD[4]

[1]Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka 1216, Bangladesh
[2]Department of Computer Science, American International University-Bangladesh, Dhaka 1229, Bangladesh
[3]Research Chair of Online Dialogue and Cultural Communication, Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[4]Department of Computer Science, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

Corresponding authors: Mejdl Safran (mejdl@ksu.edu.sa) and M. F. Mridha (firoz.mridha@aiub.edu)

This work was supported by the Research Chair of Online Dialogue and Cultural Communication, King Saud University, Riyadh, Saudi Arabia.

ABSTRACT Sentiment analysis is important for comprehending attitudes and emotions, and popular social media platform X (formally Twitter) is useful in this context. For sentiment analysis of English texts, several approaches are available but the Arabic language calls for more specialized study because of its unique qualities and subtleties. This paper presents TransNet, a Deep Attentional Hybrid Transformer model designed to classify asthma-related Arabic social media messages. TransNet combines the sequential learning capabilities of Gated Recurrent Units (GRUs) and Long Short-Term Memory Networks (LSTMs) with the resilient attention mechanisms of Transformers. The model can extract complex patterns and relationships from textual input efficiently. We use label encoding and upsampling approaches to rectify the intrinsic class imbalance in our Kaggle dataset. We also expand the dataset containing Arabic asthma-related tweets with new data, namely by substituting synonyms to add variety to the training set. To better capture the subtleties of the Arabic language, we further improve text representation by adding a pre-trained Bidirectional Encoder Representations from Transformers (BERT) tokenizer. A rigorous collection of baseline models, including Transformer+LSTM, Transformer+GRU, Transformer+CNN, and Transformer+GRU-CNN, are used to assess TransNet's performance. TransNet demonstrates its supremacy with an F1 score of 97.86% and an excellent accuracy of 97.87% in the empirical data. We use Local Interpretable Model-agnostic Explanations (LIME), which helps to understand the mechanism of the model and also ensures that our forecasts are clear and comprehensible. According to our study, TransNet performs better than conventional models in categorizing Arabic asthma postings on social media and gives useful insights.

INDEX TERMS TransNet, hybrid transformer, transformer, sentiment analysis, Arabic text classification.

I. INTRODUCTION

The emergence of social media platforms has changed the landscape of communication. It gives people a place to share experiences and seek information on a variety of issues, including health and wellness [17]. Asthma, a chronic respiratory ailment that affects millions of people worldwide [6], [73], has emerged as a popular topic of discussion in social media [33]. These platforms are Internet forums where people can share personal experiences, seek support, and voice concerns about asthma treatment, medication, triggers, and lifestyle changes [45]. Understanding the dynamics of these social media debates provides an opportunity for healthcare practitioners and academics to acquire real-time information about public views, practices, and issues related to asthma management [61].

However, the analysis of social media data [27], particularly in languages other than English, offers distinct obstacles for linguistic intricacies and cultural subtleties. Arabic is one of the world's most frequently spoken languages, and

The associate editor coordinating the review of this manuscript and approving it for publication was Prakasam Periasamy.

it also includes asthma-related talks [35], [49]. Arabic-language social media platforms provide lots of information and insights about the attitudes, beliefs, and experiences of people with asthma in Arabic-speaking countries [9], [12], [22], [65]. Despite the abundance of accessible data, the successful processing and interpretation of Arabic text data remains difficult because of the language's complicated morphology, grammar, and many dialects [25], [41], [64].

Traditional machine learning methods struggle to manage the complexities of Arabic text input [28], [39], [43], resulting in a subpar performance in tasks such as classification and sentiment analysis. To solve these issues, using deep learning techniques can be useful [46], [51], [58], [60], which have performed extraordinarily well in processing and analyzing natural language data. Models based on Transformer topologies [11], [42], [50], [57] include self-attention processes and perform well in capturing long-term dependencies and contextual information in text data. Furthermore, recurrent neural networks (RNNs), such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks, excel at modeling sequential input, making them valuable components in text categorization applications [8], [16], [62], [63].

Here, we present TransNet, an innovative deep-attentional hybrid transformer architecture created especially for Arabic text categorization. This model primarily focused on information relevant to asthma. Our strategy takes into account the difficulties posed by limited labeled data and linguistic intricacies by including sophisticated methods for data augmentation to reduce class imbalance and improve model resilience. By utilizing a pre-trained BERT tokenizer, TransNet seeks to improve classification accuracy by identifying the complex linguistic patterns present in Arabic text.

This study's main goal is to assess TransNet's performance in correctly categorizing Arabic text, especially in the context of conversations about asthma. For that reason, we trained TransNet against several baseline models—Transformer+LSTM, Transformer+GRU, Transformer+CNN, and Transformer+GRU-CNN—that are often employed in text categorization tasks. In this study, we used Arabic asthma tweets obtained from Kaggle for extensive testing and analysis and to illustrate TransNet's better performance and possible practical applications. Furthermore, we use the LIME framework to do interpretability analysis, which sheds light on TransNet's decision-making process and improves our comprehension of its categorization skills.

Our study contributes significantly in several ways to the creation of a Hybrid Transformer model designed explicitly for the categorization of posts about asthma.

- We present a unique deep-attentional hybrid transformer architecture designed exclusively for the categorization of Arabic text, concentrating on material linked to asthma.
- Using sophisticated methods like label encoding, upsampling, and synonym substitution for data augmentation, we address the difficulties presented by the unique linguistic properties of Arabic text and guarantee efficient learning from both minority and majority classes.
- We studied TransNet's superiority in accurately classifying Arabic text through a comparative evaluation of TransNet against baseline models such as Transformer +LSTM, Transformer+GRU, Transformer+CNN, and Transformer+GRU-CNN. We measured performance metrics such as accuracy, precision, recall, and F1-score.
- We improved comprehension of TransNet's classification skills and clarified the aspects by using the LIME framework to shed light on the decision-making process.
- Our findings highlight the importance of healthcare informatics in helping Arabic-speaking communities control asthma. We want to improve public comprehension and awareness of respiratory health concerns by using proactive monitoring and intervention strategies assisted by powerful natural language processing technology.

In summary, our unique deep learning model that combines recurrent networks with Transformer attention processes provides a reliable categorization of posts in Arabic about asthma. This study has great potential for early healthcare treatments. By advancing natural language processing in healthcare informatics, this research helps Arabic-speaking people better understand and manage respiratory health.

The remaining sections of the paper are arranged as follows. A literature review is shown in Section II, and Data description and data preprocessing is shown in Section III. The proposed methodology is discussed in Section IV, followed by the Experiment and result analysis in Section V. Section VI shows the discussion, Section VII shows Limitations and Future work, and Section VIII concludes the paper with future work.

II. LITERATURE REVIEW

We do extensive literature research to understand the areas of development and deficiency in the Arabic asthma tweet categorization field. This paper covers several topics, such as sentiment analysis, social media conversations on asthma, and the use of machine learning methods for text classification in Arabic language contexts. This section helped us to understand the current patterns, obstacles, and prospects in the field of Arabic asthma tweet categorization. Our suggested categorization approach is based on this exploration, which advances the fields of sentiment analysis and healthcare informatics in Arabic-speaking societies.

A. MACHINE LEARNING & DEEP LEARNING APPROACH

Khelil et al. [52] evaluated classifiers such as K-nearest Neighbor, Support Vector Machine (SVM), Logistic Regression (LR), and Naïve Bayes on Arabic customer reviews.

They emphasized the impact of stemming on classification performance in their study. According to the study, the combination of Snowball Stemmer, SVM, and LR achieved the highest accuracy of 91%. Abdulsalam et al. [2] highlighted AraBERT's effectiveness in detecting suicidal content in Arabic tweets, achieving an F1 score of 88% and 91% accuracy. Al Mahmoud et al. [56] proposed a multiclass sentiment classification approach using clustering-based undersampling and ensemble learning to handle imbalanced datasets. Alqahtani et al. [20] introduced a resource-free unsupervised self-labeling adaptation framework for Arabic sentiment classification, leveraging feature selection methods and hybrid word pairwise similarity techniques. Saleh et al. [70] presented an optimized heterogeneous stacking ensemble model combining RNN, LSTM, and GRU with LR, RF, and SVM to enhance Arabic sentiment analysis performance. Alharbi et al. [13] proposed a deep learning-based model using recurrent neural networks (RNN) for Arabic sentiment analysis, focusing on leveraging the capabilities of RNNs to capture the sequential nature of text for more accurate sentiment prediction. Saeed et al. [68] utilized optimized compact features and feature reduction techniques to achieve high accuracy in Arabic opinion text classification. Their approach managed to ensure efficiency in terms of both time and computational resources while maintaining classification performance. Elnagar et al. [40] introduced new datasets for single-label and multi-label Arabic text categorization tasks and conducted a comprehensive comparison of various deep learning models. This work provided significant benchmarks and facilitated further research in Arabic text classification. AlGhamdi and Khan [12] developed labelled datasets for Arabic tweets and performed an intelligent analysis using six machine learning algorithms to detect suspicious messages. Their study contributed to the development of a statistical benchmark for future research on crime detection in social media.

B. ENSEMBLE APPROACH

Alshehri et al. [23] studied the classification of Arabic speech acts on Twitter by developing the ASAD dataset of 22,352 annotated tweets. BERT-based models were used, finding the araBERTv2-Twitter model to be most effective with an accuracy of 0.84 and an F1 score of 0.73, further improved to 0.85 and 0.74 through ensemble methods. Their research highlights the effectiveness of Transformer models and data augmentation in understanding Arabic speech acts and suggests future work to improve performance for minority classes and contextual tweet analysis. Hicham et al. [47] developed an ensemble-based sentiment analysis model that outperformed other machine learning classifiers in accuracy, specificity, precision, F1 score, and sensitivity. Al-Hashedi et al. [5] proposed a machine learning model to analyze Arabic tweets, demonstrating significant improvement in F1 score compared to baseline classifiers and other single-based and ensemble-based classifiers without SMOTE. Saleh et al. [70] presented an optimized heterogeneous stacking ensemble model combining RNN, LSTM, and GRU with Logistic Regression, Random Forest, and Support Vector Machine to enhance Arabic sentiment analysis performance. Abo et al. [3] employed a multi-criteria method to empirically assess and rank classifiers for Arabic sentiment analysis, concluding that deep learning and SVM classifiers perform best, surpassing decision trees, K-nearest neighbours, and Naöve Bayes classifiers. Al-Sarem et al. [7] provided an intensive review of previous studies for the Arabic language and utilized the Technique for Order Preferences by Similarity to the Ideal Solution (TOPSIS) method to select the base classifier for AdaBoost and Bagging ensemble methods. The research demonstrates the effectiveness of ensemble techniques in Arabic language authorship attribution.

C. TRANSFORMERS APPROACH

Hossain et al. [48] propose the Transformers Approach. This transformer-based study creates an Arabic COVID-19 text identification system employing two datasets: AraEC (206,797 texts for embedding models) and AraCoV (9099 COVID and 12,195 non-COVID texts for classification). Annotation using a translation API and students obtained a Kappa score of 0.89. Expert verification balanced the 2.8 million-word corpus, and word cloud and sample distribution visualizations improved comprehension of both COVID and non-COVID article material. Shah et al. [71] introduced a modified switch transformer (MST) for detecting sarcasm and categorizing emotion and dialect in Arabic text data, which incorporates probabilistic projections via a Variational Spatial Gated Unit-MLP to improve the embedding generation process. Bani-Almarjeh and Kurdy [30] examined the performance of several model designs, including RNN-based and Transformer-based ones, and different pre-trained language models, including mBERT, AraBERT, and AraT5, for Arabic abstractive summarization. Alammary [10] synthesized the different Arabic BERT models applied to text classification, investigating the differences between them and comparing their performance to the original English BERT models. Saeed et al. [68] suggested a supervised learning strategy for Arabic review sentiment categorization that employs optimized compact features and feature reduction approaches to provide good accuracy and time/space savings. Baniata et al. [31] proposed a Transformer-based neural machine translation model for Arabic vernaculars that uses subword units to improve the behavior of the encoder's multi-head attention sublayers by capturing overall word dependencies in the input sentence. Lan et al. [54] pre-trained a bilingual BERT designed specifically for Arabic NLP and English-to-Arabic zero-shot transfer learning, and investigated GigaBERT's performance on four information extraction tasks: named entity recognition, part-of-speech tagging, argument role labeling, and relation extraction.

D. HYBRID APPROACH

Eman Aljohani [17] investigates Arabic news categorization using deep learning, introducing a hybrid neural network that combines TF-IDF, FastText embeddings, and sequential networks. The study highlights the robust performance of Logistic Regression, SVM, and LSVM, and emphasizes the superiority of FastText embeddings without stemming. Notably, the Hybrid-Bi model performs well, underscoring the necessity of incorporating choice, stemming, and ensemble approaches into Arabic NLP. Lamtougui et al. [53] present a new system for offline Arabic handwritten text recognition based on a combination of Convolutional Neural Network (CNN) and Bidirectional Long-Term Memory (BLSTM), followed by a Connectionist Temporal Classification layer (CTC). Al-Anzi [4] proposes a piecewise Stochastic Gradient Descent (SGD)-based model for sentiment classification of Arabic tweets, utilizing a TF-IDF-based term weighting scheme for textual feature representation. Alsaleh and Larabi-Marie-Sainte [21] describe a hybrid technique for Arabic Text Classification (ATC) that uses the Firefly Algorithm-based Feature Selection (FAFS) to handle the complexity of Arabic language classification. Tested using an SVM classifier, the FAFS approach enhances ATC accuracy and outperforms techniques such as InfoGain, OneR, and TF-IDF. Saeed et al. [68] propose a supervised learning approach for Arabic reviews sentiment classification, utilizing optimized compact features and feature reduction techniques to achieve high accuracy and time/space savings. Finally, Alhawarat and Aseeri [14] developed a Superior Arabic Text Categorization Deep Model (SATCDM) which achieves very high accuracy compared to current research in Arabic text categorization, outperforming similar studies on Arabic document classification tasks using 15 freely available datasets.

The literature study identifies a number of difficulties, such as dialects and informal language, in sentiment analysis of Arabic posts. Even while previous models like LSTM, AraBERT, SGRU, and SBi-GRU have demonstrated encouraging outcomes, there are still issues in handling delicate subjects like sexual harassment on social media. Our suggested method, the hybrid TransNet model, combines a deep attentional hybrid transformer with LSTM and GRU architectures to overcome these issues. Through the use of the transformer to capture long-range dependencies and the capabilities of LSTM and GRU for sequential data processing, TransNet seeks to improve sentiment analysis performance on Arabic posts while skillfully handling language subtleties and uncertainty management.

III. DATA DESCRIPTION AND DATA PREPROCESSING

The dataset used in this study comprises 213,465 posts sourced from X, described in [19] and presented in Table 1. This dataset provides comprehensive coverage of conversations about asthma in the Arabic language, making it an invaluable tool for scholars and professionals looking to learn more about the attitudes, experiences, and views

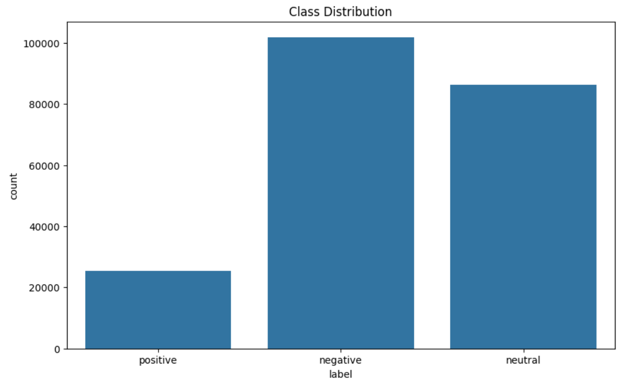


FIGURE 1. Distribution of positive, neutral, and negative sentiments in arabic asthma posts.

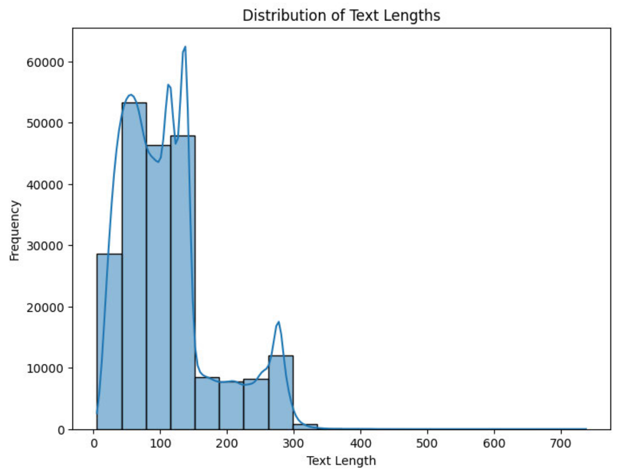


FIGURE 2. Sequence length distribution: arabic asthma posts.

of Arabic-speaking people about asthma. Arabic asthma posts are categorized into three categories depending on the sentiment: good, neutral, and negative, as shown in the bar graph in Figure 1. Of the 213,464 posts examined, 101,858 (or almost 48%) were negative. This is a large overcount of posts compared with positive and neutral posts. On the other hand, good posts make up a far smaller percentage of the dataset—just 25,425 posts, or approximately 12%—than negative posts. The remaining posts, or approximately 40% of all posts, were neutral, making up 86,181 posts. The distribution of emotions highlights the negative attitude prevalent in Arabic posts about asthma. This shows that critical or unfavorable ideas and expressions predominate over good or neutral sentiments.

A. SEQUENCE LENGTH DISTRIBUTION

To learn more about the average length of posts sent by users, the text length distribution for Arabic Asthma posts was examined. The corresponding graphic, named "Distribution of Text Lengths in Arabic Asthma posts," shows in Figure 2 the frequency of posts plotted on the x-axis against different

TABLE 1. Provides an overview of the range of feelings represented in the dataset by displaying example posts from the dataset together with the relevant sentiment labels (positive, negative, and neutral).

Comment	Translated Text	Label
يساعد نبات الكوسة فى شفاء مرض الربو وذلك لاحتوائه على فيتامين وهو مضاد قوي للأكسدة ولديه خواص مضادة للالتهاب بطريقة فعالة	Zucchini helps in curing asthma because it contains a vitamin that is a powerful antioxidant and has effective anti-inflammatory properties.	positive
الربو ملك موت مُصغر، يخليني أحتضر وبعدها يرجعني للحياه	Asthma is a miniature angel of death, it makes me die and then brings me back to life.	negative
صور شاهد كيف تقوم الأسماك بعلاج مرضى الربو في الهند	Pictures: See how fish treat asthma patients in India	neutral

TABLE 2. Top 10 words with their respective average sentiment scores and sentiment categories.

Word	Average Sentiment Score	Sentiment Category
يخاطب	0.0011	Positive
مرض	0.0009	Positive
علاج	0.0008	Positive
بعد	-0.0001	Neutral
الهي	0.0002	Positive
بطل	0.0012	Positive
شفاء	0.0010	Positive
حياة	0.0013	Positive
طبيب	0.0011	Positive
جسم	0.0009	Positive

text length bins, ranging from 0 to 700 characters. Although the labels have a maximum character count of 700, the x-axis is terminated at 600. The y-axis shows the frequency of posts that fall into each bucket. According to the graphic, most posts were fewer than 100 words, and the frequency of posts decreased noticeably as the text length increased. Although it is difficult to ascertain the precise average text length from this graph, it is clear that the majority of posts are brief messages, with a sizable fraction consisting of fewer than 100 characters. This study provides insightful information on the conciseness and shortness of Arabic Asthma posts.

B. ANALYSIS OF MOST FREQUENT WORDS IN ARABIC ASTHMA POSTS

The top 10 terms are shown in Table 2 according to their average sentiment scores and matching sentiment categories obtained from posts with Arabic asthma. According to the research, a number of terms, like بطل (hero), شفاء (healing), and حياة (life), show positive attitudes. The average sentiment ratings for these terms ranged from 0.0008 to 0.0013, which suggests a preponderance of positive attitudes. Nevertheless, certain words, with average sentiment ratings of −0.0001, respectively, show neutral feelings. Examples of these words include بعد (after). This study highlighted the range of emotional tones portrayed in the dataset by offering insights into prevalent attitudes communicated through the most frequently used phrases in Arabic asthma posts.

C. EXPLORING COMMON THEMES IN ARABIC ASTHMA POSTS

The word cloud in Figure 3 provides a visual depiction of the most common themes and issues discussed in asthma-related Arabic posts. This visualization, titled "Arabic Asthma posts Word Cloud," shows the frequencies of particular terms in the dataset to greater recurrence rates. أزمة (crisis), الربو (asthma), التنفس (breathing), الصدر (chest), and الهواء (air) are well-known phrases that are represented in the cloud. These terms illustrate the difficulties related to asthma attacks, breathing problems, and the effect of breathing, reflecting the main topics expressed by X users. The word cloud is an invaluable resource for identifying recurrent themes and comprehending the most common issues raised by people in Arabic-speaking communities concerning asthma.

FIGURE 3. Arabic asthma posts word cloud.

D. TEMPORAL TRENDS IN ARABIC ASTHMA POSTS

The temporal distribution of Arabic posts regarding asthma from 2010 to 2022 is depicted in the time series plot in Figure 4. The x-axis represents the chronology in years, while the y-axis shows the number of posts, which ranges from 0 to more than 1400. Notable tweet frequency spikes were seen across the timeline, indicating times when the debate on the subject was at its highest.

According to our data, the number of posts about asthma fluctuated throughout the years, with specific times seeing a considerable increase in activity. Several factors, such as seasonal variations, public health incidents, and asthma awareness efforts, may have caused these surges. Over time, the baseline level of tweet activity remained constant despite these swings.

Understanding X activity trends can significantly benefit organizations that treat asthma and public health authorities. By spotting trends and connecting them with external events, stakeholders might gain a better understanding

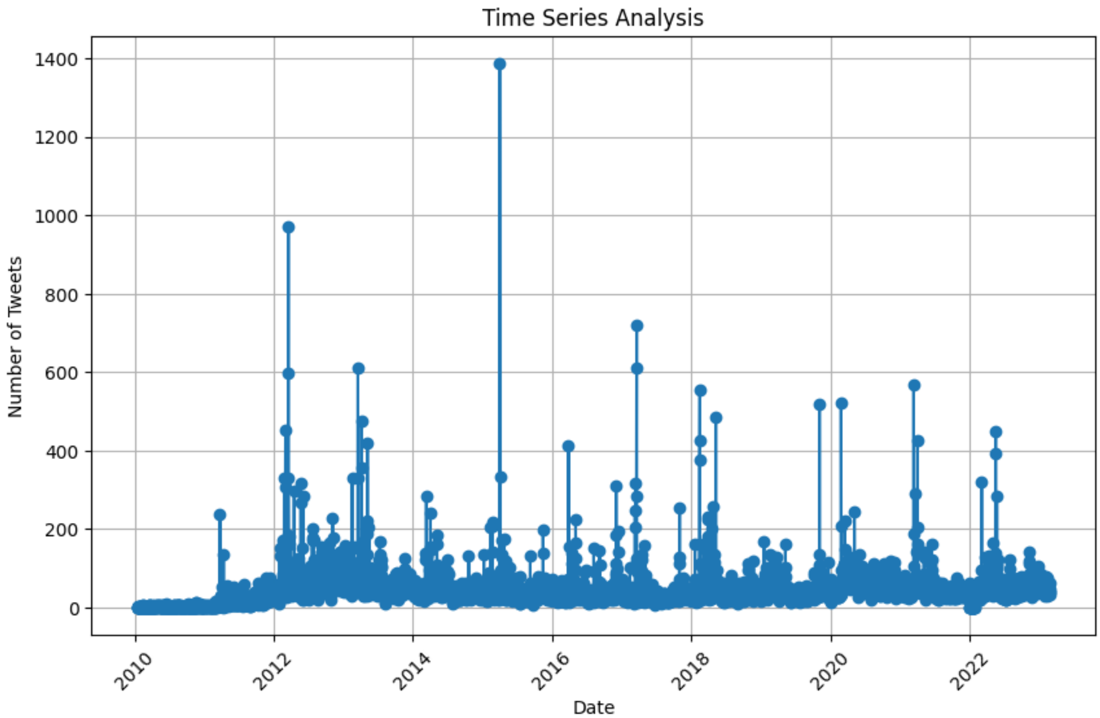


FIGURE 4. **Time Series Analysis of Arabic Asthma posts (2010-2022).**

of the dynamics of asthma discourse in Arabic-speaking communities and adjust treatments or awareness activities.

E. DATA CLEANING

To ensure the quality and integrity of the dataset, the first step in the data preparation pipeline involves careful data cleaning [15], [28], [34]. This method consists of several crucial actions to refine raw text data. To reduce noise and interfere with further research, punctuation marks were carefully deleted from the text corpus. To decrease noise and improve the model's emphasis on critical textual patterns, stop words—standard terms with no semantic value were also carefully removed from the dataset. In addition, to avoid inconsistencies and guarantee a consistent dataset structure, occurrences of null values were found in the dataset and diligently eliminated. The dataset was prepared for the following processing stages by completing these thorough data cleaning methods, which enabled more precise and insightful analyses in the creation of a deep learning model for categorizing Arabic posts linked to asthma.

F. DATA AUGMENTATION

In this work, we apply data augmentation to improve the model's robustness in Arabic asthma-related post-classification. This procedure replaces synonyms and uses WordNet to expand the variety of data in our dataset [36], [66], [76]. Given a sentence in Arabic S with n words in it:

$$S = \{w_1, w_2, \ldots, w_n\}$$

Tokenize it into separate words. We use an Arabic-compatible WordNet to extract the set of synonyms for each word w_i:

$$\text{Synonyms}(w_i) = \{s_{i1}, s_{i2}, \ldots, s_{im}\}$$

where the number of synonyms for w_i is represented by m. We choose a synonym s_{ij} at random from the collection to replace w_i if the synonym set has more than one member (i.e., $m > 1$):

$$w_i \to s_{ij}$$

The term stays the same if there are no synonyms for it or if the collection just includes the word. The chosen words are then changed to their synonyms to create the enhanced phrase S':

$$S' = \{s_{1j_1}, w_2, s_{3j_3}, \ldots, w_n\}$$

where the original words w_1 and w_3 are replaced by synonyms s_{1j_1} and s_{3j_3}, respectively; words without appropriate synonyms stay the same. For the sake of clarity, the following pseudocode may be used to describe this process:

for each $w_i \in S$ do:
  if $\text{Synonyms}(w_i) > 1$ then:
    $s_{ij} \leftarrow \text{random\_choice}(\text{Synonyms}(w_i))$
    $w_i \to s_{ij}$
  else:
    w_i remains unchanged
end for

This results in an improved sentence that adds variation while preserving the original meaning. For instance, "The inhaler helps relieve asthma symptoms" may become يساعد جهاز الاستنشاق في تخفيف أعراض الربو ("The inhaler helps alleviate asthma symptoms") instead of البخاخ يخفف من أعراض الربو("The spray reduces asthma symptoms"). Additionally, "Asthma attacks can be triggered by allergies" may become يمكن أن تسبب الحساسية نوبات الربو ("Allergies can cause asthma attacks") instead of تؤدي الحساسية إلى نوبات الربو("Allergies lead to asthma attacks"). By adding value to our dataset, this technique lessens class imbalance and enhances the generalization of the model.

For every text item in our dataset, we employ this augmentation approach to produce an extended version that includes both the original and supplemented sentences. This method ensures that our deep learning model is trained on a wide range of instances, which will improve its performance and resilience when categorizing asthma-related Arabic social media postings.

G. UPSAMPLING TECHNIQUE: ADDRESSING CLASS IMBALANCE

A popular data augmentation method in machine learning to overcome the concerns of class imbalance is upsampling, which creates an artificial increase in the number of examples in the minority class. Upsampling attenuates the impacts of class imbalance and enhances model performance in the context of sentiment analysis [18] in case of unequal sentiment categories. A random selection process is used to replicate or synthesize instances from the minority class to the size of the majority class. Adding this kind of augmentation to the data exposes the model to a more equal representation of the different sentiment categories to reduce bias towards the majority class. Upsampling is beneficial when coupled with other preprocessing steps and model architectures to create a robust sentiment analysis pipeline to capture the nuances of sentiment expressions in text data effectively.

H. LABEL ENCODING

To facilitate the classification of Arabic asthma-related posts, we convert categorical labels into one-hot encoding using TensorFlow's `to_categorical` function [19]. Each unique label in the dataset is mapped to a numerical index using a dictionary comprehension based on the DataFrame's unique labels:

$\text{label\_dict} = \{\text{label} : \text{idx} \mid \text{idx}, \text{label} \in \text{enumerate(df['label'].unique())}\}$

This step ensures consistency across training, validation, and test datasets. Subsequently, the number of unique classes K is determined to establish the dimensions of the one-hot encoded vectors ($\text{num\_classes} = K$). For each dataset split (training, validation, and test), categorical labels are transformed into one-hot encoded vectors using the `to_categorical` function:

$$\text{train\_labels} = (\textit{train\_labels}.\text{map(label\_dict)})$$
$$\text{val\_labels} = (\textit{val\_labels}.\text{map(label\_dict)})$$
$$\text{test\_labels} = (\textit{test\_labels}.\text{map(label\_dict)})$$

This transformation converts each categorical label y_i into a binary vector $\mathbf{y}_i$ of length K, where K is the number of classes. The vector $\mathbf{y}_i$ is defined as:

$$\mathbf{y}_i[j] = \begin{cases} 1 & \text{if } j = y_i \\ 0 & \text{otherwise} \end{cases}$$

This encoding scheme is pivotal for training deep learning models, enabling efficient multi-class classification of Arabic text data related to asthma across the different dataset splits.

I. TEXT TOKENIZATION

Text Tokenization is necessary for the study as it helps the machine to understand the human language. Through the use of a pre-trained BERT tokenizer created especially for Arabic [69], each text entry is encoded and tokenized into numerical representations appropriate for model contributions [1]. Special tokens like [CLS] and [SEP] are incorporated by the tokenizer to ensure consistent formatting across inputs and to define sentence boundaries [55]. Then, in order to standardize the input size, sequences are either padded or truncated to a maximum length of 240 tokens ($\text{max\_length} = 240$), with [PAD] tokens used for padding and extra tokens discarded for truncation:

$$\text{tokens} = \text{tokenizer.encode\_plus}(T, \text{max\_length} = 240)$$

Additionally, the tokenizer generates an attention mask that distinguishes between non-padded (1) and padded tokens (0), optimizing the model's focus during training and inference:

$$\text{attention\_mask} = \text{tokens['attention\_mask']}$$

Post-tokenization, text data is converted into arrays of input IDs (input_ids) and attention masks (attention_mask) for:

- - Training (train_input_ids, train_attention_mask)
- - Validation (val_input_ids, val_attention_mask)
- - Test (test_input_ids, test_attention_mask)

These arrays are further processed to remove additional dimensions, ensuring compatibility with a deep learning architecture tailored to classify Arabic social media content pertaining to asthma proficiently.

IV. PROPOSED METHODOLOGY

Figure 5 shows the architecture of TransNet for Arabic post classification. This is a hybrid deep learning model that combines Transformer, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) networks. To transform the input text data into tokens, which are then translated into input IDs with an associated attention mask, the model first used a pre-trained tokenizer. These input IDs are converted into vectors by the embedding layer and then

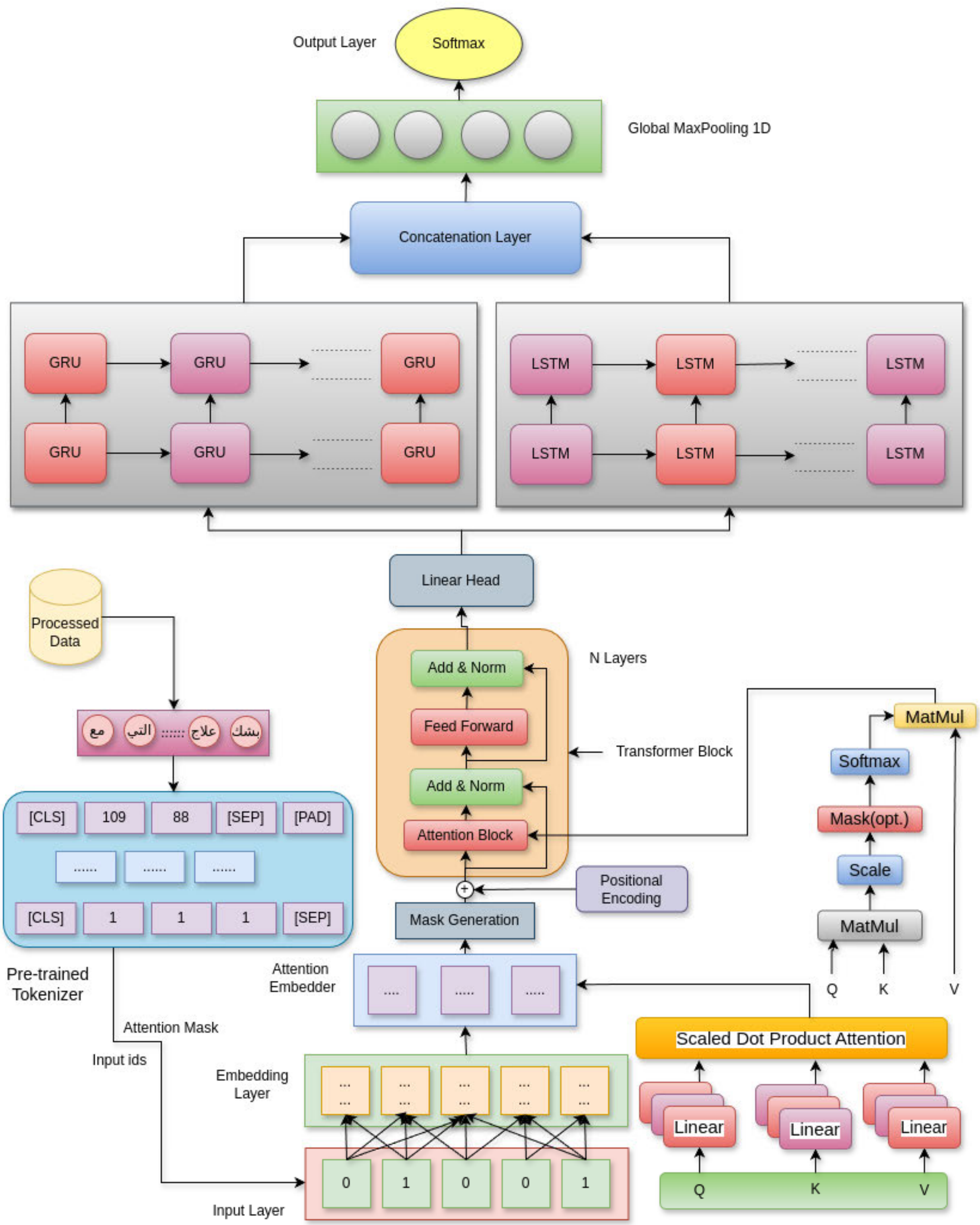


FIGURE 5. **Architecture of deep attentional hybrid transformer for arabic posts classification: TransNet.**

run via an attention embedder. The embedded input is processed by the transformer block, which consists of several layers with elements including add and norm operations, feed-forward networks, and scaled dot-product attention. The embedded vectors are simultaneously supplied to other LSTM and GRU networks. To minimize dimensionality, the outputs from the LSTM, GRU, and Transformer block were concatenated and run through a global max pooling layer. After a linear head, the probability distribution across the target classes was generated by a softmax output layer. This design makes use of the advantages of each distinct network type, including the Transformer's capacity to manage long-range dependencies, LSTM's long-term memory, and GRU's effectiveness in updating the hidden state. By efficiently collecting different features of sequential data, this hybrid model seeks to enhance text classification problems by utilizing the complementary characteristics of the Transformers, LSTMs, and GRUs. This overcomes the shortcomings of any single model by developing a solid system for comprehending the text's context and temporal sequences. The proposed method created a deep-learning model to classify Arabic posts related to asthma. Recurrent neural networks (RNNs) and transformer-based attention mechanisms are used in the model architecture to extract global semantic linkages and local sequential dependencies from text input. First, an attention embedder module was designed to translate the input embeddings into query (q), key (k), and value (v) vectors, enabling better feature representation. The model may then sequentially apply many layers of attention mechanisms to highlight the relevant parts in the input text and generate insightful representations. The model additionally incorporates GRU and LSTM layers. The final design includes a pooling layer and a deep softmax layer for categorization. The model is trained using a preprocessed dataset that has undergone label encoding, upsampling for data balance, and tokenization using a pre-trained BERT tokenizer. The objective of this study is to develop a robust model that can accurately classify asthma-related posts in Arabic, enhancing respiratory health awareness and care for populations that speak Arabic.

A. TransNet MODEL TRAINING ALGORITHM

Algorithm 1 outlines the training procedure for the TransNet model, particularly emphasizing its ability to process and understand Arabic asthma-related textual data through a series of layers, including Transformers, LSTM, and GRU units. Here, we'll explore in-depth the computations involved in each layer, starting with the attention mechanism and subsequent transformations.

- **Input Layer**: The input layer of the TransNet model plays a pivotal role in processing sequential data for classification tasks. Initialized with an input shape of (240,), it defines the length of each sequence processed by the model. This shape accommodates sequences of 240 tokens or features, making it versatile for handling a wide range of text and sequence lengths. Each sequence is encoded into dense vectors through an embedding layer, where tokens are transformed into fixed-size representations. This initial transformation ensures that the model can effectively learn meaningful representations of the input data, which is essential for subsequent layers, such as attention mechanisms and recurrent units, to capture dependencies and patterns within the sequences. The input layer's configuration, including the vocabulary size and embedding dimensions, establishes the foundational framework upon which the TransNet model effectively processes and learns from sequential input data.
- **Embedding Layer**: The embedding layer in the TransNet model converts input tokens into dense vectors of fixed size, facilitating effective representation learning for sequential data processing. Initialized with a vocabulary size vocab_size = tokenizer.vocab_size + 1 and embedding dimension embed_dim = 128, this layer transforms each token into a dense vector $\mathbf{e}_i \in \mathbb{R}^{128}$, where i denotes the index of the token in the vocabulary. The embedding operation can be mathematically represented as:

$$\mathbf{e}_i = \text{Embedding}(i), \quad i = 1, 2, \ldots, \text{vocab\_size}$$

Here, Embedding$(\cdot)$ represents the embedding function that maps token indices to dense vectors. Dropout regularization is applied with a rate dropout_rate = 0.1 to the output of the embedding layer to prevent overfitting:

$$\mathbf{e}'_i = \text{Dropout}(\mathbf{e}_i), \quad \mathbf{e}'_i \in \mathbb{R}^{128}$$

This ensures that during training, a fraction of the elements in $\mathbf{e}_i$ are randomly set to zero, aiding in the generalization of the model. The embedding layer's output $\mathbf{e}'_i$ serves as the initial dense representation of input tokens, providing a foundational step for subsequent layers to process and extract features from the embedded representations.
- **Attention Embedding Process**: The attention embedding process in the TransNet model involves several key steps aimed at transforming input embeddings into query (Q), key (K), and value (V) matrices. These matrices are fundamental for computing scaled dot-product attention, a mechanism that allows the model to attend to different parts of the input sequence selectively.
  1) **Dense Layer Transformation:** Initially, the embedded tokens $\mathbf{e}'_i$ are passed through dense layers to generate the query $\mathbf{q}$, key $\mathbf{k}$, and value $\mathbf{v}$ matrices:

  $$\mathbf{q} = \text{Dense}(\mathbf{e}'_i), \quad \mathbf{k} = \text{Dense}(\mathbf{e}'_i), \quad \mathbf{v} = \text{Dense}(\mathbf{e}'_i)$$

  Here, Dense$(\cdot)$ represents a dense (fully connected) layer that applies a linear transformation to each token embedding $\mathbf{e}'_i$.
  2) **Multi-Head Attention:** Next, the $\mathbf{Q}, \mathbf{K}$, and $\mathbf{V}$ matrices undergo multi-head attention computation. In the TransNet model, multi-head attention is typically implemented with 2 heads (num_heads = 2), allowing the model to attend to information from different representation subspaces jointly.
  For each head h, the scaled dot-product attention mechanism computes attention scores α_{ij}^h between query $\mathbf{q}$ and key $\mathbf{k}$ matrices:

  $$\alpha_{ij}^h = \frac{\mathbf{q}_i \cdot (\mathbf{k}_j)^T}{\sqrt{d_k}}$$

  where $\mathbf{q}_i$ and $\mathbf{k}_j$ are rows of matrices $\mathbf{Q}$ and $\mathbf{K}$ respectively, and d_k is the dimensionality of the key vectors (embed_dim/num_heads in the TransNet model).
  3) **Attention Weights and Output:** The attention scores α_{ij}^h are scaled and softmax across the sequence to obtain attention weights $\mathbf{A}^h$:

  $$\mathbf{A}^h = \text{softmax}(\frac{\mathbf{QK}^T}{\sqrt{d_k}})$$

Algorithm 1 TransNet Model Training Algorithm

```
Require:
Input shape: (240, )
Vocabulary size: vocab_size = tokenizer.vocab_size + 1
Embedding dimension: embed_dim = 128
Number of attention heads: num_heads = 2
Feed-forward dimension: ff_dim = 128
Number of Transformer layers: num_layers = 2
Dropout rate: dropout_rate = 0.1
Number of output classes: num_classes = len(label_dict)
Ensure:
function model(input_shape, vocab_size, embed_dim, num_heads, ff_dim, num_layers, dropout_rate, num_classes)
    Initialize input tensors:
      input_ids ← Input tensor with shape input_shape
      attention_mask ← Input tensor with shape input_shape
    Apply embedding layer:
      embedded ← Embedding layer with input dimension vocab_size and output dimension embed_dim
    Apply dropout to embedding_layer
    Compute attention embeddings:
      q ← Dense layer applied to embedded
      k ← Dense layer applied to embedded
      v ← Dense layer applied to embedded
    Create attention mask:
      mask ← Lower triangular matrix and invert it
    Apply Transformer layers:
    for i ← 1 to num_layers do
        Compute scaled dot-product attention:
          attention_output ← Multi-head attention applied to q, k, v, and mask
          Apply dropout to attention_output
        Apply residual connection and layer normalization:
          out1 ← Layer normalization of embedded + attention_output
        Apply feed-forward network with ReLU activation:
          ffn_output ← Dense layer with ReLU activation applied to out1
          Apply dropout to ffn_output
        Apply residual connection and layer normalization:
          out2 ← Layer normalization of out1 + ffn_output
        Update embedded to out2
    end for
    Apply LSTM and GRU layers:
      lstm_output ← LSTM layer applied to embedded
      gru_output ← GRU layer applied to embedded
      concatenated ← Concatenate lstm_output and gru_output
    Apply pooling and output layers:
      pooled_output ← Global max pooling applied to concatenated
      outputs ← Dense layer with softmax activation applied to pooled_output
    Return the model:
      model ← Model with inputs [input_ids, attention_mask] and outputs outputs
      Return model
end function
```

where $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n]$ and $\mathbf{K} = [\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_n]$ are matrices formed by stacking query and key vectors.

4) **Weighted Sum:** Finally, the weighted sum of value vectors $\mathbf{v}$ using attention weights $\mathbf{A}^h$ computes the output of multi-head attention $\mathbf{O}^h$:

$$\mathbf{O}^h = \mathbf{A}^h \mathbf{V}$$

where $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n]$ is the matrix of value vectors.

5) **Multi-Head Concatenation:** In practice, the outputs $\mathbf{O}^1$ and $\mathbf{O}^2$ from each head are concatenated and linearly transformed to form the final output of the attention mechanism, providing a comprehensive representation of the attended input tokens.

The attention embedding process in the TransNet model enables the effective capture of token relationships and dependencies within input sequences. By transforming embeddings into query, key, and value matrices and computing multi-head attention, the model can focus on relevant information for subsequent processing layers. This mechanism is crucial for enhancing the model's ability to understand and utilize sequential data in tasks such as text classification and natural language understanding.

- **Transformer Block**: Applying Transformer layers in the TransNet model involves a series of operations designed to process and extract hierarchical representations from input sequences. Each Transformer layer consists of multi-head attention and feed-forward networks, augmented by residual connections and layer normalization, which collectively enable effective learning of contextual dependencies across tokens.

1) **Multi-Head Attention:** Each Transformer layer begins by computing multi-head attention:
   - **Query (Q), Key (K), and Value (V) Computation:** Similar to the attention embedding process, embeddings are transformed into Q, K, and V matrices using dense layers:

$$\mathbf{Q} = \text{Dense}(\mathbf{X}), \quad \mathbf{K} = \text{Dense}(\mathbf{X}),$$
$$\mathbf{V} = \text{Dense}(\mathbf{X})$$

   where $\mathbf{X}$ represents input embeddings or intermediate outputs.
   - **Scaled Dot-Product Attention:** Attention scores are computed between Q and K matrices and then softmaxed to obtain attention weights:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}$$

   where d_k is the dimensionality of the key vectors.
   - **Multi-Head Mechanism:** Multiple sets of Q, K, and V matrices (heads) are processed in parallel to capture different aspects of token relationships.

2) **Residual Connection and Layer Normalization:** After attention computation, residual connections are applied:

$$\text{Residual}(\mathbf{X}) = \mathbf{X} + \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$$
$$\text{LayerNorm}(\text{Residual}(\mathbf{X}))$$

3) **Feed-Forward Networks:** Following normalization, each Transformer layer employs feed-forward networks (FFNs):

$$\text{FFN}(\mathbf{X}) = \text{ReLU}(\text{Linear}(\mathbf{X}))$$

Dropout regularization is applied to the output of the FFN:

$$\text{Dropout}(\text{FFN}(\mathbf{X}))$$

4) **Residual Connection and Layer Normalization (Again):** Similar to the attention mechanism, the output of the feed-forward network is added back via a residual connection and normalized:

$$\text{LayerNorm}(\text{Residual}(\text{FFN}(\mathbf{X})))$$

5) **Iteration:** These steps are iteratively applied for each Transformer layer (typically num_layers = 2 in TransNet):
   - Input embeddings are passed through multiple Transformer layers sequentially.
   - Each layer refines the representation of the sequence, leveraging multi-head attention and feed-forward networks to capture both local and global dependencies effectively.

By applying Transformer layers in the TransNet model, the architecture harnesses the power of self-attention mechanisms to capture intricate relationships between tokens in input sequences. Through iterative processing and hierarchical representation learning, Transformer layers enable the model to learn and utilize contextual information efficiently, making them well-suited for tasks requiring understanding of sequential data, such as natural language processing and sequence classification.

- **LSTM (Long Short-Term Memory) Layer:** LSTM units are equipped with a memory cell $\mathbf{C}_t$ that can maintain information over long periods, addressing the challenge of capturing long-term dependencies in sequences. The LSTM's operation involves several key components:
   - **Forget Gate:** Determines which information from the previous cell state $\mathbf{C}_{t-1}$ should be discarded or retained. It takes as input the concatenation of the current input $\mathbf{x}_t$ and the previous hidden state $\mathbf{h}_{t-1}$, passes through a sigmoid function, and outputs a forget vector $\mathbf{f}_t$:

$$\mathbf{f}_t = \sigma(\mathbf{W}_f[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f)$$

   - **Input Gate:** Decides which new information to store in the cell state $\mathbf{C}_t$. It computes a new candidate cell state $\tilde{\mathbf{C}}_t$ using $\mathbf{x}_t$ and $\mathbf{h}_{t-1}$, scaled by an input gate $\mathbf{i}_t$ obtained from a sigmoid function:

$$\mathbf{i}_t = \sigma(\mathbf{W}_i[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i)$$
$$\tilde{\mathbf{C}}_t = \tanh(\mathbf{W}_C[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_C)$$

   - **Update to Cell State:** The cell state $\mathbf{C}_t$ is updated by combining the previous cell state after forgetting some information and adding the new candidate values scaled by the input gate:

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{C}}_t$$

  - **Output Gate:** Controls which information from the cell state is used to compute the output hidden state $\mathbf{h}_t$:

$$\mathbf{o}_t = \sigma(\mathbf{W}_o[\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o)$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{C}_t)$$

LSTMs are effective in scenarios requiring modeling of long-range dependencies, such as language modeling and machine translation.

- **GRU (Gated Recurrent Unit) Layer**: GRU units are a simplified version of LSTMs, designed to streamline computation while retaining effectiveness in capturing temporal dependencies:
  - **Update Gate:** Combines the functionalities of the input and forget gates in LSTMs into a single update gate $\mathbf{z}_t$:

$$\mathbf{z}_t = \sigma(\mathbf{W}_z[\mathbf{h}_{t-1}, \mathbf{x}_t])$$
$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \tilde{\mathbf{h}}_t$$

  - **Candidate Hidden State:** Computed using $\mathbf{x}_t$ and $\mathbf{h}_{t-1}$, adjusted by a reset gate $\mathbf{r}_t$:

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_h[\mathbf{r}_t \odot \mathbf{h}_{t-1}, \mathbf{x}_t])$$

GRUs are computationally more efficient than LSTMs and are suitable for applications where reducing model complexity and training time is important.

Integration in TransNet Model: In the TransNet model, LSTM and GRU layers are integrated to process sequential data representations obtained from earlier layers, such as embeddings and Transformer blocks. They capture both short-term dependencies (GRU) and long-term dependencies (LSTM) within the input sequences. By sequentially processing tokens and learning representations through recurrent connections, LSTM and GRU layers contribute to the model's ability to understand and leverage context from sequential data, enhancing performance in tasks such as sentiment analysis, text classification, and sequence prediction.

LSTM and GRU layers play crucial roles in the TransNet model by enabling the effective processing of sequential data, capturing temporal dependencies, and learning representations essential for complex tasks in natural language processing and other sequential data domains. Their integration provides the model with the capability to handle varying lengths of input sequences and extract meaningful features that contribute to accurate predictions and classifications.

- In the TransNet model, after processing through LSTM and GRU layers, the outputs from these recurrent units are concatenated to capture complementary representations of the input sequence. The LSTM layer provides a robust mechanism for capturing long-term dependencies, while the GRU layer offers computational efficiency and effective short-term dependency modeling.

Concatenating the outputs of these layers allows the model to leverage both types of representations simultaneously. This concatenated output integrates the strengths of both LSTM and GRU in understanding the sequence dynamics, providing a comprehensive representation that preserves and combines diverse aspects of sequential data.

By concatenating LSTM and GRU outputs, the TransNet model enhances its capacity to learn nuanced patterns and dependencies within sequences, contributing to improved performance in tasks such as sentiment analysis, sequence classification, or any application requiring a thorough understanding of sequential data structures.

- **Global Average Pooling Layer**: After processing through LSTM, GRU, or other recurrent layers in the TransNet model, the output sequence is typically high-dimensional and contains rich contextual information. Global average pooling (GAP) is employed as a method to condense this spatial information into a single feature vector while retaining the most salient features from the entire sequence.

Functionality: Global average pooling computes the average value of each feature map across all spatial locations. Specifically, for a tensor $\mathbf{X}$ with dimensions $N \times H \times W \times C$, where N is the batch size, H and W are the spatial dimensions, and C is the number of channels, the global average pooling operation is defined as:

$$\text{GAP}(\mathbf{X})_{i,j,k} = \frac{1}{H \times W} \sum_{m=1}^{H} \sum_{n=1}^{W} \mathbf{X}_{i,m,n,k}$$

This operation computes the average value $\text{GAP}(\mathbf{X})_{i,j,k}$ for each channel k across all spatial locations (j, k) in the tensor $\mathbf{X}$, resulting in a $N \times 1 \times 1 \times C$ tensor.

Advantages:
  - **Dimensionality Reduction:** Global average pooling reduces the spatial dimensions of the output tensor to 1×1, effectively compressing the information in each feature map.
  - **Translation Invariance:** Unlike fully connected layers, which require fixed-size inputs, global average pooling provides a translation-invariant representation. This property is particularly useful in tasks where the position of features in the input sequence should not affect the model's prediction.
  - **Regularization:** Global average pooling acts as a form of regularization by reducing overfitting, as it aggregates information from the entire feature map rather than focusing on specific spatial locations.

Integration in TransNet Model: After LSTM and GRU layers, the TransNet model employs global average pooling to aggregate information across the entire sequence of features extracted by recurrent layers. The resulting global average pooled tensor serves as a compact representation of the input sequence, which is then fed into subsequent dense layers for classification or regression tasks.

Global average pooling in the TransNet model plays a critical role in summarizing the high-dimensional output from recurrent layers into a condensed representation suitable for final prediction tasks. By averaging spatial information across all positions, it ensures that the model focuses on the most informative aspects of the input sequence while maintaining efficiency and generalization capability. This makes it a valuable component in deep learning architectures for handling sequential data effectively.

- **Output Layer with Softmax Activation:** In the TransNet model, the output layer plays a crucial role in transforming the learned features from preceding layers into probabilistic predictions for multi-class classification tasks. This layer typically includes a dense (fully connected) layer followed by a softmax activation function.

Functionality 1. **Dense Layer**: The dense layer computes a linear transformation of the input features, mapping them to logits for each class. Let $\mathbf{z} \in \mathbb{R}^K$ denote the vector of logits, where K is the number of classes. The logits z_j for class j are computed as:

$$\mathbf{z} = \mathbf{W} \cdot \mathbf{x} + \mathbf{b}$$

where $\mathbf{x}$ is the input vector from the preceding layer, $\mathbf{W}$ is the weight matrix, and $\mathbf{b}$ is the bias vector.

2. **Softmax Activation**: Following the dense layer, the softmax activation function is applied to convert the logits $\mathbf{z}$ into class probabilities $\hat{\mathbf{y}}$:

$$\hat{\mathbf{y}} = \text{softmax}(\mathbf{z})$$

$$\hat{y}_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

where $\hat{y}_j$ represents the predicted probability of class j.

Interpretation: The softmax function normalizes the logits $\mathbf{z}$ into a probability distribution over K classes, ensuring that the sum of probabilities across all classes equals one. This transformation allows the TransNet model to output confident predictions by assigning high probabilities to the most likely classes based on the learned features.

Integration in TransNet Model: After processing through preceding layers such as LSTM, GRU, and global average pooling, the TransNet model's output layer utilizes softmax to provide a probabilistic interpretation of the input sequence. This final layer is crucial for tasks such as sentiment analysis or text classification, where predicting the correct class label with high confidence is essential.

Training and Loss Function: During training, the output layer is optimized using categorical cross-entropy loss, which measures the difference between predicted probabilities $\hat{\mathbf{y}}$ and the actual labels $\mathbf{y}$:

$$\text{Loss} = -\sum_{i=1}^{N} \sum_{j=1}^{K} y_{ij} \log(\hat{y}_{ij})$$

where N is the number of samples, K is the number of classes, y_{ij} is the ground truth label (one-hot encoded), and $\hat{y}_{ij}$ is the predicted probability for class j.

In summary, the output layer with softmax activation in the TransNet model converts learned features into class probabilities, enabling effective multi-class classification. By leveraging the softmax function, the model outputs interpretable predictions that facilitate decision-making in various applications of natural language processing and sequential data analysis.

B. BASE LINE ALGORITHMS

Four basic architectures—Transformer-GRU-CNN, Transformer-GRU, Transformer-LSTM, and Transformer-LSTM-GRU—are examined in the section on the baseline algorithms. For text categorization problems, these designs show different combinations of transformer layers, recurrent neural networks (RNNs), and attention algorithms. Each model presents a unique method for utilizing attention mechanisms in conjunction with recurrent or convolutional layers to capture semantic linkages and sequential dependencies within input text data. We want to learn more about the efficacy of various architectural decisions in handling text classification problems by analyzing the performance of these baseline algorithms. This will provide us with valuable benchmarks for contrasting and evaluating the proposed hybrid Transformer-LSTM-GRU model [59], [72], [75].

- **Transformer-GRU-CNN:** We provide a unique architecture for text classification problems in the Transformer-GRU-CNN section, which combines the transformer mechanism with layers of Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU). The goal of this design is to efficiently collect various characteristics of incoming text data using utilizing the advantages of each component. The Transformer technique allows the model to recognize long-range relationships within a text sequence and pay attention to global contextual information. The recurrent nature of GRU layers makes it easier to describe temporal dynamics and sequential relationships in the data. Furthermore, CNN layers are utilized for convolutional operations to extract local patterns and features from text input. Through the integration of these elements, the transformer-GRU-CNN model provides a complete method for text categorization that can concurrently capture global and local data. Our objective is to evaluate the performance of this architecture and compare it with other baseline algorithms to ascertain how well it performs in handling text categorization jobs through thorough testing and assessment.

- **Transformer-GRU:** In the Transformer-GRU section, we suggest a fusion architecture for text classification problems that combines Gated Recurrent Units (GRU) with a transformer mechanism. The goal of this design is to use the strengths of the two models to capture various facets of incoming text data efficiently. Through self-attention methods, the transformer mechanism allows the model to attend to global contextual information and capture long-range relationships within the text sequence. Conversely, the recurrent nature of GRU layers makes them excellent for capturing temporal dynamics and sequential relationships in the data. Through the integration of these elements, the Transformer-GRU model provides a comprehensive method for text categorization that can concurrently capture global and local data. We sought to compare this architecture's performance with other baseline algorithms to study its efficacy in handling text categorization tasks through comprehensive testing and assessment.
- **Transformer-LSTM:** For text classification problems, we suggest a hybrid architecture in the Transformer-LSTM section that combines the transformer mechanism with Long Short-Term Memory (LSTM) layers. The goal of this design is to efficiently capture the various characteristics of incoming text data by utilizing the advantages of both models. Through self-attention processes, the transformer mechanism makes it easier to capture long-range relationships and global contextual information within the text sequence. Conversely, the recurrent nature of LSTM layers makes them excellent for capturing temporal dynamics and sequential relationships in the data. Through the integration of these elements, the Transformer-LSTM model provides a complete method for text categorization that can simultaneously capture global and local data. Our objective is to evaluate the performance of this architecture and compare it to other baseline algorithms to ascertain how well it performs in handling text categorization jobs through thorough testing and assessment.
- **Transformer-CNN:** In the Transformer-CNN section, we provide a unique architecture for text classification problems that combine the transformer mechanism with Convolutional Neural Networks (CNN). The goal of this fusion is to capture the various characteristics of the input text data efficiently by utilizing the advantages of both models. Through self-attention processes, the Transformer mechanism makes it easier to capture long-range relationships and global contextual information within the text sequence. Through convolutional processes, CNN layers are skilled at identifying the specific patterns and characteristics present in the text input. Through the integration of these elements, the Transformer-CNN model provides a thorough method for text categorization that can concurrently capture global and local data. We want to evaluate the performance of this architecture by conducting extensive experiments and comparisons with other baseline methods to determine how well it handles text categorization problems.

C. LIME EXPLANATIONS

The LIME (Local Interpretable Model-agnostic Explanations) framework is utilized in the TransNet model to provide interpretability for individual predictions. LIME generates local explanations by perturbing the input text and observing the changes in the model's predictions. This process highlights the most influential words or phrases that contribute to the prediction, thus offering insight into the model's decision-making process [44], [67].

The LimeTextExplainer is instantiated with the class names derived from the dataset's labels. The explain_instance function is defined to generate explanations for a given text. It uses the model_predict function, which tokenizes the text using a BERT tokenizer, processes it into the appropriate format for the model, and obtains predictions. The function then calls explainer.explain_instance to produce an explanation, displaying the results in an HTML format for easy visualization [67].

Mathematically, LIME approximates the complex model f with an interpretable model g around a given instance x. The interpretable model is trained on a perturbed dataset Z generated by making slight alterations to x. The LIME explanation is found by minimizing the following objective function:

$$\xi(x) = \arg\min_{g \in G} \; \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

where:

- $\mathcal{L}(f, g, \pi_x)$ measures how well g approximates f in the locality defined by π_x.
- π_x is a proximity measure that assigns weights to perturbed samples based on their similarity to x.
- $\Omega(g)$ is a complexity measure to ensure that g remains interpretable [44], [67].

In this case, the BERT tokenizer processes the input text into token IDs and attention masks, which are then used to predict probabilities across different classes. The explanations highlight the contribution of each token to the prediction by showing how perturbing each token affects the output [67].

This approach helps in understanding how the model interprets various sentiments expressed in texts about asthma, ranging from negative and positive to neutral sentiments. By explaining instances such as severe asthma attacks, the effectiveness of medication, and general statements about asthma management, LIME provides valuable insights into the inner workings of the TransNet model, facilitating transparency and trust in its predictions [44], [67].

D. HYPARAMETERS

The suggested TranNet, Transformer-LSTM, Transformer-GRU, Transformer-CNN, and Transformer-GRU-CNN are among the models whose hyperparameter values are shown in Table 3. Max_len, num_heads, embed_dim, ff_dim, num_layers, num_classes, Dropout Rate, and particular settings for CNN, LSTM, and GRU units are important hyperparameters. Every model makes use of the Adam optimizer with a 0.0001 learning rate and a loss function based on category cross-entropy. For the majority of models, batch sizes are 32; however, TranNet uses a batch size of 16. All models have early stopping patience set at 3 epochs, with the exception of TranNet, which uses 4 epochs. A total of twenty epochs are used to train each model.

E. MODEL TRAINING AND OPTIMIZATION

We used a methodical approach to optimize the model and optimize the parameters to improve the performance of the proposed architecture. The input data were fed into the model during the training phase, and the weights of the model were iteratively adjusted depending on the calculated loss between the predicted and actual labels. Because of its adaptive learning rate method, which allows for effective optimization by varying the learning rates for each parameter separately, the Adam optimizer is a well-suited option. We also focus on the model's performance metrics on a validation dataset, such as loss and accuracy, to ensure that it does not overfit and can be applied to new data. We used dropout layers, which randomly deactivate a portion of the neurons during training to add regularization to improve generalization and reduce overfitting. By carefully adjusting hyperparameters such as batch sizes, learning rates, and dropout rates, we aim to maximize the performance of the model without sacrificing computational effectiveness. Lastly, to avoid unnecessary computation and possible overfitting, we use early stopping approaches to end training when the model's performance on the validation set stops improving. Following these guidelines, we ensure that our model performs as well as possible while also being robust and scalable for a variety of text categorization jobs.

V. EXPERIMENT AND RESULT ANALYSIS

We performed a thorough analysis of our suggested models in the Experiment and Result Analysis section, utilizing TensorFlow 2.15.0 and Python 3.10.13 on the Kaggle GPU T4 environment. We trained and assessed our models on large-scale text classification tasks by taking advantage of the GPU's processing capability, which enables us to handle and analyze massive volumes of textual data quickly and effectively. To evaluate the effectiveness of each model variant—Transformer-LSTM, Transformer-GRU, and Transformer-CNN architecture, we created a set of experiments that included a range of datasets and classification tasks. We closely tracked essential performance indicators, such as accuracy, precision, recall, and F1-score during the tests to learn more about the predictive power and generalizability of the models to new data.

A. PERFORMANCE ANALYSIS PARAMETERS

In the Performance Analysis Parameters section, we employed a wide range of metrics to measure the predictive power and generalizability of our suggested models to determine their efficacy. We quantified the percentage of correctly identified occurrences relative to the total number of instances by measuring accuracy. Recall gauges the capacity of the model to identify positive cases from all the actual positive instances accurately. In contrast, precision assesses the capacity of the model to categorize positive instances among all cases predicted as accurately positive. The harmonic mean of the accuracy and recall, or F1-score, offers a fair evaluation of the model's performance. To further depict the trade-off between the valid positive rate and false positive rate across various categorization thresholds, we evaluate the Receiver Operating Characteristic (ROC) curve. In addition, we examine the confusion matrix, which shows the numbers of true positive, true negative, false positive, and false pessimistic predictions, to learn more about the model's performance. Our goal was to thoroughly assess the predicted performance and robustness of our models on a range of text categorization tasks by utilizing these performance analysis metrics.

Essential measures, including accuracy (ACC), precision (P), recall (R), and F1-score ($F1$), are used to assess the performance of our models. Here, these metrics are computed:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1-score} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

where:

- TP is the number of true positives,
- TN is the number of true negatives,
- FP is the number of false positives, and
- FN is the number of false negatives.

These equations enabled us to quantitatively assess our models' predictive capabilities, providing valuable insights into their performance across various text classification tasks.

B. EXPERIMENT WITH MAXIMUM LENGTH

Table 4 represents Performance metrics for a range of models, examined with maximum sequence lengths of 120 and 240. These models include Trans-LSTM, Trans-GRU, Trans-CNN, Trans-GRU-CNN, and the proposed TranNet. Train accuracy, validation accuracy, test accuracy, recall (R), precision (P), and F1 score are among the metrics. Outperforming all previous configurations, the suggested TranNet model with a maximum length of 240 achieves the

TABLE 3. Hyperparameter configurations for the Transformer-LSTM, Transformer-GRU, Transformer-CNN, Transformer-GRU-CNN, and the suggested TranNet, among other models.

Hyperparameters	Transformer-LSTM	Transformer-GRU	Transformer-CNN	Transformer-GRU-CNN	Proposed TranNet
max_len	240	120	240	120	240
num_heads	8	8	6	6	2
embed_dim	128	128	128	128	128
ff_dim	128	128	128	128	128
num_layers	8	8	6	6	2
num_classes	3	3	3	3	3
Dropout Rate	0.2	0.2	0.1	0.1	0.1
CNN (Filters)	-	-	64	64	-
CNN (Kernel Size)	-	-	3	3	-
LSTM Units	128	-	-	-	256
GRU Units	-	128	-	64	256
Optimizer	adam	adam	adam	adam	adam
Batch Sizes	32	32	32	32	16
Learning Rate	0.0001	0.0001	0.0001	0.0001	0.0001
Loss Function	categorical_crossentropy				
Early Stopping Patience	3	3	3	3	4
Epochs	20	20	20	20	20

TABLE 4. Performance metrics for various models with different maximum lengths. The metrics include train accuracy, validation accuracy, test accuracy, precision (P), recall (R), and F1 score.

Models	Maximum Length	Train Accuracy	Val Accuracy	Test Accuracy	P	R	F1
Trans-LSTM	120	94.18%	91.90%	92.99%	93.06%	92.99%	92.99%
	240	94.41%	93.58%	93.50%	93.47%	93.49%	93.47%
Trans-GRU	120	87.94%	88.18%	87.97%	87.90%	87.97%	87.92%
	240	87.22%	87.66%	87.76%	87.73%	87.76%	87.74%
Trans-CNN	120	86.91%	86.97%	87.25%	87.19%	87.25%	87.20%
	240	87.01%	87.85%	87.82%	87.82%	87.82%	87.75%
Trans-GRU-CNN	120	89.39%	89.16%	89.16%	89.09%	89.16%	89.10%
	240	89.62%	89.05%	88.92%	89.13%	88.93%	88.78%
Proposed TranNet	120	98.22%	97.48%	97.54%	97.54%	97.54%	97.54%
	240	**98.63%**	**98.05%**	**97.87%**	**97.86%**	**97.86%**	**97.86%**

greatest values in test accuracy (97.87%), precision (97.86%), recall (97.86%), train correctness (98.63%), and F1 score (97.86%). These findings demonstrate TranNet's efficacy at extended sequence lengths.

C. PERFORMANCE COMPARISON WITH BASE LINE MODELS AND PROPOSED TRANSNET

The performance of five distinct models—Trans-LSTM, Trans-GRU, Trans-CNN, Trans-GRU-CNN, and the suggested TransNet—with and without data augmentation is thoroughly compared in Table 5. Six measures are used to assess each model's performance: F1-Score (F1), Precision (P), Recall (R), Test Accuracy, Validation Accuracy, and Training Accuracy. In general, adding data augmentation improves each model's performance on all criteria. Impressively, the suggested TransNet model performs best when data augmentation is used, achieving remarkable results in train accuracy of 98.63%, validation accuracy of 98.05%, test accuracy of 97.87%, and precision, recall, and F1-score of 97.86%. This illustrates how data augmentation significantly improves model performance, especially for the suggested TransNet, which tops all assessed metrics. Table 6 provides a thorough analysis of the performance of five distinct models—Trans-LSTM, Trans-GRU, Trans-CNN, Trans-GRU-CNN, and the suggested TransNet—that were assessed in three sentiment categories: Positive, Negative, and Neutral, both with and without data augmentation. Precision (P), Recall (R), and F1-Score (F1) are used to gauge each model's performance. According to the findings, each model performs better overall across all sentiment categories and metrics when data augmentation is used. The suggested TransNet model performs best with nearly perfect scores for the Positive sentiment (0.99 in Precision, 1.00 in Recall, and 0.99 in F1-Score) and consistently high scores of 0.97 for all metrics in both the Negative and Neutral sentiments.

D. PROPOSED MODEL PERFORMANCE

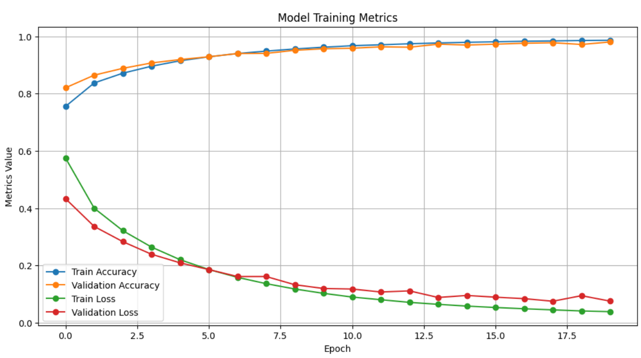
The performance metrics of TransNet are shown in Figure 6, which emphasizes the significance of data augmentation during the training phase. The graphs demonstrate the model's learning efficiency and generalization capacity in both training and validation accuracy and loss across a number of epochs. The metrics with data augmentation during 20 epochs are displayed in Figure 6a, where TransNet exhibits a consistent improvement in training and validation accuracy before stabilizing at the conclusion. Simultaneously, there is a constant decline in training and validation losses, suggesting improved model performance and less overfitting. On the other hand, although the improvements are not as noticeable as in the supplemented scenario, Figure 6b, which

TABLE 5. The performance comparison table among the models: Trans-LSTM, Trans-GRU, Trans-CNN, Trans-GRU-CNN, and the proposed TransNet. Each model is evaluated with (Augmentation=1) and without (Augmentation=0) data augmentation. metrics include training accuracy, validation accuracy, test accuracy, precision (P), Recall (R), and F1-Score (F1).

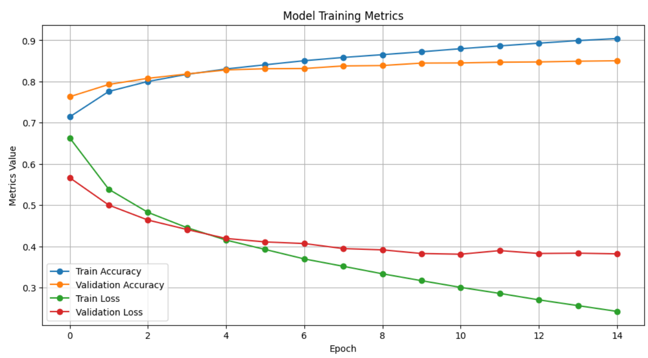
Models	Augmentation	Train Accuracy	Val Accuracy	Test Accuracy	P	R	F1
Transformer-LSTM	0	92.42%	88.78%	88.82%	88.71%	88.78%	88.74%
	1	94.41%	93.58%	93.50%	93.47%	93.49%	93.47%
Transformer-GRU	0	88.15%	86.64%	86.40%	86.29%	86.35%	86.31%
	1	87.22%	87.66%	87.76%	87.73%	87.76%	87.74%
Transformer-CNN	0	86.05%	85.49%	85.35%	85.21%	85.29%	85.21%
	1	87.01%	87.85%	87.82%	87.82%	87.82%	87.75%
Transformer-GRU-CNN	0	87.55%	85.48%	85.74%	85.91%	85.72%	85.77%
	1	89.62%	89.05%	88.92%	89.13%	88.92%	88.78%
Proposed TransNet	0	90.42%	85.03%	84.89%	83.35%	81.60%	82.40%
	1	98.63%	98.05%	97.87%	97.86%	97.86%	97.86%

TABLE 6. Performance Comparison of Different Models with and without Data Augmentation. The table compares the performance of five different models: Trans-LSTM, Trans-GRU, Trans-CNN, Trans-GRU-CNN, and the proposed TransNet. Each model is evaluated with (Augmentation=1) and without (Augmentation=0) data augmentation across three sentiment categories: Positive, Negative, and Neutral. Performance metrics shown include Precision (P), Recall (R), and F1-Score (F1).

Models	Augmentation	Positive			Negative			Neutral		
		P	R	F1	P	R	F1	P	R	F1
Transformer-LSTM	0	0.92	0.96	0.94	0.85	0.88	0.86	0.89	0.83	0.86
	1	0.96	0.98	0.97	0.93	0.91	0.92	0.92	0.92	0.92
Transformer-GRU	0	0.90	0.94	0.92	0.83	0.83	0.83	0.86	0.81	0.84
	1	0.92	0.94	0.93	0.85	0.86	0.85	0.85	0.87	0.85
Transformer-CNN	0	0.86	0.94	0.90	0.84	0.81	0.83	0.85	0.81	0.83
	1	0.90	0.95	0.93	0.89	0.81	0.85	0.85	0.87	0.86
Transformer-GRU-CNN	0	0.93	0.88	0.90	0.81	0.86	0.83	0.84	0.83	0.84
	1	0.90	0.98	0.94	0.93	0.79	0.85	0.85	0.89	0.87
Proposed TransNet	0	0.79	0.72	0.75	0.85	0.89	0.87	0.86	0.84	0.85
	1	0.99	1.00	0.99	0.97	0.97	0.97	0.97	0.97	0.97



(a) Model training metrics with augmentation.



(b) Model training metrics without augmentation.

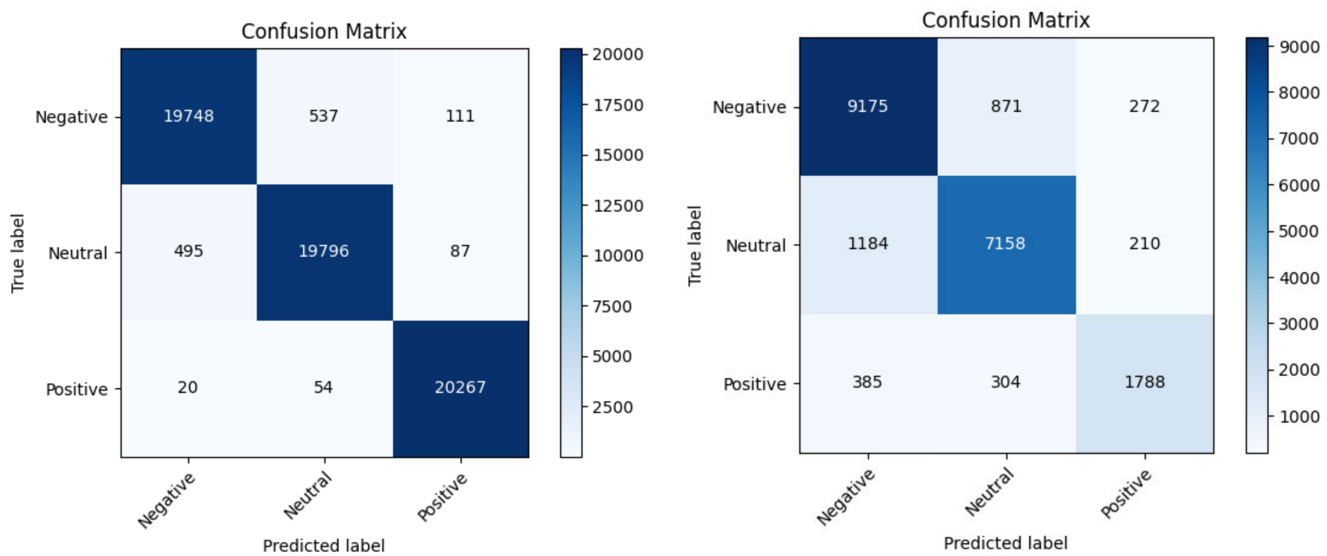
FIGURE 6. The graph of training and validation accuracy and loss throughout 10 epochs.

displays the metrics over a 15-epoch period without data augmentation, nevertheless demonstrates an increased trend in training and validation accuracy and a decreased trend in losses. TransNet's resilience and capacity for achieving high accuracy and low loss through efficient training are demonstrated by these visualizations, and performance is further enhanced by data augmentation.

Confusion matrices are shown in Figure 7 to show how well the suggested model TransNet performs in terms of classification with and without data augmentation. The matrices provide a graphic depiction of the actual vs anticipated labels in a number of areas, emphasizing the accuracy of the model and the type of errors it makes.

With few misclassifications, TransNet accurately categorized most of the negative, neutral, and positive labels shown in Figure 7a. In particular, 537 neutrals were anticipated as negative, and 111 positives were forecasted as negative. It properly predicted 19,748 negative, 19,796 neutral, and 20,267 positive cases.

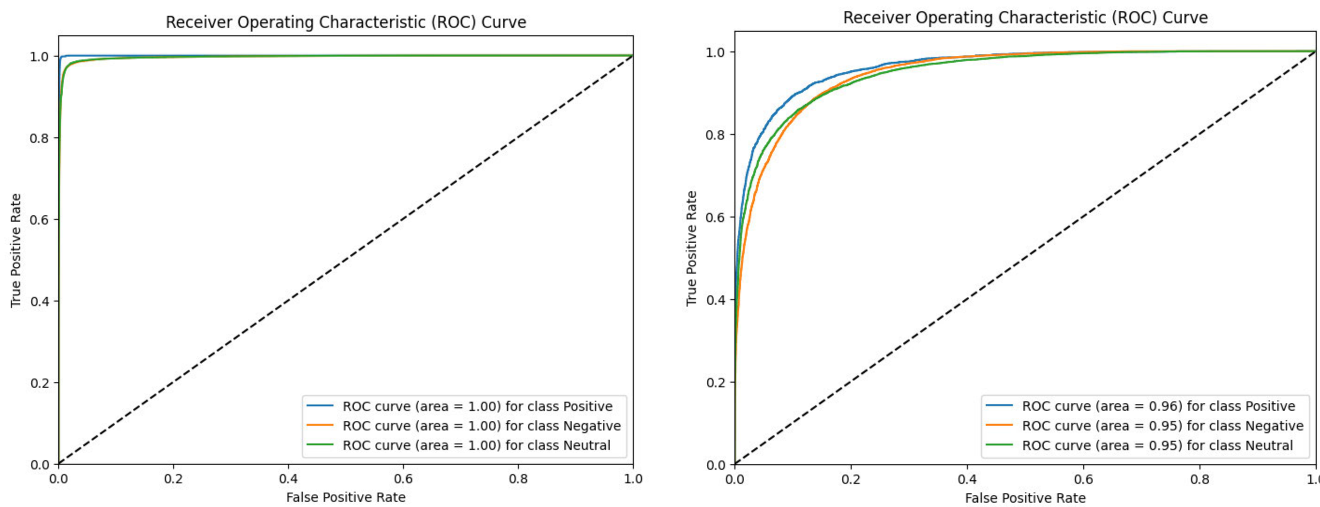
The confusion matrix without data augmentation is shown in Figure 7b. TransNet continues to function very well, although its classification accuracy is substantially less than that of the supplemented model. It accurately identified 9,175



(a) The confusion matrix demonstrates the effectiveness of the TransNet model with augmentation.

(b) The confusion matrix demonstrates the effectiveness of the TransNet model without augmentation.

FIGURE 7. **The diagram depicts the distribution of actual vs anticipated labels across several categories, providing a visual representation of the model's classification precision and the nature of the mistakes made.**



(a) The ROC curve illustrates the effectiveness of the TransNet model with augmentation.

(b) The ROC curve illustrates the effectiveness of the TransNet model without augmentation.

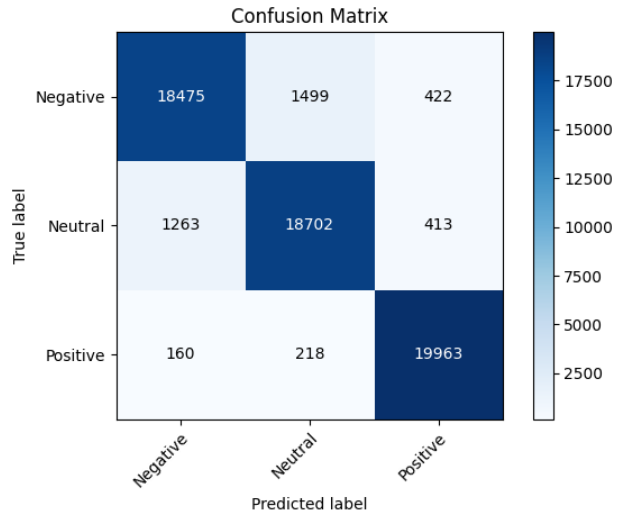
FIGURE 8. **ROC curves were generated for the Fusion Transformer-XL model, representing its performance across different classes. Each curve corresponds to the model's classification performance.**

negative, 7,158 neutral, and 1,788 positive cases, for example. However, there are more misclassifications, including 1,184 neutral cases that were expected to be negative and 304 positive cases that were supposed to be neutral.

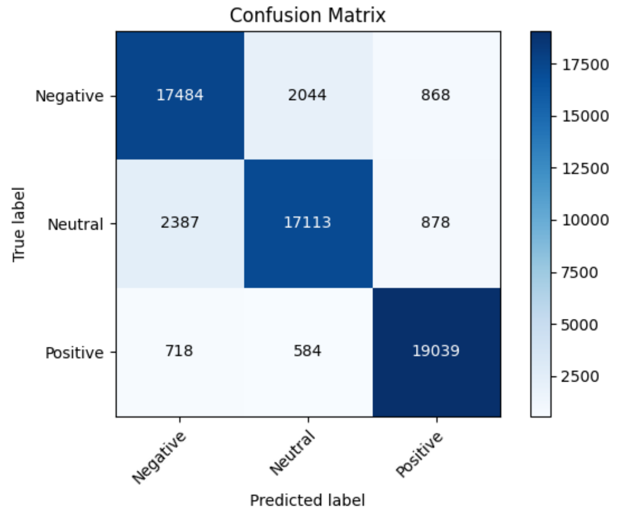
These confusion matrices demonstrate how well data augmentation works to increase TransNet's classification accuracy by lowering mistakes and raising overall precision.

The suggested TransNet model's classification performance is assessed using the Receiver Operating Characteristic (ROC) curves, which are shown in Figure 8 with and without data augmentation. These graphs, which plot the true positive rate versus the false positive rate, demonstrate the model's capacity to discriminate between various classes.

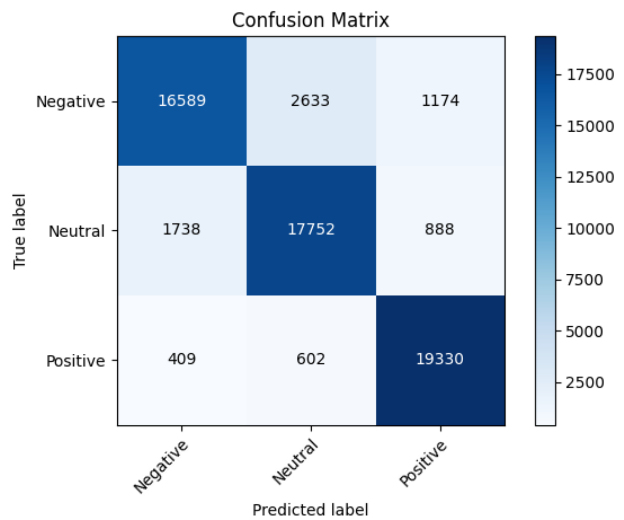
The TransNet ROC curves with data augmentation in Figure 8a demonstrate remarkable performance, with areas under the curve (AUC) for the positive, negative, and neutral classes exceeding 1.00. With 100% sensitivity and specificity for every class, the model demonstrates flawless classification abilities and demonstrates its great precision and dependability in differentiating between labels.



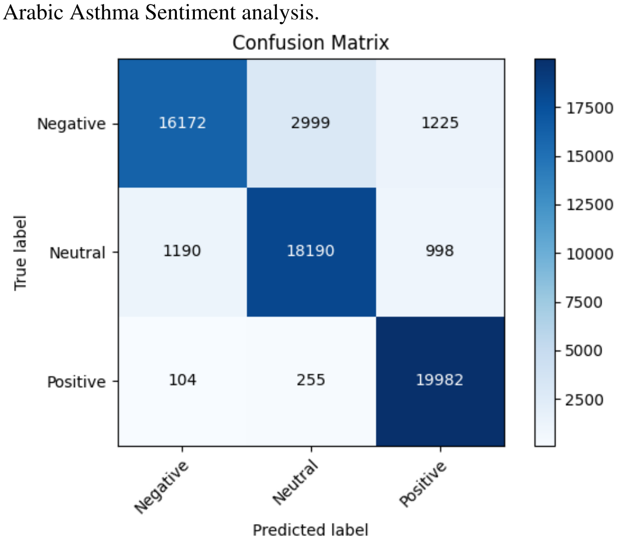
(a) The matrix shows the transformer-LSTM model's effectiveness in Arabic Asthma Sentiment analysis.



(b) The matrix shows the transformer-GRU model's effectiveness in Arabic Asthma Sentiment analysis.



(c) The matrix shows the transformer-CNN model's effectiveness in Arabic Asthma Sentiment analysis.



(d) The matrix shows the transformer-GRU-CNN model's effectiveness in Arabic Asthma Sentiment analysis.

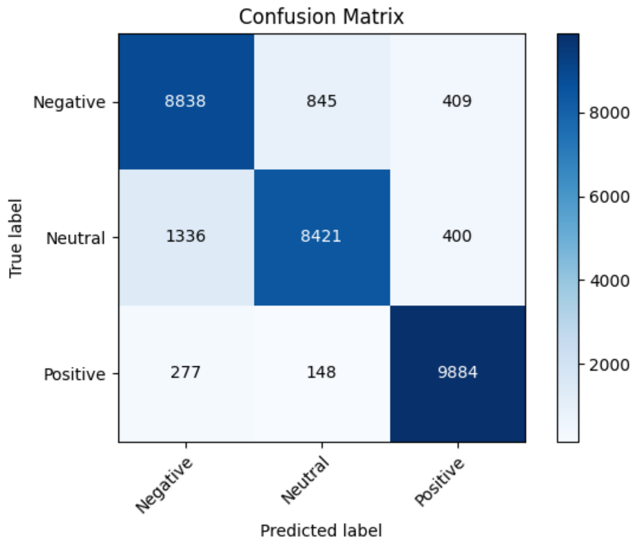
FIGURE 9. The diagram shows the distribution of actual vs. predicted labels, illustrating the baseline model's classification accuracy and error patterns in Arabic Asthma Sentiment analysis while applying data augmentation.

On the other hand, the TransNet ROC curves without data augmentation are shown in Figure 8b. The AUC values are somewhat lower, with 0.98 for positive, 0.95 for negative, and 0.95 for neutral classes, despite the performance being high overall. This implies that in the absence of data augmentation, the model is somewhat less accurate and less capable of flawless classification.

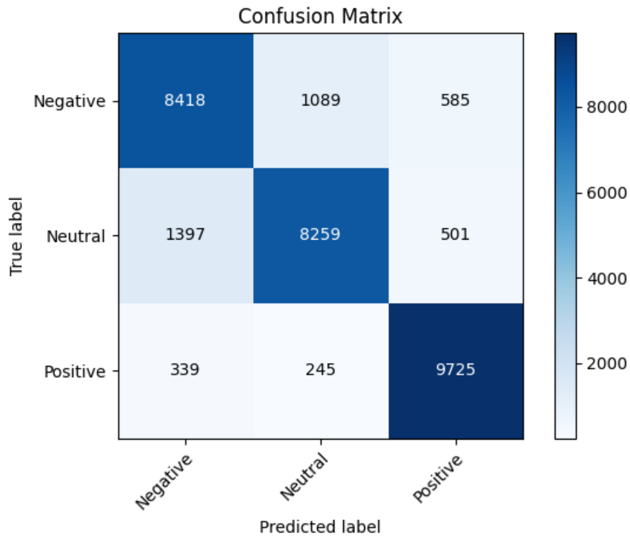
Overall, these ROC curves highlight how data augmentation significantly improved classification performance and improved TransNet's capacity to reliably identify a variety of categories with increased sensitivity and specificity.

E. BASELINE MODELS PERFORMANCE

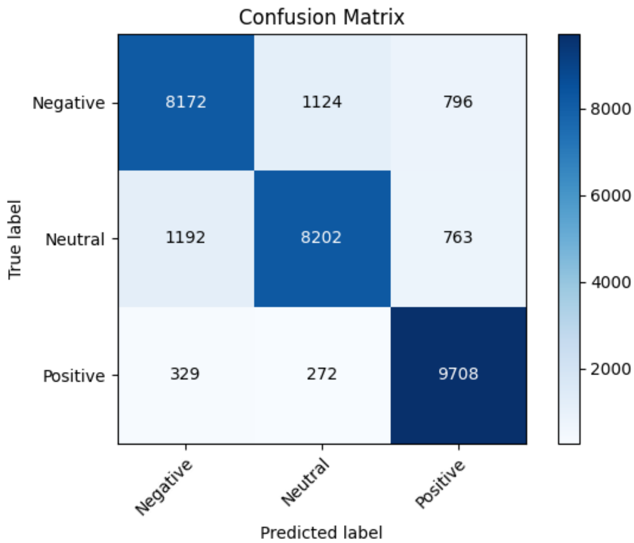
Figure 9 presents the confusion matrices that show how different models performed in the Arabic Asthma Sentiment analysis assignment, emphasizing the use of data augmentation. The actual vs anticipated label distribution for each of the three emotion categories—positive, neutral, and negative—is displayed in subfigures (a-d). The transformer-LSTM model is shown in Subfigure 9a, and it performs well, with a sizable percentage of examples accurately categorized in all categories. The transformer-GRU model, which also has strong classification accuracy, is highlighted in Subfigure 9b. Subfigure 9c illustrates the transformer-CNN model, which has a similar level of performance but a little higher misclassification rate in relation to the other models. Last but not least, subfigure 9d illustrates the transformer-GRU-CNN model, demonstrating its superiority in sentiment analysis while maintaining a significant degree of accuracy in recognizing positive attitudes. Overall, these matrices highlight the usefulness of data augmentation in improving model performance for



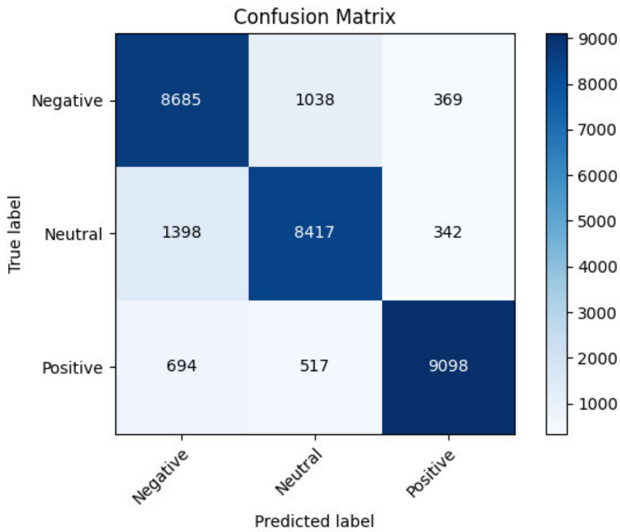
(a) The matrix shows the transformer-LSTM model's effectiveness in Arabic Asthma Sentiment analysis.



(b) The matrix shows the transformer-GRU model's effectiveness in Arabic Asthma Sentiment analysis.



(c) The matrix shows the transformer-CNN model's effectiveness in Arabic Asthma Sentiment analysis.



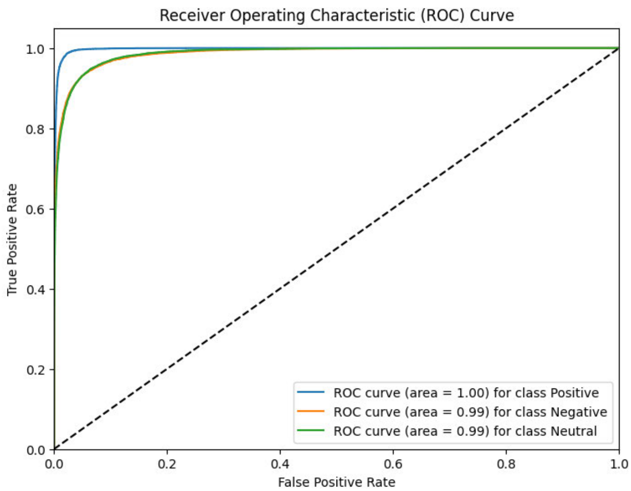
(d) The matrix shows the transformer-GRU-CNN model's effectiveness in Arabic Asthma Sentiment analysis.

FIGURE 10. The diagram shows the distribution of actual vs. predicted labels, illustrating the baseline model's classification accuracy and error patterns in Arabic Asthma Sentiment analysis without data augmentation.

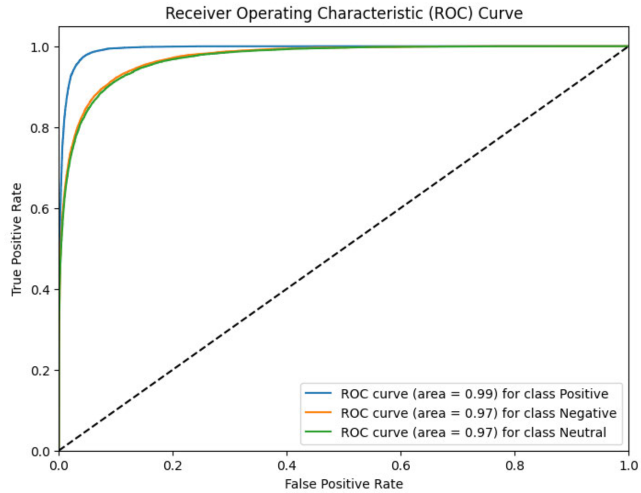
Arabic Asthma Sentiment analysis by offering a thorough overview of each model's classification accuracy and error patterns.

Without the use of data augmentation, Figure 10 displays the confusion matrices showing how different models performed in the Arabic Asthma Sentiment analysis assignment. The distribution of actual vs expected labels for the three sentiment categories—Negative, Neutral, and Positive—is shown in each of the subfigures (a-d). The transformer-LSTM model is shown in Subfigure 10a, where it performs well but has some misclassifications, especially in the Negative and Neutral categories. The transformer-GRU model is shown in Subfigure 10b. It likewise performs well with fewer misclassifications in the Negative category. The transformer-CNN model is shown in Subfigure 10c, which shows comparatively balanced performance in all categories but with a higher number of mistakes in the Negative and Neutral labels. Finally, compared to previous models, the transformer-GRU-CNN model in subfigure 10d exhibits effective classification with fewer misclassifications in the Negative and Neutral categories. These matrices show the intricacies and efficacy of sentiment analysis in Arabic Asthma Sentiment without data augmentation, offering a comprehensive perspective of each model's classification accuracy and error patterns.

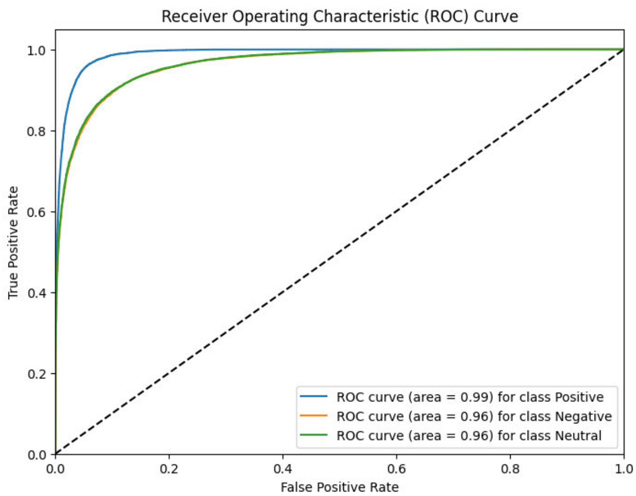
The Arabic Asthma Sentiment analysis baseline models' ROC curves are shown in Figure 11, which also illustrates



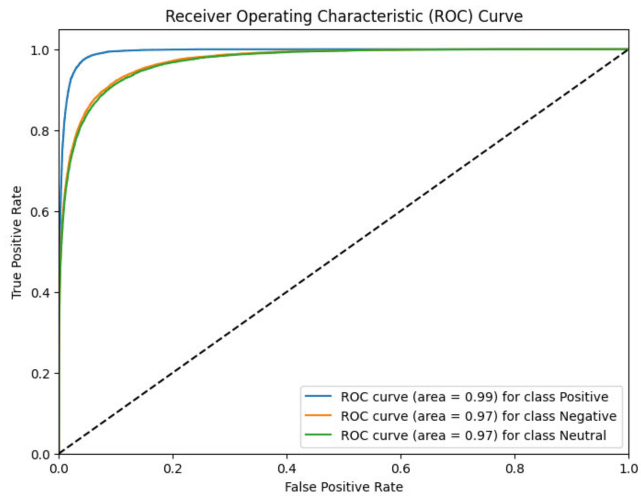
(a) The ROC curve illustrates the effectiveness of the transformer-LSTM model in Arabic Asthma Sentiment analysis.



(b) The ROC curve illustrates the effectiveness of the transformer-GRU model in Arabic Asthma Sentiment analysis.



(c) The ROC curve illustrates the effectiveness of the transformer-CNN model in Arabic Asthma Sentiment analysis.



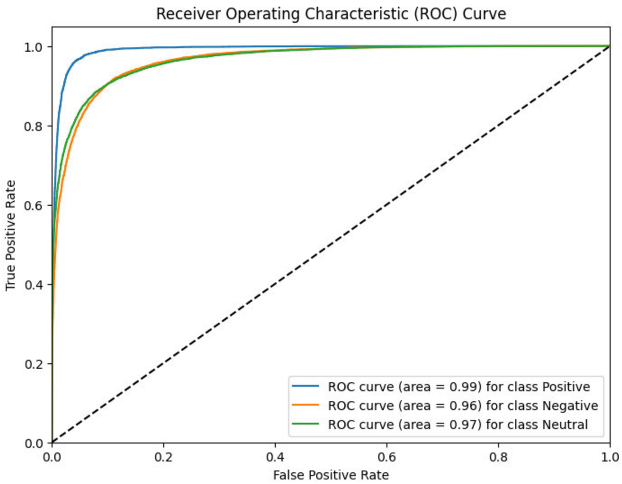
(d) The ROC curve illustrates the effectiveness of the transformer-GRU-LSTM model in Arabic Asthma Sentiment analysis.

FIGURE 11. ROC curves were generated for the baseline models, representing their performance across different classes. Each curve corresponds to the model's classification performance in Arabic Asthma Sentiment analysis while applying data augmentation.

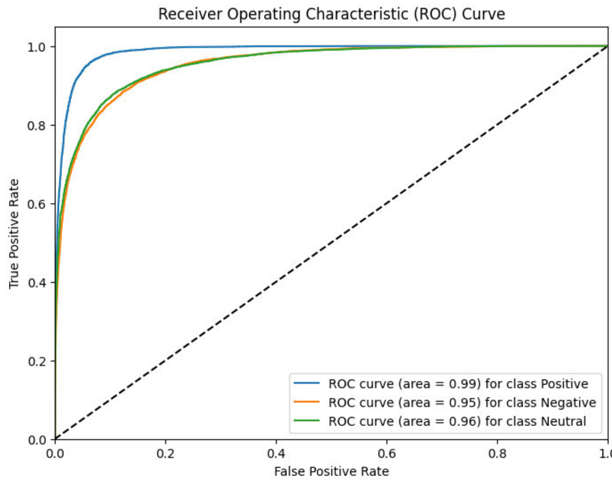
how well the models perform when data is augmented. The transformer-LSTM model is shown in Subfigure 11a, where it exhibits good classification performance with AUC values of 1.00 for Positive, 0.99 for Negative, and 0.99 for Neutral classes. With AUC values of 0.99 for Positive, 0.97 for Negative, and 0.97 for Neutral classes, the transformer-GRU model is shown in Subfigure 11b, suggesting strong efficacy. The transformer-CNN model is shown in Subfigure 11c, where it achieves good accuracy with AUC values of 0.99 for Positive, 0.96 for Negative, and 0.99 for Neutral classes. Finally, the transformer-GRU-LSTM model is shown in subfigure 11d, demonstrating strong performance with AUC values of 0.99 for Positive, 0.97 for Negative, and 0.97 for Neutral classes. These findings imply that the algorithms' capacity to categorize feelings in Arabic Asthma Sentiment analysis correctly is much improved by data augmentation.

F. LIME EXPLANATIONS

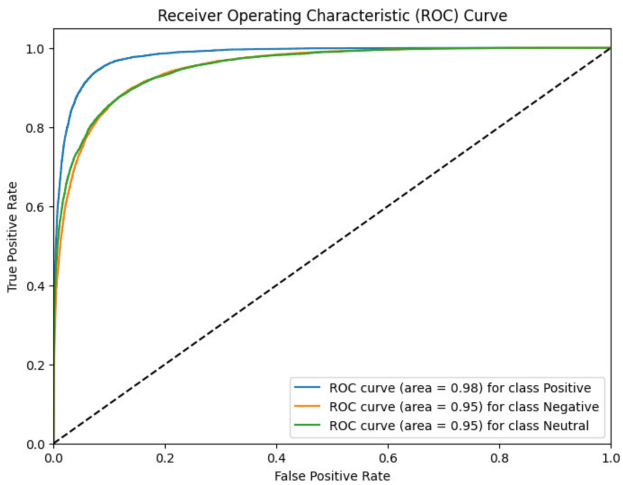
Without data augmentation, Figure 12 shows ROC curves assessing the effectiveness of several transformer-based models in Arabic Asthma Sentiment analysis. The transformer-LSTM model may be shown in Subfigure 12a, where it achieves AUC values of 0.99 for positive, 0.95 for negative, and 0.97 for neutral attitudes. The transformer-GRU model, which likewise attains AUC values of 0.99 for Positive, 0.95 for Negative, and 0.96 for Neutral feelings, is shown in Subfigure 12b. With AUC values of 0.98 for Positive, 0.95 for Negative, and 0.95 for Neutral attitudes, the transformer-CNN model is shown in Subfigure 12c. The transformer-GRU-LSTM model, which attains AUC values of 0.98 for Positive, 0.95 for Negative, and 0.95 for Neutral attitudes, is finally shown in subfigure 12d. Even in the absence of data augmentation, these high AUC values across



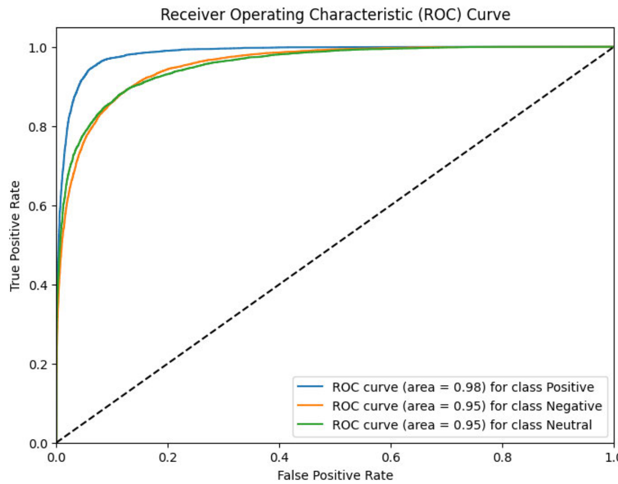
(a) The ROC curve illustrates the effectiveness of the transformer-LSTM model in Arabic Asthma Sentiment analysis.



(b) The ROC curve illustrates the effectiveness of the transformer-GRU model in Arabic Asthma Sentiment analysis.



(c) The ROC curve illustrates the effectiveness of the transformer-CNN model in Arabic Asthma Sentiment analysis.



(d) The ROC curve illustrates the effectiveness of the transformer-GRU-LSTM model in Arabic Asthma Sentiment analysis.

FIGURE 12. ROC curves were generated for the baseline models, representing their performance across different classes. Each curve corresponds to the model's classification performance in Arabic Asthma Sentiment analysis without data augmentation.

all classes show that the models do a good job of accurately categorizing attitudes in the Arabic Asthma dataset.

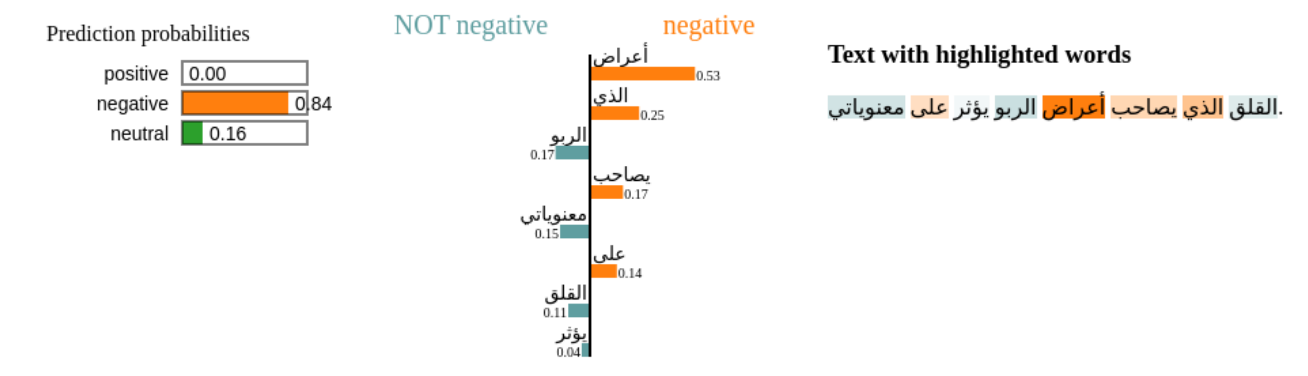
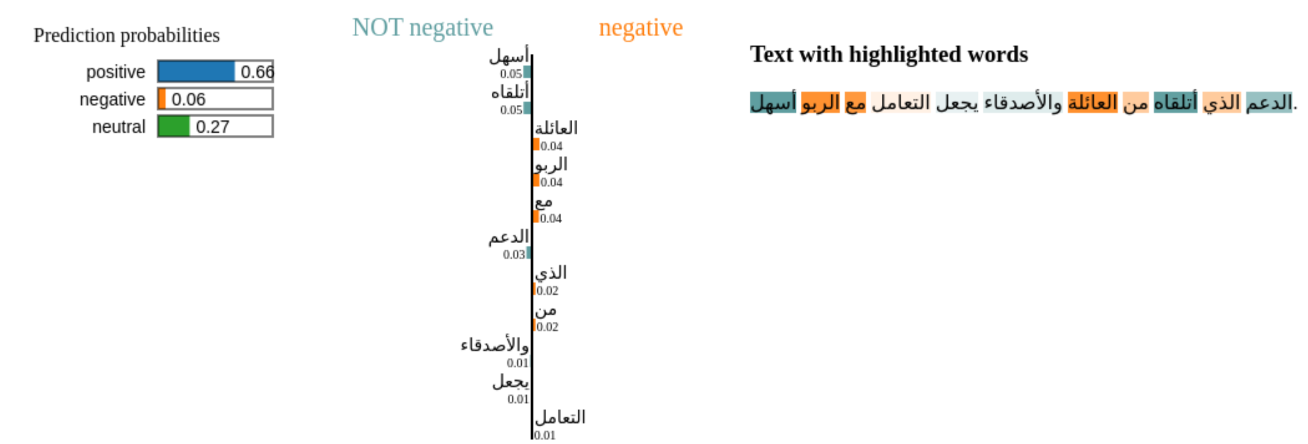
Table 7 illustrates the performance of the suggested TransNet model on various unseen text samples. Each section within the table comprises one column showcasing the actual text and the projected probabilities for each class. The LIME explanation, depicted graphically, highlights specific phrases that had the most substantial impact on the model's decision-making process. This visualization aids in understanding the rationale behind the model's predictions.

In the first example, "القلق الذي يصاحب أعراض الربو يؤثر على معنوياتي " ("The anxiety accompanying asthma symptoms affects my morale"), the negative sentiment prediction of 0.84 is justified by the presence of words like "القلق " (anxiety) and "أعراض " (symptoms), which carry negative connotations.

In the second example, "الدعم الذي أتلقاه من العائلة والأصدقاء يجعل التعامل مع الربو أسهل " ("The support I receive from family and friends makes dealing with asthma easier"), the positive sentiment prediction of 0.66 is supported by words like "أتلقاه " (receive) and "أسهل " (easier), indicating a beneficial effect and, hence, a positive sentiment. The contributions of the words "العائلة " (family) and "الأصدقاء " (friends) further enhance the positive sentiment by emphasizing the supportive relationships. The minor negative sentiment contributions from words like "الدعم " (support) and "التعامل " (dealing) do not significantly affect the overall positive interpretation, as they are neutral terms in the given context. Therefore, the overall sentiment accurately reflects the positive impact of support from family and friends in making it easier to manage asthma.

In the third example, "اليوم أخذت الدواء في الوقت المحدد " ("Today, I took the medication at the specified time"), the neutral sentiment prediction of 0.90 is correct since the text is plain and true. Words such as "الدواء" (medication) and "المحدد" (specified) do not evoke strong

TABLE 7. **The presented table showcases the suggested TransNet model on different unseen text samples. The table consists of one column: the actual text, the words that had a significant impact on the model's choice, and the projected probabilities for each class. The LIME explanation offers elucidation on the specific phrases that had the largest impact on the model's forecast, hence facilitating comprehension of the model's decision-making process.**

Unseen Text




positive or negative feelings, leading to the neutral categorization.

G. COMPARATIVE ANALYSIS AMONG AND EXISTING WORK

Our work focuses on the sentiment analysis of asthma-related posts using various deep learning architectures shown in Table 8. We used a dataset consisting of asthma-related posts collected from X. The applied classifiers included Transformer-LSTM, Transformer-GRU, Transformer-CNN, Transformer-GRU-CNN, and the proposed TransNet model. Among these, the best classifier achieved an accuracy of 97.87%. Our study demonstrates the effectiveness of deep learning techniques in sentiment analysis of health-related social media data.

VI. DISCUSSION

The development and evaluation of TransNet for Arabic post classification demonstrate significant advancements in handling class imbalance and improving text representation in natural language processing tasks. By preprocessing the data with label encoding, upsampling techniques, and data augmentation through synonym replacement, we effectively mitigate class imbalance issues and enhance the diversity of the training dataset. The utilization of a pre-trained BERT tokenizer further refines text representation, leveraging its

TABLE 8. Comparative analysis among and existing work.

Reference	Dataset	Applied Classifier	Best Classifier	Accuracy
Khelil et al. [52]	HARD	SVM, NB, LR, KNN	LR	91%
Abdulsalam et al. [2]	Suicidality Arabic posts	NB, SVM, KNN, RF, XGBoost, AraBert, Ara-ELECTRA, AraGPT2	AraBert	91%
Alshehri et al. [23]	ASAD & ArSAS	BiLSTM, BERT-based ensemble	BERT-based ensemble	88%
Alswaidan & Menai [24]	IAEDS, AETD, SemEval-2018	DF, HEF, HEF+DF	HEF+DF	51%
Elfaik & Nfaoui [38]	ArX, ASTD	CNN+BiLSTM	BiLSTM	92.61%
Elfaik [37]	SemEval-2018	LSTM, LSTM-BiLSTM	LSTM-BiLSTM	53%
Al Wazrah & Al-humoud [74]	Arabic Gulf posts	SVM, GTU, LSTU, AraBERT, Ensemble Model	Ensemble Model	90.21%
Benali et al. [32]	Arabic posts	mBert, QARiB, Arabic Bert, AraBert	MARBERT	67.40%
Alzaid & Fkih [26]	Two X-based datasets, three movie review datasets	NB, RF, LR, KNN, DT, FDT, BiLSTM	Hybrid Fuzzy Model	86%
AlZoubi et al. [29]	Arabic posts	BiGRU_CNN, CNN, XGB, Ensemble	Ensemble	69.2%
Our Work	Asthma posts	Transformer-LSTM, Transformer-GRU, Transformer-CNN, Transformer-GRU-CNN, Proposed TransNet	Proposed TransNet	97.87%

robust language understanding capabilities to capture the nuances of Arabic text more efficiently.

In our comparative analysis, TransNet was benchmarked against several baseline models, including Transformer + LSTM, Transformer + GRU, Transformer + CNN, and Transformer + GRU-CNN. This comprehensive evaluation reveals that TransNet consistently outperforms these baseline models, indicating its superior ability to integrate different neural network architectures and capture complex dependencies in the data. Moreover, the application of LIME (Local Interpretable Model-agnostic Explanations) provides valuable insights into the model's decision-making process, highlighting the most influential features in each prediction. This interpretability is crucial for practical applications, particularly in the healthcare domain, where understanding the rationale behind model predictions can inform more effective and personalized interventions. Overall, the integration of sophisticated preprocessing, advanced model architecture, and interpretability tools in TransNet presents a robust framework for Arabic text classification, showcasing its potential for broader applications and further advancements in NLP.

VII. LIMITATION AND FUTURE WORK

TransNet demonstrates promising performance in Arabic post-classification, but several limitations need to be considered in this study. One notable limitation is the reliance on labeled data for training, which can be costly and time-consuming to acquire, especially for specialized domains such as healthcare. Addressing this limitation could involve exploring semi-supervised or transfer learning approaches to leverage larger unlabeled datasets effectively.

Another challenge is the computational intensity of the model, particularly when multiple complex layers like transformers, LSTMs, and GRUs are integrated. Future work could focus on optimizing the model architecture and exploring more efficient algorithms or hardware accelerators to reduce inference time without compromising accuracy.

Furthermore, while data augmentation techniques like synonym replacement enhance dataset diversity, their effectiveness can vary depending on the specific characteristics of the Arabic language and the domain of interest. Investigating domain-specific augmentation methods or adapting existing techniques for Arabic text could yield further improvements in model robustness.

The evaluation of TransNet primarily focused on binary asthma posts classification from the Kaggle dataset. Future research could extend this evaluation to multi-class classification tasks. Researchers can also consider other related healthcare domains to assess the model's generalizability and scalability across diverse datasets and applications.

Addressing these limitations and exploring these avenues for future work will contribute to advancing TransNet's applicability, performance, and interpretability in Arabic text classification and beyond, facilitating its integration into real-world applications and decision-making processes.

VIII. CONCLUSION

In this study, we introduced TransNet, a deep-attentional hybrid transformer model for Arabic post-classification, specifically focusing on asthma-related content. By using advanced preprocessing techniques such as label encoding, upsampling, and data augmentation through synonym replacement, along with the utilization of a pre-trained BERT tokenizer, TransNet effectively addresses challenges associated with class imbalance and enhances text representation.

Through comprehensive evaluation against baseline models like Transformer+LSTM, Transformer+GRU, Transformer+CNN, and Transformer+GRU-CNN, TransNet consistently demonstrated superior performance, highlighting its ability to integrate diverse neural network architectures and capture intricate dependencies within Arabic text data. The incorporation of LIME for model interpretation further enhances the transparency and trustworthiness of TransNet's predictions, particularly in healthcare applications where understanding the rationale behind predictions is crucial for informed decision-making.

The research can be improved by optimizing computational efficiency, exploring semi-supervised and transfer learning approaches, adapting data augmentation techniques for Arabic text, and extending the evaluation to multi-class classification tasks and other healthcare domains. These efforts aim to enhance TransNet's scalability, generalizability, and interpretability, ultimately advancing its utility in real-world applications and contributing to advancements in natural language processing for Arabic languages.

DATASET AVAILABILITY

Dataset is available at https://www.kaggle.com/datasets/mtesta010/arabic-asthma-tweets

ACKNOWLEDGMENT

The authors thank the Research Chair of Online Dialogue and Cultural Communication at King Saud University, Riyadh, Saudi Arabia, for funding this research.

REFERENCES

[1] M. F. Abdelfattah, M. W. Fakhr, and M. A. Rizka, "ArSentBERT: Fine-tuned bidirectional encoder representations from transformers model for Arabic sentiment classification," *Bull. Electr. Eng. Informat.*, vol. 12, no. 2, pp. 1196–1202, Apr. 2023.

[2] A. Abdulsalam, A. Alhothali, and S. Al-Ghamdi, "Detecting suicidality in Arabic tweets using machine learning and deep learning techniques," *Arabian J. Sci. Eng.*, pp. 1–14, Mar. 2024.

[3] M. E. M. Abo, N. Idris, R. Mahmud, A. Qazi, I. A. T. Hashem, J. Z. Maitama, U. Naseem, S. K. Khan, and S. Yang, "A multi-criteria approach for Arabic dialect sentiment analysis for online reviews: Exploiting optimal machine learning algorithm selection," *Sustainability*, vol. 13, no. 18, p. 10018, Sep. 2021.

[4] F. S. Al-ANZI, "An effective hybrid stochastic gradient descent Arabic sentiment analysis with partial-order microwords and piecewise differentiation," *Fractals*, vol. 30, no. 8, Dec. 2022, Art. no. 2240222.

[5] A. Al-Hashedi, B. Al-Fuhaidi, A. M. Mohsen, Y. Ali, H. A. G. Al-Kaf, W. Al-Sorori, and N. Maqtary, "Ensemble classifiers for Arabic sentiment analysis of social network (Twitter data) towards COVID-19-related conspiracy theories," *Appl. Comput. Intell. Soft Comput.*, vol. 2022, pp. 1–10, Jan. 2022.

[6] W. Al-Qerem, A. Jarab, A. Q. A. Bawab, A. Hammad, J. Ling, F. Alasmari, K. A. Oweidat, and S. Ibrahim, "Assessing the validity and reliability of the Arabic versions of mini asthma quality of life questionnaire and asthma control test in adult patients with asthma: A factor analysis study," *Saudi Pharmaceutical J.*, vol. 31, no. 12, Dec. 2023, Art. no. 101878.

[7] M. Al-Sarem, F. Saeed, A. Alsaeedi, W. Boulila, and T. Al-Hadhrami, "Ensemble methods for instance-based Arabic language authorship attribution," *IEEE Access*, vol. 8, pp. 17331–17345, 2020.

[8] M. Al-Smadi, O. Qawasmeh, M. Al-Ayyoub, Y. Jararweh, and B. Gupta, "Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews," *J. Comput. Sci.*, vol. 27, pp. 386–393, Jul. 2018.

[9] A. Alabdullatif, B. Shahzad, and E. Alwagait, "Classification of Arabic Twitter users: A study based on user behaviour and interests," *Mobile Inf. Syst.*, vol. 2016, pp. 1–11, Jan. 2016.

[10] A. S. Alammary, "BERT models for Arabic text classification: A systematic review," *Appl. Sci.*, vol. 12, no. 11, p. 5720, Jun. 2022.

[11] A. Alduailej and A. Alothaim, "AraXLNet: Pre-trained language model for sentiment analysis of Arabic," *J. Big Data*, vol. 9, no. 1, p. 72, Dec. 2022.

[12] M. A. AlGhamdi and M. A. Khan, "Intelligent analysis of Arabic tweets for detection of suspicious messages," *Arabian J. Sci. Eng.*, vol. 45, no. 8, pp. 6021–6032, Aug. 2020.

[13] A. Alharbi, M. Kalkatawi, and M. Taileb, "Arabic sentiment analysis using deep learning and ensemble methods," *Arabian J. Sci. Eng.*, vol. 46, no. 9, pp. 8913–8923, Sep. 2021.

[14] M. Alhawarat and A. O. Aseeri, "A superior Arabic text categorization deep model (SATCDM)," *IEEE Access*, vol. 8, pp. 24653–24661, 2020.

[15] S. O. Alhumoud, M. I. Altuwaijri, T. M. Albuhairi, and W. M. Alohaideb, "Survey on Arabic sentiment analysis in Twitter," *Int. J. Comput. Inf. Eng.*, vol. 9, no. 1, pp. 364–368, 2015.

[16] S. O. Alhumoud and A. A. Al Wazrah, "Arabic sentiment analysis using recurrent neural networks: A review," *Artif. Intell. Rev.*, vol. 55, no. 1, pp. 707–748, Jan. 2022.

[17] A. Aljohani, N. Alharbe, R. E. Al Mamlook, and M. M. Khayyat, "A hybrid combination of CNN attention with optimized random forest with grey wolf optimizer to discriminate between Arabic hateful, abusive tweets," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 36, no. 2, Feb. 2024, Art. no. 101961.

[18] S. N. Almuayqil, M. Humayun, N. Z. Jhanjhi, M. F. Almufareh, and D. Javed, "Framework for improved sentiment analysis via random minority oversampling for user tweet review classification," *Electron.*, vol. 11, no. 19, p. 3058, 2022.

[19] M. Alotaibi and A. Omar, "An investigation of asthma experiences in Arabic communities through Twitter discourse," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 5, pp. 460–469, 2023.

[20] Y. Alqahtani, N. Al-Twairesh, and A. Alsanad, "Improving sentiment domain adaptation for Arabic using an unsupervised self-labeling framework," *Inf. Process. Manage.*, vol. 60, no. 3, May 2023, Art. no. 103338.

[21] D. Alsaleh and S. Larabi-Marie-Sainte, "Arabic text classification using convolutional neural network and genetic algorithms," *IEEE Access*, vol. 9, pp. 91670–91685, 2021.

[22] O. Alsemaree, A. S. Alam, S. S. Gill, and S. Uhlig, "Sentiment analysis of Arabic social media texts: A machine learning approach to deciphering customer perceptions," *Heliyon*, vol. 10, no. 9, May 2024, Art. no. e27863.

[23] K. Alshehri, A. Alhothali, and N. Alowidi, "Arabic tweet act: A weighted ensemble pre-trained transformer model for classifying Arabic speech acts on Twitter," 2024, *arXiv:2401.17373*.

[24] N. Alswaidan and M. E. B. Menai, "Hybrid feature model for emotion recognition in Arabic text," *IEEE Access*, vol. 8, pp. 37843–37854, 2020.

[25] G. Alwakid, T. Osman, and T. Hughes-Roberts, "Challenges in sentiment analysis for Arabic social networks," *Proc. Comput. Sci.*, vol. 117, pp. 89–100, Jan. 2017.

[26] M. Alzaid and F. Fkih, "Sentiment analysis of students' feedback on e-learning using a hybrid fuzzy model," *Appl. Sci.*, vol. 13, no. 23, p. 12956, Dec. 2023.

[27] S. M. Alzanin, A. M. Azmi, and H. A. Aboalsamh, "Short text classification for Arabic social media tweets," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 9, pp. 6595–6604, Oct. 2022.

[28] A. Alzoubi, A. Alaiad, K. Alkhattib, A. J. Alkhatib, A. Abu Aqoulah, A. B. Alawnah, and O. Hayajnah, "Detection of depression from Arabic tweets using machine learning," *Sustain. Mach. Intell. J.*, vol. 6, pp. 1–3, Mar. 2024.

[29] O. Al Zoubi, S. K. Tawalbeh, and M. Al-Smadi, "Affect detection from Arabic tweets using ensemble and deep learning techniques," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 6, pp. 2529–2539, Jun. 2022.

[30] M. Bani-Almarjeh and M.-B. Kurdy, "Arabic abstractive text summarization using RNN-based and transformer-based architectures," *Inf. Process. Manage.*, vol. 60, no. 2, Mar. 2023, Art. no. 103227.

[31] L. H. Baniata, I. K. E. Ampomah, and S. Park, "A transformer-based neural machine translation model for Arabic dialects that utilizes subword units," *Sensors*, vol. 21, no. 19, p. 6509, Sep. 2021.

[32] B. A. Benali, S. Mihi, N. Laachfoubi, and A. A. Mlouk, "Arabic named entity recognition in Arabic tweets using BERT-based models," *Proc. Comput. Sci.*, vol. 203, pp. 733–738, Jan. 2022.

[33] O. Edo-Osagie, B. De La Iglesia, I. Lake, and O. Edeghere, "Deep learning for relevance filtering in syndromic surveillance: A case study in asthma/difficulty breathing," in *Proc. 8th Int. Conf. Pattern Recognit. Appl. Methods*, 2019, pp. 491–500.

[34] O. Edo-Osagie, G. Smith, I. Lake, O. Edeghere, and B. De La Iglesia, "Twitter mining using semi-supervised classification for relevance filtering in syndromic surveillance," *PLoS ONE*, vol. 14, no. 7, Jul. 2019, Art. no. e0210689.

[35] M. El-Masri, N. Altrabsheh, and H. Mansour, "Successes and challenges of Arabic sentiment analysis research: A literature review," *Social Netw. Anal. Mining*, vol. 7, no. 1, pp. 1–22, Dec. 2017.

[36] Z. Elberrichi, A. Rahmoun, and M. A. Bentaalah, "Using wordnet for text categorization," *Int. Arab J. Inf. Technol. (IAJIT)*, vol. 5, no. 1, 2008.

[37] H. Elfaik and E. H. Nfaoui, "Combining context-aware embeddings and an attentional deep learning model for Arabic affect analysis on Twitter," *IEEE Access*, vol. 9, pp. 111214–111230, 2021.

[38] H. Elfaik and E. H. Nfaoui, "Deep bidirectional LSTM network learning-based sentiment analysis for Arabic text," *J. Intell. Syst.*, vol. 30, no. 1, pp. 395–412, Dec. 2020.

[39] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for Arabic sentiment analysis," *Informat.*, vol. 8, no. 4, p. 79, Nov. 2021.

[40] A. Elnagar, R. Al-Debsi, and O. Einea, "Arabic text classification using deep learning models," *Inf. Process. Manage.*, vol. 57, no. 1, Jan. 2020, Art. no. 102121.

[41] A. Farghaly and K. Shaalan, "Arabic natural language processing: Challenges and solutions," *ACM Trans. Asian Lang. Inf. Process.*, vol. 8, no. 4, pp. 1–22, Dec. 2009.

[42] I. A. Farha and W. Magdy, "Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection," in *Proc. 6th Arabic Natural Lang. Process. Workshop*, 2021, pp. 21–31.

[43] D. Gamal, M. Alfonse, E.-S.-M. El-Horbaty, and A.-B.-M. Salem, "Implementation of machine learning algorithms in Arabic sentiment analysis using N-gram features," *Proc. Comput. Sci.*, vol. 154, pp. 332–340, Jan. 2019.

[44] D. Garreau and U. Von Luxburg, "Explaining the explainer: A first theoretical analysis of lime," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 1287–1296.

[45] M. Habib, M. Faris, A. Alomari, and H. Faris, "AltibbiVec: A word embedding model for medical and health applications in the Arabic language," *IEEE Access*, vol. 9, pp. 133875–133888, 2021.

[46] M. Heikal, M. Torki, and N. El-Makky, "Sentiment analysis of Arabic tweets using deep learning," *Proc. Comput. Sci.*, vol. 142, pp. 114–122, Jan. 2018.

[47] N. Hicham, S. Karim, and N. Habbat, "Customer sentiment analysis for Arabic social media using a novel ensemble machine learning approach," *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 13, no. 4, pp. 4504–4515, Aug. 2023.

[48] M. R. Hossain, M. M. Hoque, N. Siddique, and M. A. A. Dewan, "AraCovTexFinder: Leveraging the transformer-based language model for Arabic COVID-19 text identification," *Eng. Appl. Artif. Intell.*, vol. 133, Jul. 2024, Art. no. 107987.

[49] S. M. Hussain, S. A. Farhana, and S. M. Alnasser, "Time trends and regional variation in prevalence of asthma and associated factors in Saudi arabia: A systematic review and meta-analysis," *BioMed Res. Int.*, vol. 2018, pp. 1–9, Jan. 2018.

[50] I. E. Karfi and S. E. Fkihi, "An ensemble of Arabic transformer-based models for Arabic sentiment analysis," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 8, pp. 561–567, 2022.

[51] E. A. H. Khalil, E. M. F. E. Houby, and H. K. Mohamed, "Deep learning for emotion analysis in Arabic tweets," *J. Big Data*, vol. 8, no. 1, p. 136, Dec. 2021.

[52] H. F. Khelil, M. F. Ibrahim, H. A. Hussein, and R. K. Naser, "Evaluation of different stemming techniques on Arabic customer reviews," *J. Techn.*, vol. 6, no. 2, pp. 1–8, Feb. 2024.

[53] H. Lamtougui, H. El Moubtahij, H. Fouadi, and K. Satori, "An efficient hybrid model for Arabic text recognition," *Comput., Mater. Continua*, vol. 74, no. 2, pp. 2871–2888, 2023.

[54] W. Lan, Y. Chen, W. Xu, and A. Ritter, "An empirical study of pre-trained transformers for Arabic information extraction," 2020, *arXiv:2004.14519*.

[55] T. Limisiewicz, J. Balhar, and D. Marecek, "Tokenization impacts multilingual language modeling: Assessing vocabulary allocation and overlap across languages," 2023, *arXiv:2305.17179*.

[56] R. H. Al Mahmoud, B. H. Hammo, and H. Faris, "Cluster-based ensemble learning model for improving sentiment classification of Arabic documents," *Natural Lang. Eng.*, pp. 1–39, Jun. 2023.

[57] E. A. Mohamed, W. N. Ismail, O. A. S. Ibrahim, and E. M. G. Younis, "A two-stage framework for Arabic social media text misinformation detection combining data augmentation and AraBERT," *Social Netw. Anal. Mining*, vol. 14, no. 1, p. 53, Mar. 2024.

[58] A. Mohammed and R. Kora, "Deep learning approaches for Arabic sentiment analysis," *Social Netw. Anal. Mining*, vol. 9, no. 1, p. 52, Dec. 2019.

[59] U. Naseem, I. Razzak, K. Musial, and M. Imran, "Transformer based deep intelligent contextual embedding for Twitter sentiment analysis," *Future Gener. Comput. Syst.*, vol. 113, pp. 58–69, Dec. 2020.

[60] A. B. Nassif, A. Elnagar, I. Shahin, and S. Henno, "Deep learning for Arabic subjective sentiment analysis: Challenges and research opportunities," *Appl. Soft Comput.*, vol. 98, Jan. 2021, Art. no. 106836.

[61] P. M. O'Byrne, C. Jenkins, and E. D. Bateman, "The paradoxes of asthma management: Time for a new approach?" *Eur. Respiratory J.*, vol. 50, no. 3, Sep. 2017, Art. no. 1701103.

[62] E. Omara, M. Mousa, and N. Ismail, "Character gated recurrent neural networks for Arabic sentiment analysis," *Sci. Rep.*, vol. 12, no. 1, p. 9779, Jun. 2022.

[63] A. H. Ombabi, W. Ouarda, and A. M. Alimi, "Deep learning CNN-LSTM framework for Arabic sentiment analysis using textual information shared in social networks," *Social Netw. Anal. Mining*, vol. 10, pp. 1–13, Dec. 2020.

[64] O. Oueslati, E. Cambria, M. B. Hajhmida, and H. Ounelli, "A review of sentiment analysis research in Arabic language," *Future Gener. Comput. Syst.*, vol. 112, pp. 408–430, Nov. 2020.

[65] H. Pratiwi, R. Benko, and I. Y. Kusuma, "Navigating the asthma network on Twitter: Insights from social network and sentiment analysis," *Digit. HEALTH*, vol. 10, Jan. 2024, Art. no. 20552076231224075.

[66] D. Refai, S. Abu-Soud, and M. J. Abdel-Rahman, "Data augmentation using transformers and similarity measures for improving Arabic text classification," *IEEE Access*, vol. 11, pp. 132516–132531, 2023.

[67] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 1135–1144.

[68] R. M. K. Saeed, S. Rady, and T. F. Gharib, "Optimizing sentiment classification for Arabic opinion texts," *Cognit. Comput.*, vol. 13, no. 1, pp. 164–178, Jan. 2021.

[69] A. Safaya, M. Abdullatif, and D. Yuret, "KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media," 2020, *arXiv:2007.13184*.

[70] H. Saleh, S. Mostafa, A. Alharbi, S. El-Sappagh, and T. Alkhalifah, "Heterogeneous ensemble deep learning model for enhanced Arabic sentiment analysis," *Sensors*, vol. 22, no. 10, p. 3707, May 2022.

[71] S. M. A. H. Shah, S. F. H. Shah, A. Ullah, A. Rizwan, G. Atteia, and M. Alabdulhafith, "Arabic sentiment analysis and sarcasm detection using probabilistic projections-based variational switch transformer," *IEEE Access*, vol. 11, pp. 67865–67881, 2023.

[72] C. Cao, Z. Sun, Q. Lv, L. Min, and Y. Zhang, "VS-TransGRU: A novel transformer-GRU-based framework enhanced by visual-semantic fusion for egocentric action anticipation," 2023, *arXiv:2307.03918*.

[73] H. Tarraf, H. Al-Jahdali, A. H. Al Qaseer, A. Gjurovic, H. Haouichat, B. Khassawneh, B. Mahboub, R. Naghshin, F. Montestruc, and N. Behbehani, "Asthma control in adults in the middle east and North Africa: Results from the ESMAA study," *Respiratory Med.*, vol. 138, pp. 64–73, May 2018.

[74] A. A. Wazrah and S. Alhumoud, "Sentiment analysis using stacked gated recurrent unit for Arabic tweets," *IEEE Access*, vol. 9, pp. 137176–137187, 2021.

[75] R. Ren, Z. Liu, Y. Li, W. X. Zhao, H. Wang, B. Ding, and J. R. Wen, "Sequential recommendation with self-attentive multi-adversarial network," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 89–98.

[76] S. A. Yousif, Z. N. Sultani, and V. W. Samawi, "Utilizing Arabic wordnet relations in Arabic text classification: New feature selection methods," *IAENG Int. J. Comput. Sci.*, vol. 46, no. 4, p. 2019, 2019.

MD. MITHUN HOSSAIN received the degree in computer science and engineering from Bangladesh University of Business and Technology (BUBT). He had a strong emphasis on artificial intelligence (AI) throughout his academic career, which was indicative of his enthusiasm and deep interest in cutting-edge technology. He has a strong foundation in computer science and is a dynamic researcher who is primarily interested in expanding the field of artificial intelligence. His dedication to pushing the limits of AI technology highlights a viable career path in the rapidly changing field of computer science, contributing significantly to both the theoretical and applied elements of intelligent systems. His current research interests include deep learning, machine learning, generative AI, natural language processing, and the quest for artificial general intelligence.

MD. SHAKIL HOSSAIN received the degree in computer science and engineering from Bangladesh University of Business and Technology (BUBT). He has quickly advanced in the fields of robotics, artificial intelligence (AI), machine learning, and deep learning. During his academic tenure, he made substantial contributions to the development of Bengali humanoid robots, BONGO, and Edu Bot, demonstrating his technical prowess and collaborative abilities. He moved easily into the professional world, beginning as an Assistant Engineer at Robo Tech Valley. He improved his robotics and automation talents while working on projects to improve manufacturing operations. His interest in AI and machine learning led him to his current position as an AI Data Trainer at Invisible Technology Inc. His record of achievements is replete with accomplishments, including leading his team to victory at the Digital Technology and Innovation Expo 2019 and excelling in competitions, such as the YESIS12 project exhibition segment at RUET and the Bissuron Renewable Energy competition. His work with Japan–Bangladesh Robotics and Advanced Research Centre (JBRATRC) demonstrates his dedication to collaborative research. He demonstrated his organizational skills by taking on leadership roles in events, such as the NACTAR Robotics Olympiad 2022 and the ROBO TECH OLYMPIAD 2023 As a member of the organizing committee. Furthermore, his active pursuit of research opportunities and contributions to scholarly papers demonstrate his commitment to improving technology and making a significant effect in the field.

MD. SHAKHAWAT HOSSAIN received the Diploma degree in computer engineering and the B.Sc. degree in computer science and engineering (CSE) from Bangladesh University of Business and Technology (BUBT), demonstrating a strong foundation in computer science principles. His scholastic experience has been distinguished by a relentless pursuit of excellence and a desire to master a wide range of technological fields. In this session, he achieved his extra-curricular activities certification on hardware-software integration and system architecture from UEI. These dual degrees in computer engineering demonstrate his wide knowledge base and ability to tackle complex technological difficulties. Throughout his academic studies, he demonstrated a significant interest in artificial intelligence (AI), generative AI, machine learning, deep learning, and exploratory data analysis (EDA) establishing the groundwork for his important contributions to these cutting-edge domains. With a strong educational background and a love for technology, he is well-positioned to push the frontiers of what is possible, emerging as a visionary leader in the ever-changing technological scene, with ambitions to pioneer future research projects.

M. F. MRIDHA (Senior Member, IEEE) received the Ph.D. degree in AI from Jahangirnagar University, in 2017. From 2012 to 2019, he was a CSE Department Faculty Member with the University of Asia Pacific and the Graduate Head. From 2019 to 2022, he was an Associate Professor and the Chairperson of the Department of Computer Science and Engineering, Bangladesh University of Business and Technology. He is currently an Associate Professor with the Department of Computer Science, American International University-Bangladesh (AIUB). He is also the Founder and the Director of the Advanced Machine Intelligence Research Laboratory (AMIR Lab). For more than 15 years, he has been with the master's and undergraduate students as a Supervisor of their thesis work. He has authored/edited several books with Springer and published more than 200 journals and conference papers. His research work contributed to the reputed journals of *Scientific Reports* (Nature), *Knowledge-Based Systems*, *Artificial Intelligence Review*, IEEE Access, *Sensors*, *Cancers*, and *Applied Sciences*. His research interests include computer vision (CV), artificial intelligence (AI), machine learning (ML), deep learning, and natural language processing (NLP). He has served as a program committee member for several international conferences/workshops and an Editorial Board Member of several journals, including *PLOS One* journal. He has also served as a Reviewer for reputed journals, such as IEEE Transactions on Neural Networks, *Artificial Intelligence Review*, IEEE Access, *Knowledge Base System*, *Expert System*, *Bioinformatics*, and *Springer Nature*.

MEJDL SAFRAN received the M.Sc. and Ph.D. degrees in computer science from Southern Illinois University Carbondale, in 2013 and 2018, respectively. He is currently an Assistant Professor with the Department of Computer Science, King Saud University, where he has been a Faculty Member, since 2007, and the Director of the Research Chair of Online Dialogue and Cultural Communication. Since 2018, he has been providing part-time consulting services in the field of artificial intelligence to private and public organizations and firms. He has published papers in refereed journals and conference proceedings, such as *ACM Transactions on Information Systems*, IEEE Transactions on Reliability, IEEE Transactions on Instrumentation and Measurement, *IEEE Industrial Electronics Magazine*, *Journal of Big Data*, *Mathematics*, IEEE International Conference on Cluster, and IEEE International Conference on Computer and Information Science. His research interests include computational intelligence, artificial intelligence, deep learning, pattern recognition, natural language processing, predictive analytics, developing efficient recommendation algorithms for large-scale systems, predictive models for online human activities, machine learning algorithms for performance management, and modeling and analyzing user behavior.

SULTAN ALFARHOOD received the Ph.D. degree in computer science from the University of Arkansas. He is currently an Assistant Professor with the Department of Computer Science, King Saud University (KSU). Since joining KSU in 2007, he has made several contributions to the field of computer science through his research and publications. His research interests include machine learning, recommender systems, linked open data, text mining, and ML-based IoT systems. His work includes proposing innovative approaches and techniques to enhance the accuracy and effectiveness of these systems. His recent publications have focused on using deep learning and machine learning techniques to address challenges in these domains. His research continues to make significant contributions to the field of computer science and machine learning. His work has been published in several high-impact journals and conferences.