IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 8 April 2024, accepted 21 April 2024, date of publication 25 April 2024, date of current version 6 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3393856

RESEARCH ARTICLE

Detection of Sarcasm in Urdu Tweets Using Deep Learning and Transformer Based Hybrid Approaches

MUHAMMAD EHTISHAM HASSAN[1], MASROOR HUSSAIN[1], IFFAT MAAB[2], USMAN HABIB[3], (Senior Member, IEEE), MUHAMMAD ATTIQUE KHAN[4], (Senior Member, IEEE), AND ANUM MASOOD[5]

[1]Department of Computer Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi, Swabi, Khyber Pakhtunkhwa 23460, Pakistan
[2]Technology Management for Innovation, The University of Tokyo, Tokyo 113-8654, Japan
[3]Software Engineering Department, FAST School of Computing, National University of Computer & Emerging Sciences, Islamabad 44000, Pakistan
[4]Department of Computer Science and Mathematics, Lebanese American University, Beirut 1102 2801, Lebanon
[5]Department of Physics, Norwegian University of Science and Technology, 7491 Trondheim, Norway

Corresponding author: Anum Masood (anum.masood@ntnu.no)

ABSTRACT Sarcasm has a significant role in human communication especially on social media platforms where users express their sentiments through humor, satire, and criticism. The identification of sarcasm is crucial in comprehending the sentiment and the communication context on platforms like Twitter. This ambiguous nature of the expression of content presents the detection of sarcasm as a considerable challenge in natural language processing (NLP). The importance and challenges increase further, especially in languages like Urdu where resources for NLP are limited. The traditional rule-based approaches lack the desired performance because of the subtle and context-based nature of sarcasm. However, the recent advancements in NLP, particularly the transformer architecture-based large language models (LLMs) like BERT offer promising solutions. In this research, we have utilized a newly created Urdu sarcasm dataset comprising 12,910 tweets manually re-annotated into sarcastic and non-sarcastic classes. These tweets were derived from the public Urdu tweet dataset consisting of 19,995 tweets. We have established baseline results using deep learning classifiers comprising CNN, LSTM, GRU, BiLSTM, and CNN-LSTM. To comprehensively capture the contextual information, we propose a novel hybrid model architecture that integrates multilingual BERT (mBERT) embeddings with BiLSTM and multi-head attention (MHA) for Urdu sarcasm. The proposed mBERT-BiLSTM-MHA model demonstrates superior performance by achieving an accuracy of 79.51% and an F1 score of 80.04%, outperforming deep learning classifiers trained with fastText word embeddings.

INDEX TERMS Urdu sarcasm detection, NLP, deep learning, LLMs, BERT embeddings, hybrid model, multi-head attention.

I. INTRODUCTION

The uprise of digital platforms such as social media, discussion forums, and messaging platforms has changed the mode of communication [1]. People use these platforms to express their opinions on various issues, products, and services. This data generated online is used by many organizations including government and private sectors to get insight into public sentiments and opinions regarding products, services, and events. This poses a need for effective data handling serving the purpose of management and analysis as a result of an increase in the expansion of information [2]. Numerous techniques in the literature, including statistical, machine learning,

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

and deep learning (DL) have been employed to analyze and classify the large-scale data collected from social media [3], [4]. For NLP tasks using text classification, DL-based techniques have been used extensively and have yielded improved performance in comparison to other approaches [5], [6].

Sentiment analysis (SA) may be quite beneficial in understanding people's actual thoughts and behaviors concerning a specific issue as well as in comprehending individual opinions and behavior [7]. Approaches like SA can be thought of as text categorization methods that primarily consider language and emotions that people communicate through textual and non-textual content [8]. Sarcasm is a common way for people to voice their thoughts on social media, which makes it challenging to analyze and a primary problem in SA [9].

Sarcasm can be considered as a type of sentiment in which users or people use positive words to convey their negative opinions [10]. For example, the sentence "The government is doing a wonderful job in eliminating people and poverty" is sarcastic demonstrating negative sentiment regarding government performance with words having positive opinions such as "a wonderful job". These inconsistencies and shifts in polarities make sarcasm detection an essential part of enhancing the performance of SA. Detection of sarcasm can be challenging, as its topic and domain dependency increase task complexity. Specifically, the posts on social media which are informal texts further enhance the challenge to analyze sarcasm. The complexity can not be reduced unless considerable information is available regarding the specific topic, context relating to the situation, and the domain [11].

Sarcasm detection in SA is not limited to the English language, which is rich in dialect but can be extended to other languages such as Urdu which has comparatively small footprints. The Urdu language can be considered among languages that are deprived of resources and there is a need for standard datasets as the availability of language resources to scholars is very limited [12]. The Urdu language is widely spoken and the national language of Pakistan [13]. However, it's also widespread in the states of India and Bangladesh due to its historical roots. Sarcasm detection in Urdu SA holds importance as for other dialects. Studies have been conducted on Urdu SA, where various approaches such as lexicon [14], [15], [16], machine learning [17], [18], and deep learning [19], [20] have been employed for sentiment classification. The summary of the Urdu datasets from the literature is presented in Table 1. However, no previous research on sarcasm detection on Urdu text has been performed to the best of our knowledge and no standard dataset is available for Urdu sarcastic tweets. This research is focused on detecting sarcasm in Urdu tweets using DL classifiers and transformer-based hybrid neural network architecture.

The main contributions of this study can be enumerated as follows:

- Establish a balanced Urdu sarcasm dataset for sarcasm identification in Urdu tweets.
- We present a novel hybrid model named mBERT-BiLSTM-MHA for sarcasm detection that employs mBERT which is further enhanced with a Bidirectional Long Short-Term Memory model (BiLSTM) combined with a MHA mechanism.
- Set baseline results obtained from sarcasm detection in Urdu tweets using deep learning models and proposed hybrid model with an attention mechanism.

The distribution of the remainder paper is structured as follows: Section II provides an overview of related literature on sarcasm detection. Section III delineates the methodology employed in this study. Section IV and Section V report the experimental setup and discuss the obtained results. Finally, Section VI concludes the paper, highlighting potential future directions.

II. RELATED WORK

The categorization of sarcasm is a comparatively active topic of research in NLP and various experiments have been performed for its detection. However, research in this domain is notably limited, particularly in low-resource Asian languages such as Hindi and Punjabi. The scarcity of research extends to the Urdu language, as our findings yielded no pertinent studies focusing on sarcasm detection. The following sections report past research on Urdu SA, sarcasm detection, and an overview of our approach to Urdu sarcasm detection.

A. SENTIMENT ANALYSIS IN URDU

Hashim and Khan [24] explored SA in Urdu news headlines using a dataset with 1,000 opinions. They developed sentiment lexicons for news text and achieved an accuracy of 86%. Their approach demonstrated the effectiveness of lexicons in analyzing sentiments in Urdu news articles. Another study by Ahmad and Edalati [23] used lexicon and rule-based approaches to classify sentiments in Urdu tweets collected from social media. They used five algorithms (Naïve Bayes (NB), Vader, Textblob, Stanza, and Flair) to categorize tweets into positive and negative classes. Notably, they attained a 70% accuracy using Flair on the Urdu corpus. Sana et al. [22] developed two Urdu emotion datasets comprising 1,000 reviews in the domain of smartphones and 1,100 sports reviews. They classified the emotions associated with the tweets using five machine learning (ML) classifiers comprising NB, Support Vector Machines (SVM), K-Nearest Neighbor (KNN), and Random Forest (RF). SVM classifier demonstrated superior performance, achieving an accuracy of 80.5% on the smartphone corpus and 81.09% on the sports dataset. Mukhtar et al. [17] created a corpus comprising 6,025 sentences from various genres of Urdu blogs. Their study focused on sentiment analysis subjectivity. They manually classified sentences into neutral, negative, and positive categories. KNN classifier achieved

TABLE 1. Overview of current Urdu datasets.

Corpus	Description	Classes	Algorithms	Acc (%)	Corpus Available
Urdu SA [19]	10,008 reviews (Different genres)	3	Rule-based, N-gram, machine learning and deep learning	68.56	Yes
Urdu Text SA [21]	6,000 sentences (Blogs/News)	2	Deep learning	77.9	No
Urdu SA [20]	9,312 reviews (Various domains)	3	Rule-based, machine learning and deep learning	77.61	Yes
Urdu Emotion dataset [22]	1,000 tweets (Smartphone)	6	Machine learning	80.5	Yes
Urdu Emotion dataset [22]	1,200 tweets (Sports)	6	Machine learning	81.09	Yes
Urdu Tweet SA [23]	10,000 reviews	2	Lexicon and rule-based	70	Yes
Urdu Blogs [17]	6,025 sentences (Different genres)	3	Machine learning	-	No
Urdu News [24]	1,000 opinions (News headlines)	3	Lexicon based	86	No

maximum performance with an accuracy of 67.01% among the classifier utilized in the study. In a study on Urdu SA using a human-annotated dataset of 10,000 reviews, Safder et al. [19] employed a range of algorithms to analyze sentiments. They adopted n-gram models, rule-based approaches, SVM, and DL models namely: Recurrent Convolutional Neural Network (RCNN), Convolutional Neural Network (CNN), and LSTM. The DL model RCNN outperforms other classifiers achieving an accuracy of 68.56% for ternary categorization. Khan et al. [20] presents a comprehensive investigation into SA in the Urdu language. Establishing a manually annotated dataset comprising 9,312 reviews spanning various domains, the research explores the efficacy of different methodologies comprising rule-based systems, ML algorithms, and DL techniques (i.e., CNN and LSTM). Text representation schemes of n-gram features and fastText embeddings were adopted in the study.

B. SARCASM CLASSIFICATION

1) LEXICON-BASED APPROACHES FOR SARCASM DETECTION

A lexicon-based approach was adopted by Riloff et al. [25], in which they used a bootstrapping technique to learn positive and negative sentiment phrases in English tweets by maintaining bags-of-lexicons with n-gram phrases i.e., unigram, bigram, and trigram. Prominent similar studies by Camp [26] and Joshi et al. [9] on automatic sarcasm detection exhibit sarcasm presence in a sentence characterized by polarity conflict and juxtaposition. A parsing-based approach for lexical generation was adopted by Bharti et al. [27] for identifying sarcasm in English tweets. They developed more broader approach by using four bags corresponding to lexicons which comprised of two sentiments (positive and negative) and two situations (positive and negative). Moreover, these phrases were utilized to identify the sarcasm occurrence using positive sentiment in a situation that was negative and determining sarcasm in a positive situation with negative sentiment.

2) RULE-BASED APPROACHES FOR SARCASM DETECTION

Another approach in the literature that is effective in sarcasm detection is the rule-based approach that utilizes the semantic and syntactic aspects of the sentence. This approach is effective in identifying sarcasm in sentences as it uses the phrase pattern and analyzes the lexical structure pertaining to the sentence in any language. The semantic model, which emphasizes more on the words meaning, their structure, their relationship with other words in the language, and their contextual usage, is the most effective rule-based approach [28]. Bharti et al. [27] used a semantic-based approach for the identification of sarcasm in Hindi tweets with two rule-based algorithms. The study identifies sarcasm using the combination of an interjection along an enhancer. A rule-based approach was also presented by Riloff et al. [25] that determines the appearance of verbs in a sentence with positive emotion and negative situation. An iterative algorithm is proposed in the study that extracts positive verbs and negative situation phrases from English tweets. The rule-based approaches have limitations and may not produce the desired performance in sarcasm detection when there isn't any explicit satire indicated in the sentence [29].

3) MACHINE LEARNING APPROACHES FOR SARCASM DETECTION

ML algorithms have been applied for NLP tasks especially for SA, due to their stability feature and high performance [30], [31], [32]. Machine learning algorithms has the capacity to self-observe in accordance with the specified specification and dataset. Onan [33] applied an ML approach for identifying sarcasm in English tweets using classifiers namely: NB, logistic regression (LR), SVM, and KNN. They also employed n-gram features and TF-IDF weighting schemes for text representation. A different study by González-Ibánez et al. [34] employed dictionary-based lexical features, unigrams, and pragmatic aspects along with SVM and LR classifiers to identify sarcasm in Twitter tweets. The machine learning approach relies on manual feature construction and to identify sarcasm automatically, researchers have been exploring DL techniques.

4) DEEP LEARNING AND HYBRID APPROACHES FOR SARCASM DETECTION

DL methods can induce features automatically through neural networks that are employed to learn from large datasets [35]. These techniques based on deep learning are making it possible to automatically detect sarcasm by capturing the subtle and long-range semantic characteristics, which can be challenging to capture using manual feature engineering. Ghosh et al. [36] used an attention mechanism in conjunction with LSTM to detect sarcasm in discussion forums and tweets. The study's conclusions demonstrate that utilizing an LSTM with sentence-level attention on both context and reply outperforms an LSTM model focused solely only the reply. Zhang et al. [37] used a bi-directional Gated Recurrent Neural Network (Bi-GRU) in another study to locally extract syntactic and semantic data from English tweets. Additionally, the authors also utilize a pooling neural network to automatically extract contextual information from users' past tweets. Using a hybrid technique for sarcasm detection, Ghosh and Veale [38] demonstrated improved performance by combining an LSTM, a Deep Neural Network (DNN), and a CNN. To develop a robust and effective automatic sarcasm detection system, the research examines DL and ML classifiers using a range of features. Poria et al. [39] used another deep learning-based technique that utilizes pre-trained CNN for sarcasm detection. They employed CNN to extract sentiment, emotion, and personality attributes in the study. In order to prioritize contextual features over lexical and syntactic patterns, Amir et al. [40] construct a DNN. By learning user embeddings, their network contributes to the improvement of context-based sarcasm detection. Hazarika et al. [41] presented a framework to encode the features using user profiling and utilizing user embedding. Performance in terms of classification on a large corpus is considerably enhanced when utilized in conjunction with content-based feature extractors like CNNs. Plepi and Flek [42] introduced a graph attention network to identify sarcasm in English tweets on social media. Their approach involves capturing the complex connections between sarcastic tweets and their conversational context by explicitly modeling users' social and historical context together.

C. AUTOMATIC URDU SARCASM DETECTION

In this research, we employed DL models and proposed a novel mBERT-BiLSTM-MHA model for Urdu sarcasm detection. The related work provides insights into very limited research on Urdu SA particularly sarcasm detection. This research focuses exclusively on text for sarcasm identification in each individual post. Our research creates robust baseline results for Urdu sarcasm identification across many social platforms. The prediction of Urdu sarcastic text is a strictly binary classification task, wherein each sentence is assigned to one of two categories: sarcastic or non-sarcastic. Compared to sarcasm, sentiment analysis (SA) is not perceived to be a strictly binary problem as the text can be categorized into multiple sentiment classes, such as positive, negative, or neutral [23], [24].

The general approach for the categorization of sarcastic text involves various stages. The initial stage comprises data acquisition, which includes gathering text data and annotation using various tools and techniques. The data acquired then goes through pre-processing steps which structures and normalizes that input data. The next stage is feature extraction to transform input into an array of descriptive features. The use of classifiers from DL or ML and the performance assessment of these models come next. We have used the Urdu sarcastic tweet dataset [43], which was further cleaned and annotated to Urdu-specific sarcastic tweets. The annotation was done manually on the corpus, details are provided in the following subsequent sections. The pre-processing steps are performed to normalize the data for word embedding features and deep learning classifiers are applied for the automatic detection of sarcasm in Urdu text. The performance of the applied classifiers is accessed taking into consideration the evaluation metrics. Figure 1 depicts the overall flow diagram of the DL models architecture used in our research for sarcasm detection.

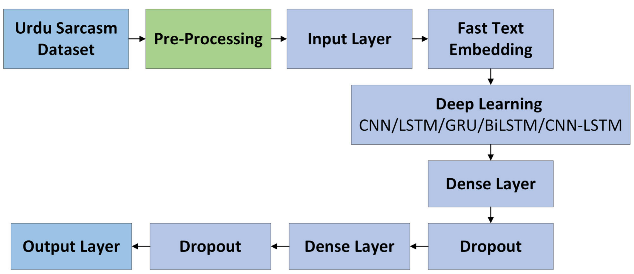

FIGURE 1. Flow diagram of deep learning models.

III. METHODOLOGY

In our experimental study, we investigate deep learning models namely, CNN, LSTM, GRU, Bi-LSTM, and CNN-LSTM, leveraging fastText word embeddings for identifying sarcasm. These classifiers are selected based on their notable performance in NLP-related tasks [29], [44], [45], [46], [47]. Moreover, we propose a hybrid mBERT-BiLSTM-MHA approach that uses mBERT embeddings and applies Bi-LSTM with MHA to effectively capture sarcasm. The detail of the classifiers and the attention mechanism adopted in the experiment analysis is explained in the following subsections.

A. CONVOLUTIONAL NEURAL NETWORK (CNN)

CNNs are DL architectures that extract significant information from the training data automatically using convolution operations. These operations after several applications grasp contextual local features from a sentence and form a global feature vector. This automatic feature extraction mechanism allows CNN not to rely on hand-crafted features that are used in conventional supervised classifiers. We utilized a CNN architecture tailored for our experiment on Urdu sarcastic sentences. CNNs have been effective in Urdu SA [19], [20], considering their traditional application in computer vision. In our approach, each sentence was represented as a two-dimensional array of fastText embeddings. A one-dimensional CNN (Conv1D) layer with 32 filters and a kernel size of 4 was applied to the embeddings to extract local features. To prevent overfitting, L2 regularization with a coefficient of 0.01 was employed. Max-pooling was then applied to resulting feature maps to aggregate the most important features, resulting in a consolidated representation of the sentence. Subsequently, these features were fed into fully connected layers for further processing and prediction. Early stopping mechanisms are implemented to monitor model performance and prevent overfitting during training iterations.

B. LONG SHORT TERM MEMORY (LSTM)

LSTM exhibits the recurrent neural architecture with memory units compared to RNNs, which facilitates addressing the vanishing and exploding gradient problem associated with the vanilla RNN model [48]. This architecture has been widely applied in various text classification problems [36], [44], [49], due to its ability to capture long-term dependencies that are present in sequential data. In our approach for Urdu sarcasm detection, we used fastText embeddings to preprocess the text data before passing it through a 64-unit LSTM layer. The LSTM layer's ability to retain state over prolonged time enables it to effectively leverage the context of each word within the sentence, thus enhancing sarcasm detection accuracy. The LSTM output at each timestep is passed a fully connected dense layer for prediction. The dense layer aggregates the information across all time steps and produces a fixed-length vector representation. The vector is then passed into subsequent dense layers, which further process the information to make predictions. Dropout layers are incorporated between these dense layers to prevent overfitting. This process allows the LSTM model to effectively capture the sequential dependencies within the input text and utilize them to make accurate predictions on sarcastic text.

C. GATED RECURRENT UNIT (GRU)

GRU sequential models were introduced as an alternative to LSTMs and exhibit the same performance as LSTM in a range of tasks [50]. GRU also exhibits similarities with LSTM in addressing the challenges of vanishing and exploding gradient issues that arise in RNNs. However, architectural simplicity allows GRU to perform more rapidly compared to LSTM and makes it easier to compute [51]. The information flow through the GRU unit is controlled and modulated through the update and reset gates. GRU has produced significant performance in NLP problems such as opinion mining [45], [52]. In our experiment, we employed a GRU layer with 64 units for Urdu sarcasm detection. Each unit in the GRU layer processes the embeddings, capturing dependencies between words through gates that regulate information flow. After processing, the output from the GRU layer is fed to fully connected dense layers. The final output neuron, with a sigmoid activation function, produces the sarcasm prediction.

D. BIDIRECTIONAL LONG SHORT TERM MEMORY (BI-LSTM)

Bi-LSTMs are sequential models that were proposed to address the limitations present in LSTM due to its unidirectional processing of text input [53]. Bi-LSTM has the ability to analyze text in both forward and backward directions, allowing the model to utilize the contextual relationship of past and future. This bi-directional contextual information contributes to a more comprehensive interpretation of sarcasm within the text [54].

We utilized Bi-LSTM to leverage their ability to process text in both directions ensuring a comprehensive interpretation of contextual relationships, crucial for understanding sarcasm nuances. BiLSTM layer with 80 units was used in our study for processing the input sequences in both forward and backward directions. The output from the Bi-LSTM layer, acquired by aggregating the forward and backward hidden states at each time step, serves as the foundation for subsequent analysis. The mathematical formula for the Bi-LSTM layer is provided as follows:

For time steps $t = 1, 2, 3, \ldots T$, let X be the input sequence and $h_t^{(f)}$ represent the forward hidden states and $h_t^{(b)}$ represent backward hidden states at time step t.

$$h_t^{(f)} = \text{LSTM\_Forward}(x_t, h_{t-1}^{(f)}) \tag{1}$$

$$h_t^{(b)} = \text{LSTM\_Backward}(x_t, h_{t-1}^{(b)}) \tag{2}$$

In the equation (1) and equation (2), *LSTM_Forward* denotes the forward LSTM layer and *LSTM_Backward* denotes the backward LSTM layer.

$$h_t = [h_t^{(f)}, h_t^{(b)}] \tag{3}$$

where h_t is the output of Bi-LSTM at time t from the merge of forward and backward hidden states and x_t is the respective input at each time step t.

Following the Bi-LSTM layer, we incorporated two fully connected dense layers in our model architecture. The output neuron makes the final prediction indicating the likelihood of the input text being sarcastic or non-sarcastic.

E. CNN-LSTM

A hybrid CNN-LSTM model can produce high performance and reduced error measures on various NLP tasks compared to employing CNN and LSTM models in individual capacity [47], [55]. We used a hybrid CNN-LSTM model

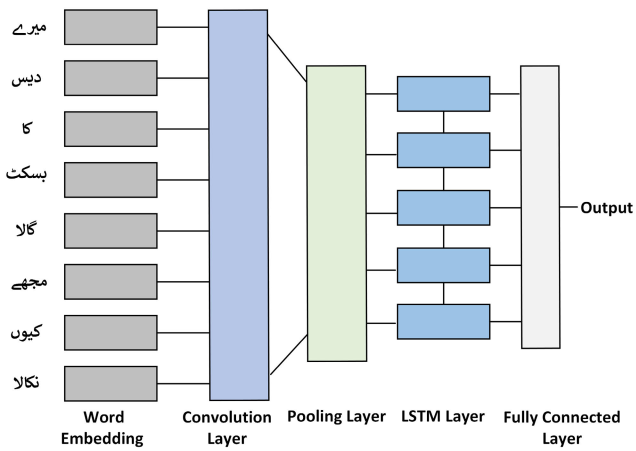

FIGURE 2. The hybrid CNN-LSTM architecture.

in our study to identify sarcasm leveraging the strengths of both CNNs and LSTM networks. The CNN-LSTM model employed comprises an initial convolutional layer comprising 32 units, which processes the word embedding input and extracts local features. These local features are then passed to a max pooling layer to reduce dimensionality before being fed to the LSTM layer as shown in Figure 2. The LSTM layer configured with 32 units in our experiment, is effective in capturing the long-term dependencies, which are challenging for CNNs to capture due to the localized nature of convolutional layers [56]. The output of the LSTM layer is passed to fully connected dense layers for prediction.

F. MULTI HEAD ATTENTION MECHANISM

The MHA mechanism, which focuses on different segments of the input sequence, significantly impacts the extent to which the deep learning model captures different contexts. Introduced by Vaswani et al. [57], MHA mechanism focuses on various aspects of the information by concatenating the output of multiple attention mechanisms that are executed concurrently.

1) SCALED DOT-PRODUCT ATTENTION

Attention is calculated independently for each head using the scaled dot-product attention mechanism. The attention method uses the dot product of Query (Q) and Key (K) to compute the attention scores, scaling the result to prevent the dot product from growing noticeably. The scaling operation is useful for mitigating issues related to vanishing and exploding gradients. The acquired weights are subjected to the *softmax* function for normalization and then multiplied with the value (V). The following is the mathematical representation of the scaled dot-product attention mechanism:

$$\mathrm{Attention}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \tag{4}$$

In the above equation (4), d_k shows the key vectors dimensionality. With the use of this attention mechanism, the model can give various sequence segments varying degrees of priority.

2) MULTI HEAD ATTENTION

The MHA works in parallel across numerous attention heads, where each attention head has its own acquired linear projections for Q, K, and V before being passed to a scaled dot-product attention mechanism. The architecture of the MHA mechanism is presented in Figure 3. Later, the outputs from these heads undergo concatenation and linear transformation.

The mathematical formula is provided below:

$$\mathrm{Multihead}(Q, K, V) = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)W_O \tag{5}$$

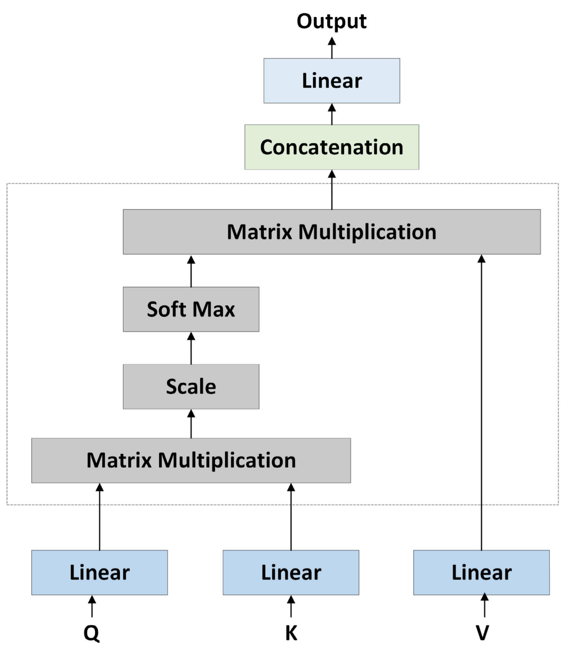

FIGURE 3. Flow chart representation of MHA mechanism.

The weight matrix for the final linear transformation is denoted by W_O, where h stands for the number of attention heads. The model's capacity to process various input components concurrently is ensured by concatenation.

G. PROPOSED HYBRID MBERT-BILSTM-MHA MODEL

The approach that we propose in this paper is a combination of mBERT and BiLSTM with multi-head attention to detect sarcasm in Urdu Tweets using text-only. This section provides a brief background of mBERT before discussing the proposed mBERT-BiLSTM-MHA approach. Figure 4 illustrates the architecture of the proposed mBERT-BiLSTM-MHA model.

1) MBERT

Bidirectional encoder representations from transformers, or BERT, is a generalization-exhibiting pre-trained language

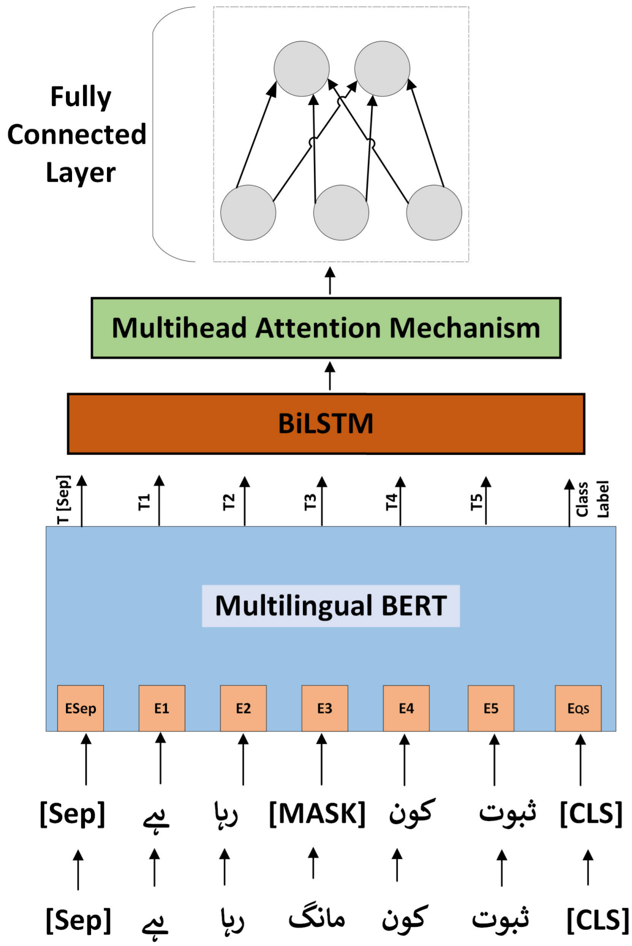

FIGURE 4. Proposed mBERT-BiLSTM-MHA model architecture.

model that can be tailored to user requirements for a variety of downstream tasks [58]. The BERT architecture utilizes the use of a bi-directional transformer, which allows the model to better comprehend word context through effective pre-training [58]. The past approaches for language representation comprised techniques like one hot encoding and more effective neural representations based embedding schemes like GloVe and word2vec [59]. Fixed-length feature vectors are generated by these models to represent words. BERT provides a notable improvement over traditional models like word2vec by capturing context-dependent representations. These dynamic representations provide a more nuanced understanding of words within their context. There are two phases to the BERT model training process: pre-training and fine-tuning. During the unsupervised pre-training phase, the BERT is trained using tasks such as next sentence prediction (NSP) and masked language model (MLM). In the MLM task, one or more words in a sentence are randomly masked, and the model predicts the hidden words by accounting for the unmasked words. The BERT model trained with the NSP task requires a relationship among two sentences. Using the sentence-pair information as input, the model predicts the following sentence. If the two sentences are sentence C and sentence D, the model will predict if sentence D follows sentence C as the next sentence on the corpus. The training of the BERT on two tasks i.e., MLM and NSP allows the model to capture semantic information at both declarative and textual levels. The supervised fine-tuning phase utilizes the task-specific labeled data to further train the pre-trained BERT model with minimum modifications to the BERT model for a particular downstream NLP task. Based on sizes, the BERT architecture additionally has further two variations: BERT-Base and BERT-Large. The 12 encoder layers, 768 hidden units, and 12 attention heads comprise the BERT-Base model. 1024 feed-forward units, 16 attention heads, and 24 encoder layers constitute the BERT-Large. When performing NLP categorization tasks, the input is augmented with an initial [CLS], and the final hidden state generated is then collected for the subsequent classification procedures.

mBERT [60] is the multilingual version of BERT sharing the same APIs along with architecture, and both are pre-trained in a similar unsupervised way. However, mBERT is pre-trained on a corpus comprising Wikipedia corpora that spans 104 languages including Urdu [12] compared to the BERT model that is trained on the single English corpus sourced from Wikipedia and Books Corpus [61]. The study by Pires et al. [60] shows that a relatively simple method for zero-shot-cross-lingual transfer is accomplished by mBERT and no explicit cross-lingual signals, such as word pairings or documents connected between languages are used in training. Using data from a language that is easily accessible and assessing the task in a desired language with fewer resources makes it simpler to fine-tune the model for a given task in any language that the model is familiar with. This aids in the creation of models that translate information between languages.

2) MBERT-BILSTM-MHA

Fine-tuned mBERT model has shown significant performance on downstream Urdu sentiment analysis tasks [20], showing the effectiveness of the model in Urdu-specific NLP problems. Identifying dependencies in the word embeddings produced by the pre-trained mBERT model is crucial for improving mBERT's performance on Urdu sarcastic sentences. To handle this task a BiLSTM layer is capable of learning the contextual information and capturing the sequential dependencies. For example the phrase, "You should be treated like a bouquet of flowers" conveys positive sentiment initially, however, in Urdu culture, it can be used sarcastically to convey the opposite meaning. In such cases, standalone mBERT and BiLSTM models might not produce the desired performance. mBERT can provide contextual embeddings but capturing these subtle cues, such as context accurately is challenging and leads to misinterpretations. BiLSTM has limitations in capturing all intricate dependencies in complex sentences, especially

Algorithm 1 Algorithm for Capturing Urdu Sarcasm Using the Proposed mBERT-BiLSTM-MHA Model

Input: Urdu sarcasm dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, where x_i denotes tweet in Urdu and y_i is its sarcastic and non-sarcastic label
Output: Sarcastic/Non-Sarcastic
$(Ur\_train, Ur\_test, labels\_train, labels\_test) \leftarrow$ SplitDataset (D)
tokenizer ← InitializeTokenizer () *// Initialize the BERT tokenizer*
TokenizeData (tokenizer, Ur_train, labels_train) *// Tokenize training data*
TokenizeData (tokenizer, Ur_test, labels_test) *// Tokenize testing data*
x_train ← GenEmb (x_train) *// Generate BERT embeddings for training data*
x_test ← GenEmb (x_test) *// Generate BERT embeddings for testing data*
model ← InitializeModel () *// Initialize BiLSTM with Attention classifier architecture*
trained_model ← TrainModel (model, x_train, x_test)
EvaluateModel (trained_model, x_test)
return: trained_model

Urdu. Combining both the approaches allows model to better comprehend the complex language structures, utilizing mBERT contextual comprehension and BiLSTM abilities to capture sequential dependencies. Moreover, BiLSTM with an attention mechanism allows the model to learn broader contextual relationships in sarcastic sentences as compared to BiLSTM [54].

We propose a hybrid mBERT-BiLSTM-MHA model to achieve the desired performance on Urdu text by applying BiLSTM with MHA on top of mBERT for sarcasm detection. The base architecture of mBERT with 12 encoder layers and 768 hidden units is employed to generate contextualized embeddings denoted as $(T_1, T_2, \ldots, T_n)$ for individual tokens in the input sequence as shown in Figure 4. Each T_i is a multidimensional vector that encodes information specific to the token in the input sequence. The sequential dependencies present in the Urdu mBERT embeddings are captured by the BiLSTM layer enabling the model to comprehend both past and future context. MHA mechanism is utilized on the output of the BiLSTM layer to capture complex linguistic patterns in Urdu sarcastic text. The MHA mechanism captures the relevant context by focusing simultaneously on different segments of the input sequence. During the training phase, the attention mechanism plays a crucial role in enabling the model to assign distinct weights to different parts of the Urdu sequence during backpropagation. The layer helps in making the learning more effective by highlighting the importance of the particular tokens in the context of Urdu sarcasm. The final output is obtained by processing the attended representation from MHA by a fully connected layer. The layer integrates the binary cross-entropy loss (BCE) function, facilitating model training. The BCE loss inherently combines with a sigmoid activation function, ensuring proper transformation of model outputs into probabilities for accurate classification.

Algorithm 1 outlines the proposed approach process for identifying sarcasm, where initially the dataset is split into training and testing sets. Subsequently, the BERT tokenizer is initialized for Urdu text, followed by the tokenization of tweets from both sets and the generation of BERT embeddings. A Bi-LSTM with MHA is then set up and trained on the training data for sarcasm detection. The sequence of operations applied to process the input data by employing the BiLSTM and MHA mechanism is detailed in Algorithm 2. Initially, the query (Q), key (K), and value (V) matrices are obtained from the embeddings of the BERT. The BERT embeddings are fed to the BiLSTM layer which captures the sequential dependencies and produces an output sequence. The next step is applying MHA, which involves linear transformations to the query, key, and value matrices (Q_i, K_i, and V_i, respectively) for each attention head i. The attention mechanism is then utilized to compute the attention scores Z_i using these transformed matrices. The attention outputs $Z_1, \ldots, Z_h$ from all the attention heads are concatenated to establish a single matrix Z which contains the aggregated information from all the attention heads. Finally, the attention matrix Z is projected onto a new space using a weight matrix W_O to get Z' that assists the model the proposed model in learning high-level features.

Algorithm 2 Multihead Attention on Top of mBERT-BiLSTM

Input: BERT embeddings Q,K,V
Output: Concatenated attention matrix Z'
$Z \leftarrow$ InitializeMatrix
$h \leftarrow$ NumberOfAttentionHeads
$W_{Qi}, W_{Ki}, W_{Vi} \leftarrow$ InitializeWeightMatrices
$W_O \leftarrow$ InitializeProjectionWeightMatrix
$Q,K,V \leftarrow$ ObtainQKVMatrices (Q,K,V)
for i from 1 to h **do**
  $Q_i, K_i, V_i \leftarrow$ LinearTransformations $(Q,K,V,W_{Qi},W_{Ki},W_{Vi})$
  $Z_i \leftarrow$ Attention (Q_i,K_i,V_i)
  AppendToMatrix (Z_i, Z)
end for
$Z \leftarrow$ ConcatenateAttentionOutputs $(Z_1,\ldots,Z_h)$
$Z' \leftarrow$ ProjectAttentionMatrix (Z, W_O)
return Z'

Sarcastic Urdu tweet example	Explanation
حکومت موجودہ عوام اور مہنگائی سے تنگ آ چکی ہے (The government is fed up with the current people and inflation)	Despite appearing to express concern, the statement sarcastically implies dissatisfaction or criticism towards the government's response to these issues
جس شہر کا آئی جی اغوا ہو جائے وہاں کے باسیوں کو گھبرانے کہ اجازت ہونا چاہئیے (Residents of the city where the IG is kidnapped should be allowed to panic)	The statement sarcastically ridicules the inability of law enforcement to maintain order by suggesting that panic is justified in such situations.
عوامی عدالت کے فیصلہ کی اگر کوئی حیثیت ہوتی تو ہم ٹک ٹاک نہیں کھلوالیتے (If there was any significance to the decision of the public court, we wouldn't be playing TikTok)	The statement sarcastically suggests that if public court decisions had any importance, people wouldn't waste their time playing TikTok, mocking the lack of significance given to such decisions.

FIGURE 5. Sample sarcastic tweets from Urdu sarcasm dataset.

IV. EXPERIMENT

This section presents the corpus details, preprocessing steps, text representation scheme, and the setup of the experiments performed in our study.

TABLE 2. Proposed Urdu sarcasm corpus details.

Characteristics	Urdu Sarcasm Dataset
Total count of tweets in corpus	12910
Sarcastic	6455
Non-Sarcastic	6455
Total count of tokens	194722
Per tweet average tokens	18.85
Max length	79

A. URDU SARCASM DATASET

We manually annotated the sarcastic Urdu tweet dataset [43] publicly available on Kaggle, which comprises 19,955 tweets labeled as sarcastic and non-sarcastic to create a standard Urdu sarcasm dataset. A balanced annotated dataset of 12,910 tweets was acquired after cleaning and preprocessing. The dataset comprises 6,455 sarcastic tweets and 6,455 non-sarcastic tweets related to sports, politics, and social issues. The dataset was annotated by two experts who were native speakers of the Urdu language and master students. The experts were asked to label the tweets as sarcastic if they exhibited mocking or ridiculing, exaggeration, irony, and negation. Sentences in which positive words were used to express negative statements were also annotated as sarcastic.

The sample sarcastic tweets are presented in Figure 5 and the characteristics of the annotated dataset are presented in Table 2. The Inter-Annotator agreement of 80.37 percent was achieved using Cohan's Kappa method [62] for 1,000 tweets annotated by each. The third expert was a professor with relevant experience in the Urdu language. The third expert supervised the annotation process by reviewing the results after every 500 tweets to ensure the validity and appropriate annotation pattern.

B. PREPROCESSING

Preprocessing is performed on the input text to transform and prepare the text data for feature extraction [63]. The preprocessing steps adopted in the study include:

- Removing digits, numbers, punctuations, URLs, single Urdu alphabets, hashtags, emojis, and stop words.
- Removed complete Pushto, Hindi, English, Punjabi, and Arabic language sentences.
- *UrduHack* library [64] was used to remove white spaces, normalize the tweets, and lemmatization.
- We performed Unicode normalization to ensure consistent representation of Urdu characters. The normalization filters out characters that range outside the Unicode range for Urdu script (0600-06FF).

C. FEATURE EXTRACTION

Processing and understanding the text data in NLP tasks by machine learning algorithms requires the transformation of text data to numerical features. Pre-trained word

TABLE 3. Parameter of the deep learning models used in the experiment analysis.

Parameter	CNN-1D	LSTM	GRU	Bi-LSTM	CNN-LSTM
Layer 1 Dimension	32	64	64	80	32
Layer 2 Dimension	32	32	64	64	32
Layer 2 Dropout	0.2	0.1	0.3	0.2	-
Layer 3 Dimension	16	16	16	32	32
Layer 3 Dropout	0.1	0.1	0.2	0.1	0.3
Layer 4 Dimension	-	-	-	-	16
Layer 4 Dropout	-	-	-	-	0.2
Epoch	20	30	15	10	10
Learning rate	0.0002	0.001	0.001	0.0002	0.0001
Regularization	0.01	-	-	-	-

embeddings [65] are a feature learning technique in NLP that maps vocabulary words with high-dimensional vectors of real values. The ability to fine-tune for a particular task makes these models more effective. In this study, we have used a fastText embedding model [65] trained on Wikipedia and Common Crawl corpus. The fastText embeddings have been trained to comprehend over 150 dialects of different languages, including Urdu. The ability to capture the semantic relation in Urdu words makes fastText word embeddings a better feature representation approach for this research. The deep recurrent neural models and the hybrid CNN-LSTM classifier used in this research capture the semantic similarity using fastText with *SpaCy* library tokenizer for Urdu NLP. The proposed robust hybrid model mBERT-BiLSTM-MHA uses highly contextualized mBERT embeddings for Urdu words. The proposed model utilizes the BERT tokenizer provided by the *Transformers* library for tokenization.

D. EXPERIMENT SETUP

The experiments in the study are performed on Google Colaboratory which provides free access to graphical processing units (GPU). The standard metrics of Accuracy, Precision, F1-score, and Recall are used to evaluate the performance of each model. A data split of 0.2 was implemented, with 80% of the data allocated for training and 20% for testing. Table 3 presents the parameters of CNN, LSTM, GRU, BiLSTM, and CNN-LSTM models adopted in the study, which produced the best results in identifying sarcasm in tweets. The parameters of the DL models are selected after multiple trials and hits. The models are trained using a batch size of 32, employing the Adam optimizer and the BCE loss function. Tweets are categorized into sarcastic and non-sarcastic classes through the utilization of the sigmoid activation function in the final output layer.

The parameter selection for the proposed model presented in Table 4 was finalized through multiple iterations. The mBERT architecture was configured with 12 layers and each BiLSTM contained 32 units. The overall neural network has 2 layers. During training, a batch size of 32 was utilized and BCE loss was used to evaluate model performance. Additionally, Adam optimizer was employed to adjust the model parameters iteratively. The training process spanned across 15 epochs and a learning rate of 2e-05 was chosen for gradual adjustments during training. Early stopping was used to select the best training epochs for each classifier in the experiment. The training was halted upon observing a decline in model validation accuracy or an increase in validation loss, indicating overfitting of the model on the training data.

TABLE 4. Proposed mBERT-BiLSTM-MHA hybrid model hyperparameters assigned values in the experiment analysis.

Hyperparameter	Value
mBERT layers	12
BiLSTM units	32
No of layers	2
Attention heads	2
Batch size	32
Epochs	15
Learning rate	2e-05
Loss	BCE
Optimizer	Adam

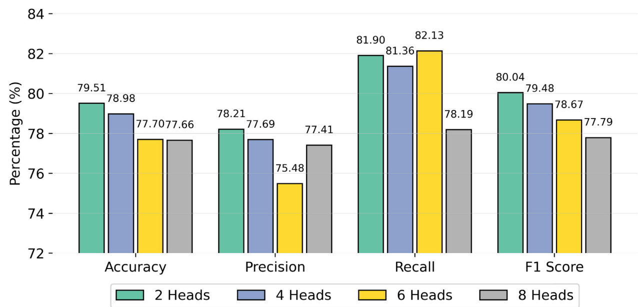

FIGURE 6. Performance comparison of the mBERT-BiLSTM-MHA model with 2, 4, 6, and 8 attention heads.

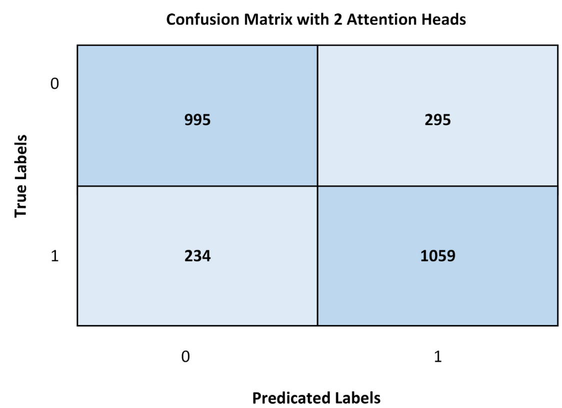

FIGURE 7. Confusion matrix of the proposed model.

V. RESULTS AND DISCUSSION

In this section, we present the results attained by the proposed model and deep learning models used in our study.

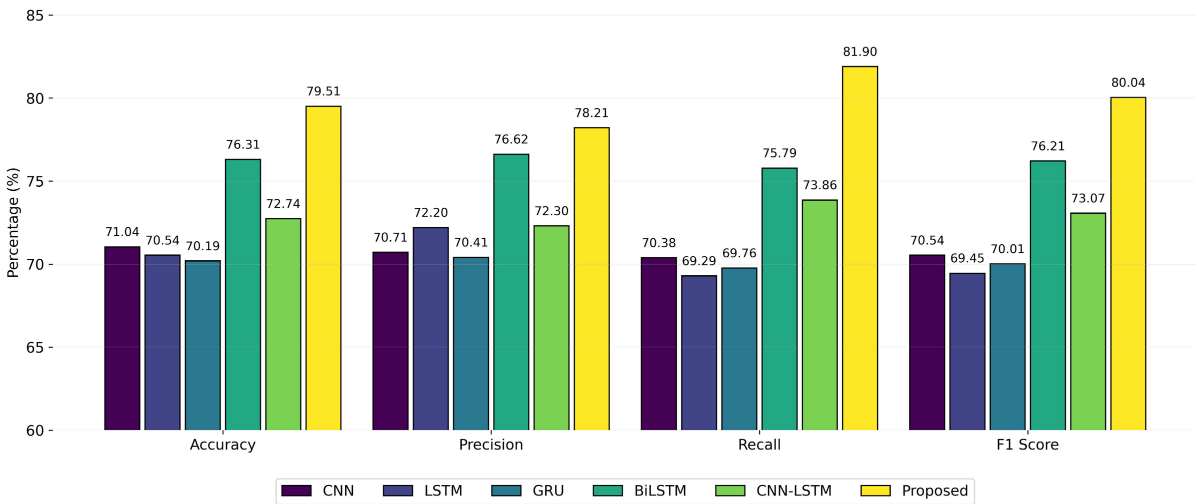

FIGURE 8. Performance comparison of the proposed model with other deep learning models applied on Urdu Sarcasm Dataset.

TABLE 5. Performance of the proposed model on different attention heads.

Attention heads	Accuracy	Precision	Recall	F1 Score
2	79.51	78.21	81.90	80.04
4	78.98	77.69	81.36	79.48
6	77.70	75.48	82.13	78.67
8	77.66	77.41	78.19	77.79

A. PERFORMANCE OF THE PROPOSED MODEL

The proposed mBERT-BiLSTM-MHA model achieves significant performance on the text-based Urdu sarcasm identification task. The performance of the proposed model was evaluated on 2, 4, 6, and 8 attention heads to capture the relevant context in the Urdu text. The performance achieved with these attention heads is presented in Table 5. The robust hybrid model achieves the best accuracy of 79.51% and an F1 score of 80.04% respectively with 2 attention heads as shown in Figure 6. The confusion matrix gives a precise analysis of the model's performance by revealing its ability to correctly classify instances and identify any misclassifications. The acquired confusion matrix of the mBERT-BiLSTM-MHA model with two attention heads is depicted in Figure 7.

B. COMPARISON WITH BASELINE MODELS.

The performance of baseline classifiers and the proposed model used in the experiment analysis is presented in Table 6. CNN achieves an accuracy of 71.04%, demonstrating the capability of the CNN model to capture local traits that are useful in determining the context [66]. The recurrent neural networks of LSTM and GRU acquire an accuracy of 70.54% and 70.19% respectively, indicating almost similar performance on the Urdu sarcasm identification task. However, the GRU model has a slightly lower precision of 70.41% and a recall of 69.76%. The hybrid CNN-LSTM has

TABLE 6. Comparison of results of the proposed model with other state-of-the-art models on Urdu sarcasm dataset.

Model	Accuracy	Precision	Recall	F1 Score
CNN	71.04	70.71	70.38	70.54
LSTM	70.54	72.20	69.29	69.45
GRU	70.19	70.41	69.76	70.01
BiLSTM	76.31	76.62	75.79	76.21
CNN-LSTM	72.74	72.30	73.86	73.07
Proposed	**79.51**	**78.21**	**81.90**	**80.04**

achieved better performance on the task with an accuracy of 72.74% and an F1 score of 73.07% compared to standalone CNN and LSTM models. This significant performance can be related to the hybrid CNN-LSTM architecture, where the convolution layer extracts local features which are passed to LSTMs for sequence prediction [67]. The BiLSTM has the best performance among the baseline classifiers used in the experiment analysis with accuracy, precision, recall, and F1 score values of 76.31%, 76.62%, 75.79%, and 76.21% respectively. The high performance of BiLSTM indicates the model's efficacy in NLP tasks and its ability to capture bidirectional contextual information [46]. The proposed mBERT-BiLSTM-MHA outperforms all the baseline models and attains the highest accuracy, precision, recall, and F1 score of 79.51%, 78.21%, 81.90%, and 80.04% respectively as shown in Figure 8. This highest performance implicates the capability of our proposed hybrid model to learn intricate linguistic patterns from rich vector representation of mBERT and focus on the relevant context in the text by employing the MHA mechanism in conjunction with BiLSTM.

VI. CONCLUSION AND FUTURE DIRECTIONS

Sarcasm detection on content generated by users using social media platforms is crucial for detailed SA. This research introduced a novel robust learning model that can detect sarcasm exclusively in Urdu text. The model utilized the Bi-LSTM to capture sequential dependencies in the contextual representation from mBERT and focused on different segments of the input in a simultaneous manner to capture relevant contextual information using the MHA mechanism. The proposed mBERT-BiLSTM-MHA achieves the highest classification accuracy of 79.51% and an F1 score of 80.04% on the predefined performance benchmarks used to evaluate the performance. The high performance compared to the employed baseline models adopted in the study indicates the effectiveness of mBERT-BiLSTM-MHA in identifying sarcasm using text only.

This research paves a path for sarcasm detection in the Urdu language using deep learning and hybrid approaches that utilize pre-trained large language model embedding and attention schemes to capture linguistic patterns. The research findings indicate its potential applicability in other low-resource languages, like Roman Urdu, which is often utilized on social media platforms, particularly in Pakistan and India. Our study provides a baseline for sarcasm detection in the Urdu language and further research is critical to investigate other approaches comprising multimodal that can be used to identify sarcasm in languages with limited resources.

ACKNOWLEDGMENT

The authors would like to thank the Directorate of the Science and Technology Department, GIK Institute, Pakistan, and The University of Tokyo, Japan, for the super-computing facility.

CONFLICT OF INTEREST

We hereby declare that we do not have any financial or non-financial conflict of interest related to the submitted work.

REFERENCES

[1] J. P. Wihbey, *The Social Fact: News and Knowledge in a Networked World*. Cambridge, MA, USA: MIT Press, 2019.

[2] L. Zhou, S. Pan, J. Wang, and A. V. Vasilakos, "Machine learning on big data: Opportunities and challenges," *Neurocomputing*, vol. 237, pp. 350–361, May 2017.

[3] S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," *Inf. Fusion*, vol. 36, pp. 10–25, Jul. 2017.

[4] A. R. Sharma and P. Kaushik, "Literature survey of statistical, deep and reinforcement learning in natural language processing," in *Proc. Int. Conf. Comput., Commun. Autom. (ICCCA)*, May 2017, pp. 350–354.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[6] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He, "A survey on text classification: From traditional to deep learning," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 2, pp. 1–41, Apr. 2022.

[7] K. Sailunaz and R. Alhajj, "Emotion and sentiment analysis from Twitter text," *J. Comput. Sci.*, vol. 36, Sep. 2019, Art. no. 101003.

[8] A. Kumar and G. Garg, "Empirical study of shallow and deep learning models for sarcasm detection using context in benchmark datasets," *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 5, pp. 5327–5342, May 2023.

[9] A. Joshi, P. Bhattacharyya, and M. J. Carman, "Automatic sarcasm detection: A survey," *ACM Comput. Surv.*, vol. 50, no. 5, pp. 1–22, 2017.

[10] S. K. Bharti, B. Vachha, R. K. Pradhan, K. S. Babu, and S. K. Jena, "Sarcastic sentiment detection in tweets streamed in real time: A big data approach," *Digit. Commun. Netw.*, vol. 2, no. 3, pp. 108–121, Aug. 2016.

[11] A. Kumar and G. Garg, "Systematic literature review on context-based sentiment analysis in social multimedia," *Multimedia Tools Appl.*, vol. 79, nos. 21–22, pp. 15349–15380, Jun. 2020.

[12] L. Khan, A. Amjad, N. Ashraf, and H.-T. Chang, "Multi-class sentiment analysis of Urdu text using multilingual BERT," *Sci. Rep.*, vol. 12, no. 1, p. 5436, Mar. 2022.

[13] M. Z. Asghar, A. Sattar, A. Khan, A. Ali, F. M. Kundi, and S. Ahmad, "Creating sentiment lexicon for sentiment analysis in Urdu: The case of a resource-poor language," *Expert Syst.*, vol. 36, no. 3, Jun. 2019, Art. no. e12397.

[14] N. Mukhtar and M. A. Khan, "Effective lexicon-based approach for Urdu sentiment analysis," *Artif. Intell. Rev.*, vol. 53, no. 4, pp. 2521–2548, Apr. 2020.

[15] N. Mukhtar, M. A. Khan, and N. Chiragh, "Lexicon-based approach outperforms supervised machine learning approach for Urdu sentiment analysis in multiple domains," *Telematics Informat.*, vol. 35, no. 8, pp. 2173–2183, Dec. 2018.

[16] K. Amjad, M. Ishtiaq, S. Firdous, and M. A. Mehmood, "Exploring Twitter news biases using Urdu-based sentiment lexicon," in *Proc. Int. Conf. Open Source Syst. Technol. (ICOSST)*, Dec. 2017, pp. 48–53.

[17] N. Mukhtar, M. A. Khan, and N. Chiragh, "Effective use of evaluation measures for the validation of best classifier in Urdu sentiment analysis," *Cognit. Comput.*, vol. 9, no. 4, pp. 446–456, Aug. 2017.

[18] N. Mukhtar and M. A. Khan, "Urdu sentiment analysis using supervised machine learning approach," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 2, Feb. 2018, Art. no. 1851001.

[19] I. Safder, Z. Mahmood, R. Sarwar, S. Hassan, F. Zaman, R. M. A. Nawab, F. Bukhari, R. A. Abbasi, S. Alelyani, N. R. Aljohani, and R. Nawaz, "Sentiment analysis for Urdu online reviews using deep learning models," *Expert Syst.*, vol. 38, no. 8, Dec. 2021, Art. no. e12751.

[20] L. Khan, A. Amjad, N. Ashraf, H.-T. Chang, and A. Gelbukh, "Urdu sentiment analysis with deep learning methods," *IEEE Access*, vol. 9, pp. 97803–97812, 2021.

[21] U. Naqvi, A. Majid, and S. A. Abbas, "UTSA: Urdu text sentiment analysis using deep learning methods," *IEEE Access*, vol. 9, pp. 114085–114094, 2021.

[22] L. Sana, K. Nasir, A. Urooj, Z. Ishaq, and I. A. Hameed, "BERS: Bussiness-related emotion recognition system in Urdu language using machine learning," in *Proc. 5th Int. Conf. Behav., Econ., Socio-Cultural Comput. (BESC)*, Nov. 2018, pp. 238–242.

[23] W. Ahmad and M. Edalati, "Urdu speech and text based sentiment analyzer," 2022, *arXiv:2207.09163*.

[24] F. Hashim and M. Khan, "Sentence level sentiment analysis using Urdu nouns," in *Proc. Conf. Lang. Technol.*, vol. 2016, 2016, pp. 101–108.

[25] E. Riloff, A. Qadir, P. Surve, L. D. Silva, N. Gilbert, and R. Huang, "Sarcasm as contrast between a positive sentiment and negative situation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2013, pp. 704–714.

[26] E. Camp, "Sarcasm, pretense, and the semantics/pragmatics distinction," *Noûs*, vol. 46, no. 4, pp. 587–634, Dec. 2012.

[27] S. K. Bharti, K. S. Babu, and S. K. Jena, "Parsing-based sarcasm sentiment recognition in Twitter data," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2015, pp. 1373–1380.

[28] B. Liu, "Sentiment analysis and opinion mining," *Synth. Lectures Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, May 2012.

[29] Y. Du, T. Li, M. S. Pathan, H. K. Teklehaimanot, and Z. Yang, "An effective sarcasm detection approach based on sentimental context and individual expression habits," *Cognit. Comput.*, vol. 14, no. 1, pp. 78–90, Jan. 2022.

[30] R. S. Jagdale, V. S. Shirsat, and S. N. Deshmukh, "Sentiment analysis on product reviews using machine learning techniques," in *Proc. Cogn. Inform. Soft Comput. (CISC)*. Singapore: Springer, 2019, pp. 639–647.

[31] M. Rathi, A. Malik, D. Varshney, R. Sharma, and S. Mendiratta, "Sentiment analysis of tweets using machine learning approach," in *Proc. 11th Int. Conf. Contemp. Comput. (IC3)*, Aug. 2018, pp. 1–3.

[32] A. Naresh and P. V. Krishna, "An efficient approach for sentiment analysis using machine learning algorithm," *Evol. Intell.*, vol. 14, no. 2, pp. 725–731, Jun. 2021.

[33] A. Onan, "Sarcasm identification on Twitter: A machine learning approach," in *Artificial Intelligence Trends in Intelligent Systems*. Cham, Switzerland: Springer, 2017, pp. 374–383.

[34] R. González-Ibánez, S. Muresan, and N. Wacholder, "Identifying sarcasm in Twitter: A closer look," in *Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2011, pp. 581–586.
[35] H. F. Nweke, Y. W. Teh, M. A. Al-Garadi, and U. R. Alo, "Deep learning algorithms for human activity recognition using mobile and wearable sensor networks: State of the art and research challenges," *Expert Syst. Appl.*, vol. 105, pp. 233–261, Sep. 2018.
[36] D. Ghosh, A. R. Fabbri, and S. Muresan, "The role of conversation context for sarcasm detection in online interactions," 2017, *arXiv:1707.06226*.
[37] M. Zhang, Y. Zhang, and G. Fu, "Tweet sarcasm detection using deep neural network," in *Proc. COLING 26th Int. Conf. Comput. Linguistics Conf., Tech. Papers*, Osaka, Japan, 2016, pp. 2449–2460.
[38] A. Ghosh and D. T. Veale, "Fracking sarcasm using neural network," in *Proc. 7th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal.*, 2016, pp. 161–169.
[39] S. Poria, E. Cambria, D. Hazarika, and P. Vij, "A deeper look into sarcastic tweets using deep convolutional neural networks," 2016, *arXiv:1610.08815*.
[40] S. Amir, B. C. Wallace, H. Lyu, and P. C. M. J. Silva, "Modelling context with user embeddings for sarcasm detection in social media," 2016, *arXiv:1607.00976*.
[41] D. Hazarika, S. Poria, S. Gorantla, E. Cambria, R. Zimmermann, and R. Mihalcea, "CASCADE: Contextual sarcasm detection in online discussion forums," 2018, *arXiv:1805.06413*.
[42] J. Plepi and L. Flek, "Perceived and intended sarcasm detection with graph attention networks," 2021, *arXiv:2110.04001*.
[43] (2023). *Urdu Sarcastic Tweets Dataset*. [Online]. Available: https://www.kaggle.com/ds/2888930
[44] L. M. Cendani, R. Kusumaningrum, and S. N. Endah, "Aspect-based sentiment analysis of indonesian-language hotel reviews using long short-term memory with an attention mechanism," in *Proc. Int. Conf. Netw., Intell. Syst. Secur.* Cham, Switzerland: Springer, 2022, pp. 106–122.
[45] M. S. Rani and S. Subramanian, "Attention mechanism with gated recurrent unit using convolutional neural network for aspect level opinion mining," *Arabian J. Sci. Eng.*, vol. 45, no. 8, pp. 6157–6169, Aug. 2020.
[46] J. S. Vimali and S. Murugan, "A text based sentiment analysis model using bi-directional LSTM networks," in *Proc. 6th Int. Conf. Commun. Electron. Syst. (ICCES)*, Jul. 2021, pp. 1652–1658.
[47] A. U. Rehman, A. K. Malik, B. Raza, and W. Ali, "A hybrid CNN-LSTM model for improving accuracy of movie reviews sentiment analysis," *Multimedia Tools Appl.*, vol. 78, no. 18, pp. 26597–26613, Sep. 2019.
[48] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, "Recent advances in recurrent neural networks," 2017, *arXiv:1801.01078*.
[49] R. Pandey, A. Kumar, J. P. Singh, and S. Tripathi, "Hybrid attention-based long short-term memory network for sarcasm identification," *Appl. Soft Comput.*, vol. 106, Jul. 2021, Art. no. 107348.
[50] C. C. Aggarwal, *Neural Networks and Deep Learning*, vol. 10, no. 978. Cham, Switzerland: Springer, 2018, p. 3.
[51] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," 2014, *arXiv:1406.1078*.
[52] S. Sachin, A. Tripathi, N. Mahajan, S. Aggarwal, and P. Nagrath, "Sentiment analysis using gated recurrent neural networks," *Social Netw. Comput. Sci.*, vol. 1, no. 2, pp. 1–13, Mar. 2020.
[53] G. Long, D. Lin, J. Lei, Z. Guo, Y. Hu, and L. Xia, "A method of machine learning for social bot detection combined with sentiment analysis," in *Proc. 5th Int. Conf. Mach. Learn. Natural Lang. Process.*, Dec. 2022, pp. 239–244.
[54] A. Kumar, V. T. Narapareddy, V. A. Srikanth, A. Malapati, and L. B. M. Neti, "Sarcasm detection using multi-head attention based bidirectional LSTM," *IEEE Access*, vol. 8, pp. 6388–6397, 2020.
[55] J. Wang, L.-C. Yu, K. R. Lai, and X. Zhang, "Dimensional sentiment analysis using a regional CNN-LSTM model," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics (Short Papers)*, vol. 2, 2016, pp. 225–230.
[56] A. Hassan and A. Mahmood, "Deep learning approach for sentiment analysis of short texts," in *Proc. 3rd Int. Conf. Control, Autom. Robot. (ICCAR)*, Apr. 2017, pp. 705–710.
[57] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[58] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[59] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1532–1543.
[60] T. Pires, E. Schlinger, and D. Garrette, "How multilingual is multilingual BERT?" 2019, *arXiv:1906.01502*.
[61] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 19–27.
[62] J. Cohen, "A coefficient of agreement for nominal scales," *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, Apr. 1960.
[63] I. Guyon and A. Elisseeff, "An introduction to feature extraction," in *Feature Extraction*. Cham, Switzerland: Springer, 2006, pp. 1–25.
[64] (2023). *Urduhack: A Python NLP Library for Urdu Language*. Accessed: Jun. 20, 2023. [Online]. Available: https://pypi.org/project/urduhack/
[65] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, Dec. 2017.
[66] A. Mishra, K. Dey, and P. Bhattacharyya, "Learning cognitive features from gaze data for sentiment and sarcasm classification using convolutional neural network," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics (Long Papers)*, vol. 1, 2017, pp. 377–387.
[67] 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, no. 1, pp. 1–13, Dec. 2020.

MUHAMMAD EHTISHAM HASSAN received the M.S. degree from NUST-EME, Rawalpindi, Pakistan. He is currently the capacity of a Doctoral Researcher with the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology (GIKI), Pakistan, and is part of the Data Engineering Management Analysis Group (DEMA) operating under the Faculty of Data Science. He was a Coordinator of ICPC Asia-West Topi for the last four years. His role as a Coordinator, of ICPC is to maintain operational facilities and regulations in the ICPC Asia-West contests. His research interests include natural language processing, machine learning, deep learning, large language models (LLMs), and data science.

MASROOR HUSSAIN is currently a Professor in data science and the Head of the Department of Computer Engineering and Data Science, Faculty of Computer Science and Engineering, GIKI. His research interests include adaptive meshing, data mining, data warehousing, finite element methods, neural networks, and parallel computing. He had participated in many national programming and software competitions, as the Representative of NUCES-FAST Lahore, like Softec, Procom, and Softcom during his undergraduate studies. He also won the first prize in SoftCom, in November 2001, for the speed programming competition. Currently, he is also the Regional Contest Director (RCD), the International Collegiate Programming Contest (ICPC), Asia Topi Region. He is also the Project Director, has expertise in the areas of high-performance computing and deep neural networks. He has supervised project(s) in the areas of sentiment analysis and fake news detection which will significantly influence the project. He possesses vast experience in parallel programming and has supervised M.S. and Ph.D. students in this area. Since managing big data will be a core area of this project, his expertise and experience in invaluable in all aspects of the project.

IFFAT MAAB received the Ph.D. degree from the Department of Technology Management for Innovation, The University of Tokyo. She was a Researcher with the Institute for Natural Language Processing, Universität Stuttgart, Germany, for on a series of large language models, with a primary focus on evaluating their performance in detecting misinformation within political news. She is associated with ICPC, Asia-West Region, and Topi as an Associate Director. She has worked at Toshiba Corporation, Japan, on Visual Question Answering (VQA) to develop models capable of answering questions based on visual content using specialized NLP libraries and computer vision tools. She has also experience at Digimart Company Limited and Crystal Method Company.

MUHAMMAD ATTIQUE KHAN (Senior Member, IEEE) received the master's and Ph.D. degrees in human activity recognition for application of video surveillance and skin lesion classification using deep learning from COMSATS University Islamabad, Pakistan. He is currently a Lecturer with the Computer Science Department, HITEC University, Taxila, Pakistan, and an Adjunct Professor with the Department of Computer Science and Mathematics, Lebanese American University, Lebanon. He has above 190 publications that have more than 6500 citations and an impact factor of more than 600 with an H-index of 50. His research interests include medical imaging, COVID-19, MRI analysis, video surveillance, human gait recognition, and agriculture plants. He is a Reviewer of several reputed journals, such as IEEE Transactions on Industrial Informatics, IEEE Transactions on Neural Networks, *Pattern Recognition Letters*, *Multimedia Tools and Application*, *Computers and Electronics in Agriculture*, *IET Image Processing*, *Biomedical Signal Processing and Control*, *IET Computer Vision*, *EURASIP Journal of Image and Video Processing*, IEEE Access, *Sensors* (MDPI), *Electronics* (MDPI), *Applied Sciences* (MDPI), *Diagnostics* (MDPI), and *Cancers* (MDPI).

USMAN HABIB (Senior Member, IEEE) received the master's degree from Norwegian University of Science and Technology (NTNU), Norway, and the Ph.D. degree from the ICT Department, Technical University of Vienna, Austria. He is currently an Associate Professor and the Head of the Software Engineering Department, FAST School of Computing, National University of Computer & Emerging Sciences (NUCES), Islamabad, Pakistan. Before joining FAST-NUCES, he was with the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology (GIKI), Swabi, and COMSATS University Islamabad, Abbottabad Campus. With over 18 years of teaching and research experience, since 2006, he has successfully completed various industrial projects along with serving in academia. He also actively engages in research and has authored numerous conferences and journal publications. His current research interests include machine learning, data analytics, pattern recognition, security, and medical image processing.

ANUM MASOOD received the B.Sc. and M.Sc. degrees in computer science from COMSATS University Islamabad, Islamabad, Pakistan, in 2012 and 2014, respectively, and the Ph.D. degree in computer science and engineering from Shanghai Jiao Tong University, Shanghai, China, in 2019. She was a Lecturer with the Department of Computer Science, COMSATS University Islamabad, from 2014 to 2020. She is currently a Postdoctoral Researcher with Norwegian University of Science and Technology, Trondheim, Norway, and affiliated with the PET Centre, St. Olav's Hospital, Trondheim, Norway. She was a Visiting Researcher with the Institute of Neuroscience and Medicine, Forschungszentrum Jlich, Institute for Cardiogenetics, University of Luebeck, Germany, and Liverpool John Moores University, U.K. Her research interests include medical image analysis, automated cancer detection, machine learning, and image processing.