IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 May 2024, accepted 12 July 2024, date of publication 18 July 2024, date of current version 26 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3429496

RESEARCH ARTICLE

Revolutionizing Urdu Sentiment Analysis: Harnessing the Power of XLM-R and GPT-2

MUHAMMAD REHAN ASHRAF[1,2], MUZAMMAL HUSSAIN[3], M. ARFAN JAFFAR[1,4], WAHEED YOUSUF RAMAY[5], AND MUHAMMAD FAHEEM[6], (Member, IEEE)

[1]Faculty of Computer Science and Information Technology, Superior University, Lahore 54600, Pakistan
[2]Department of Computer Sciences, COMSATS University Islamabad, Vehari 61000, Pakistan
[3]Department of Computer Science, Government College University Faisalabad, Sahiwal 57000, Pakistan
[4]Intelligent Data Visual Computing Research (IDVCR), Lahore 54600, Pakistan
[5]Department of Computer Science, Cholistan University of Veterinary and Animal Sciences, Bahawalpur 63100, Pakistan
[6]School of Technology and Innovations, University of Vaasa, 65200 Vaasa, Finland

Corresponding author: Muhammad Faheem (muhammad.faheem@uwasa.fi)

The authors would like to thank their affiliated universities for providing research funding to complete this research work.

ABSTRACT Sentiment analysis extracts valuable insights from textual sources using computation, textual or systematic analysis, and natural language processing. It identifies and measures the attitudes, beliefs, and emotional states individuals express through text data. Recent research on sentiment analysis has largely focused on the English language; therefore, low-resource languages are getting much less attention. Conducting sentiment analysis of low-resource languages is difficult because large datasets and related repositories are unavailable. This paper creates a new dataset for low-resource language (Urdu) to address this issue. The dataset, namely LUCSA-23, consists of more than 65,000 user reviews from various genres, including food, sports, showbiz, apps, and political reviews from developing countries, i.e., Pakistan. Urdu domain experts further annotate the created dataset. This paper proposes an Urdu sentiment analysis approach leveraging the transformer model, i.e., XLM-R and GPT-2. It preprocesses the Urdu text input, generates BERT embeddings, and passes them to the proposed classifier as input for sentiment classification. The proposed classifier is compared with machine/deep/embedded classifiers to evaluate its performance. The findings show that the proposed classifiers outperform existing state-of-the-art approaches with an accuracy of 95%.

INDEX TERMS Sentiment analysis, Urdu, XLM-R, GPT-2, classification, deep learning, BERT.

I. INTRODUCTION

In recent years, many social communication platforms, i.e., blogs, forums, Facebook, YouTube, Twitter, and Instagram, have gained popularity among users. These social networks play an important role in how individuals connect [1], [2]. According to Datareportal [3] by the beginning of 2023, there will be 5.16 billion people utilizing the internet all over the globe, equivalent to 64.4% of the planet's total population. As a result of technological advances and rising levels of awareness, more and more people are turning to the Internet for various purposes, including global communication, online commerce, exchanging information and opinions, long-distance education, and correspondence [4], [5].

The World Wide Web (WWW) has made social network conversations feasible for a single user. Consequently, textual communication, or more precisely sentiment analysis, has become essential to comprehend people's behavior [6], [7], [8], [9], [10]. In today's data-driven era, sentiment analysis is an incredibly prominent task [11]. Businesses and organizations can acquire significant insights into the attitudes, views, and feelings of their customers, stakeholders, and the general public by exploiting Natural Language Processing (NLP) and text analysis tools [12]. By mining subjective

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

information from textual data, they may comprehend their audience by making the right decisions [13]. The English language and certain European languages are regarded as highly technological and linguistically advanced [14], but many others, i.e., Bengali, Hindi, Persian, and Urdu, are categorized as low-resource languages due to their lack of digital resources [15]. The absence of linguistic resources, i.e., standardized datasets and advanced analytical tools, is the main reason behind the resource poorness of the Urdu language [16].

Urdu, commonly called Standard Urdu (Mayari Urdu), is an Indo-Aryan language widely spoken in South Asian regions. It is acknowledged as an official language alongside English in Pakistan, serving as the country's lingua franca [17]. Urdu is regarded as an Eighth Schedule language in India. Due to its distinctive characteristics, the Urdu language presents several challenges for language processing. For instance, Urdu has both informal and formal verb tenses, and every word may be either masculine or feminine. The use of loan words from Persian, Arabic, and Sanskrit further complicates the analysis of Urdu. The right-to-left writing of the Urdu language makes it difficult to distinguish between phrases, e.g., اُدھرکیاپڑاھواھے؟ (What was there?), is unclear and seems to be no spaces between the words. The scarcity of reliable lexical sources [17] and the lack of datasets of the Urdu language, mostly due to morphological complexity [18], provide substantial obstacles to effectively interpreting the sentiments expressed in Urdu. Making a high-quality machine-readable corpus is more difficult since Urdu websites use illustrations rather than the conventional way for text encodings. The well-known sentiment lexicon database is essential to developing a sentiment analysis classifier in the Urdu dialect. Several lexicons are available for the English language, including SentiWordNet, e.g., [20] and AFINN [19]. However, Urdu is a resource-poor language with severely constrained lexicons and sentimental datasets. Implementing a fully effective sentiment analysis model for Urdu is hindered by challenges in segmenting Urdu words, assessing their morphological structure, and vocabulary variations, among other issues. The state-of-the-art pre-trained models, i.e., BERT, have recently shown unmatched performance in tasks including sentiment analysis [17], [18]. These models are trained on many datasets to capture more durable semantic relationships.

This paper introduces a novel dataset, LUCSA-23, tailored specifically for the Urdu language to tackle prevalent issues. Comprising over 65,000 user reviews across diverse domains such as food, sports, showbiz, apps, and political critiques from developing regions, e.g., Pakistan, this dataset offers a comprehensive resource. Moreover, to enhance its utility, the dataset undergoes meticulous annotation by Urdu domain experts. We also propose an Urdu sentiment analysis approach that harnesses the power of transformer models, specifically XLM-R and GPT-2. The proposed approach involves preprocessing Urdu text inputs, generating BERT embeddings, and feeding them into our proposed classifier for sentiment analysis. We benchmark the proposed classifier against various machine, deep, and embedded classifiers to assess its effectiveness. The findings demonstrate that the proposed classifier significantly outperforms existing state-of-the-art approaches, achieving an impressive accuracy rate of 95%.

The main contributions of the paper are as follows:

- The Large Urdu Corpus for Sentiment Analysis (LUCSA-23) is a pioneering multi-class sentiment corpus comprising over 65,000 user reviews categorized into positive, negative, and neutral sentiments. These reviews span across a diverse spectrum of genres, encompassing domains such as food, sports, showbiz, apps, and politics. No such comprehensive dataset exists, making LUCSA-23 a valuable resource for sentiment analysis tasks in Urdu language processing.
- An Urdu sentiment analysis approach is proposed that harnesses the power of transformer models by implementing the modified GPT2 model named Urdu GPT2 (UGPT2) for multi-class Urdu sentiment analysis.

The remaining sections of this paper are organized as follows: Related work on Urdu sentiment analysis and previous datasets are covered in Section II. The dataset generation process is discussed in Section III, while the proposed methodology for sentiment analysis is explained in Section IV. Section V covers the experimental setting and the analysis of results. Finally, Section VII concludes the paper by summarizing the findings and providing directions for future research.

II. LITERATURE REVIEW

Over the past decade, text classification has more focused on sentiment analysis. The section below provides an overview of the work conducted in Urdu sentiment analysis. It highlights the different approaches utilized for sentiment analysis of Urdu text, i.e., machine/deep learning Urdu text classification. Researchers have explored these approaches to understand and analyze sentiment in Urdu text effectively.

A. MACHINE LEARNING BASED APPROACHES

Arif et al. [21] investigated Roman Urdu data and delivered the findings using several classifiers on the set. Some characteristics were used in conjunction with machine learning techniques for binary classification. The efficiency of various classifiers was measured by applying them to a sparse matrix. The highest accuracy by using machine learning classifiers was achieved with the help of the TF-IDF term weighting model. When compared to other classifiers, SVM's 96% accuracy was the highest. The authors used Lxa-pipes to conduct tokenization, POS tagging, and Lemmatization on two datasets written in Catalan and Basque, two languages with few resources available for sentiment analysis at the aspect level. For the purpose of labeling the positive or negative tone of expressions of opinion, a linear SVC classifier was trained. They trained a

CRF on the common characteristics used to extract opinion leaders, targets, and expressions. F1 is utilized for data extraction and classification, while 10-fold cross-validation was performed for assessment on 80% of the data [22].

Nawaz et al. [23] elucidated that the sentiment has been isolated using a deductive method. They unveiled a method with two distinct stages. In the first stage, they used Normalized Google Distance to extract and categorize phrases relevant to their target and goal (NGD) and a customized POS tagger to identify the various aspects. In the second stage, they used Concept Net to eliminate unnecessary details. Each word in an opinionated phrase has been analyzed using this method to determine whether or not it is an aspect word. The authors participated in SemEval Task 4 2014 [24]. They employed a machine learning technique for the confined system. They used LDA, semantic spaces, and semantic dictionaries to enlarge the constrained feature set while unconstrained. After that, they compared their findings to the top, average and SemEval baseline so that their system performs well.

The authors [25], [26] developed a Hindi-English aggressiveness annotation corpus. They have established an annotation approach by categorizing aggressiveness tags into mild, moderate, and severe categories (top-level). Aggressive, passive, and passively aggressive. Each superordinate level had two additional features: discursive role and impact. Within these were ten specific discursive impacts, each rooted in varying levels of aggressiveness. Annotation involved using three parent tags and ten child tags in a hierarchical arrangement. Prasad et al. [27] employed computations to demonstrate that Hindi and Urdu have the same grammar but distinct dictionaries.

B. DEEP LEARNING APPROACHES

Recently, researchers have explored deep learning methodologies for classifying Urdu text. Akhter et al. [28] utilized deep learning techniques to categorize Urdu comments utilized in the assembly process. They observed an enhancement in accuracy for small and medium-sized datasets by eliminating infrequent words and stop words, although this improvement was compromised for larger datasets. Their findings indicated that CNNs with three or more filters outperformed LSTM and CLSTM models. Furthermore, the authors found that a single-layer CNN with diverse filters outperformed baseline methods for document-level text categorization. Asim et al. [29] evaluated the effectiveness of various machine learning (ML), deep learning (DL), and hybrid models for document classification. Their study revealed that employing a feature selection technique based on normalized difference measures improved overall model performance. When examining the emotional tone of Roman Urdu text, their DL model, particularly LSTM, surpassed baseline ML approaches. DL techniques, utilizing word embeddings as inputs, capture semantic information and uncover relationships between different sentence parts, eliminating the need for additional rules or features. Embedding models such as FastText facilitate word encoding, while pre-trained embeddings like word2vec enable transfer learning, especially for languages lacking such resources. Given the limited exploration of DL methods for Urdu text, the researchers opted for them due to their effectiveness in sentiment categorization. They utilized embeddings for the Urdu language, i.e FastText, Urdu CoNLL17 [30], and Samar [30].

Kim et al. [31] implemented a one-layer convolutional neural network (CNN) utilizing pre-trained word vectors sourced from 100 billion words in Google News. Their model underperformed across various benchmarks without pre-trained word embeddings, but it demonstrated significant improvement with their inclusion. Meanwhile, a multimodal sentiment classifier achieved an accuracy of up to 82.5% through a supervised fuzzy rule-based framework.

Zhu et al. [33] proposed a model combining LSTM and quantum computing, exploring two distinct datasets: MELD and IEMOCAP. Additionally, they presented a technique for multimodal sentiment analysis (MSA) employing a fusion network incorporating data from multiple sources [34], like CMU-MOSI, MOSEI, and YouTube, resulting in a 2.9% increase in accuracy.

Agarwal et al. [35] introduced a deep learning-based MSA model, assessing various recurrent neural network (RNN) variations, including GRNN, LRNN, GLRNN, and UGRNN. They also developed a method for multimodal emotion classification leveraging transfer learning and a transformer modal architecture, alongside unveiling the MORSE dataset for MSA. Yao et al. [36] compared and evaluated numerous MSA strategies, highlighting the significant advancements in English while acknowledging the lag in other languages. Other researchers [37]] proposed MSA method for 3D Residual Networks in Embedded Systems. They experimented on the MOSI dataset, and their F1 score ended up being 80%.

Shakeel et al. [39] presented a model using LSTM. They concluded that the hierarchical clustering algorithm they described was the best option for classifying users into the resulting adaptive tree. Other authors [40] introduced a Recurrent Neural Network (RNN) with Deep Convolutions and Attention (ABCDM). ABCDM employs two bidirectional LSTM and GRU layers to evaluate temporal information flow, which was then used to extract past and future contexts. Additionally, attention techniques were used to highlight certain phrases in the outputs of the ABCDM's bi-directional layers. To minimize the dimensions of features and identify local properties that are not position-dependent, ABCDM combines convolution and pooling algorithms. The most common and crucial task in sentiment classification was identifying sentiment polarity.

Feature-based supervised algorithms, i.e., SVM and MaxEnt, were inferior to methods like bi-LSTM, CNN, and LSTM. Bi-LSTM architecture with multi-layer self-attention was touted to be the state-of-the-art approach right now [41]. A new benchmark with an accuracy of 59.50 percent was set.

TABLE 1. Genres and Sources for LUCSA-23 Creation.

Genres	Sources
Appliances and Tools	Urdupoint Bazauq
Showbiz	ARY, Youtube, Fashionuniverse, Tafrehmela, Urdupoint
Sports and Entertainment	ESPNcricinfo Urdu, Urdupoint Sports, Youtube Comments, Blogs
Political Reviews	News1, GeoNews Urdu, Urdupoint, Youtube Comments, Siasat Blogs
Food	Urdupoint, Friendsconor, Youtube Comments, Foodpoint
Software and Mobile Applications	Techjuice, Urduplanet, Filepuma

In [42] evaluated the efficacy of various classification models, characteristics, i.e., BERT and SVMs were compared. BERT-based monolingual models trained on the target language data outperform state-of-the-art models by 4% and 5% in terms of Jaccard score for Arabic and Spanish, respectively. The BERT models achieved an accuracy of 90% for Arabic and 80% for Spanish, showcasing their effectiveness in multilingual sentiment analysis tasks. These findings highlighted the potential of utilizing BERT-based models for sentiment analysis in Arabic and Spanish, demonstrating their superior performance compared to other approaches in this context. An attention-based Bidirectional CNN-RNN emerges as a robust deep learning solution addressing large feature dimensionality and weighting challenges. To achieve its cutting-edge performance, the model leverages bidirectional contexts, position-invariant local data, and pooling algorithms for sentiment polarity identification [43].

Gan et al. [44] employed a sequence tagging approach utilizing conditional random fields (CRFs) and bidirectional gated recurrent units (BiGRUs). This method effectively extracts information from various aspects and integrates it with Glove embeddings to enrich aspect-level sentiment analysis (ALSA) models. Using ML and DL classifiers, an effort was made to conduct cross-domain S.A in Urdu [45]. Furthermore, they endeavored to conduct cross-domain sentiment analysis in Urdu by employing both traditional machine learning (ML) and deep learning (DL) classifiers. Building upon this effort, researchers analyzed the landscape of Urdu emotion detection to assess its current status and potential future directions [46]. They created a specialized dataset tailored for characteristic-based sentiment analysis to facilitate this analysis, allowing for comprehensive investigations into emotional nuances in Urdu text [47].In parallel, studies explored the cognitive strategies involved in sentiment analysis, particularly in identifying sarcasm and its implications [48]. Researchers also embarked on a multistep process involving the extraction of English sentences, translation into Urdu, grammatical correction using natural language processing (NLP), and subsequent sentiment analysis using ML techniques [49]. Using ML methods, they could extract English sentences, translate them into Urdu, fix grammatical mistakes, and analyze their sentiment [50]. Expanding the scope, Urdu and English tweets and news data were gathered and analyzed using ML algorithms, shedding light on perspectives related to the dengue epidemic [51]. Additionally, leveraging machine learning and deep learning methods, researchers constructed an annotated corpus for emotion identification using the Urdu Nastalique Emotions Dataset (UNED) [52]. Emotion classification was performed using machine learning on a large-scale Urdu dataset of labels [53]. Leveraging Bidirectional Encoder Representations from Transformers (BERT), intent detection was performed in Urdu following data extraction [54]. To address the challenge of identifying dangerous text in Urdu, researchers devised a stacking model combining Naive Bayes for learning and Logistic Regression for meta-learning, showcasing superior performance compared to previous methods [55], [56].

Vyas et al. [57] proposed an automated system to extract positive, negative, and neutral sentiments from tweets and classify them using machine-learning (ML) models. Their hybrid method uses lexicon-based sentiment analysis and supervised ML for tweet classification. Using precision, accuracy, recall, and F1 score, the framework found that most sentiments were positive (38.5%) or neutral (34.7%). Long short-term memory (LSTM) neural network-based technique achieved 83% accuracy.

Safder et al. [58] proposed an Urdu deep learning sentiment analysis model and an open-source corpus of 10,008 reviews from 566 online forums on sports, food, software, politics, and entertainment. The study aims to generate a human-annotated Urdu corpus for sentiment analysis research and to evaluate current models utilizing it. The study compares LSTM, RCNN, Rule-Based, N-gram, SVM, CNN, and LSTM. The RCNN model outperforms others with 84.98% binary and 68.56% ternary classification accuracy.

Altaf et al. [59] utilized linguistic characteristics of the Urdu language to do sentiment analysis at the sentence level. Additionally, it employs standard machine-learning techniques to classify idioms and proverbs. We create a dataset that includes idioms, proverbs, and sentences from the news domain. After carefully analyzing the Urdu language, we extract features from the dataset based on part-of-speech tags, boolean values, and numeric values. The experimental findings demonstrate that the J48 classifier has the highest performance in sentiment classification, with an accuracy of 90% and an F-measure of 88%.

Overall, these endeavors represent a concerted effort to advance sentiment analysis capabilities in Urdu, spanning various methodologies and datasets to unravel the complexities of emotional expression in the language.

III. LUCSA-23 DATASET CREATION

It is noteworthy that five domain experts enrolled in PhD-Urdu meticulously annotated user reviews over a period exceeding a year. Criteria and standards for the annotation process are established and presented in Section III-A, and the corpus is annotated manually in accordance with these guidelines. The annotated data is compiled in a Google spreadsheet, documenting pertinent information such as Annotator ID#, Phrase, Label, and Domain. The entire dataset's Inter-Annotator Agreement (IAA) is calculated at 0.72 using Fleiss' kappa method. This high agreement, coupled with consistent average scores, underscores the professionals' understanding of and adherence to the manual annotation rules throughout the process. The researchers' overarching objective is to expand the dataset further. Most publicly available annotated datasets, in contrast to LCUSA-23, are considerably smaller and encompass phrases from a limited number of classes. Notably, existing datasets typically only encompass negative and positive classes.

A. ANNOTATION GUIDELINES

- Sentences containing derogatory slang terms, such as میٹر گھوم جانا, are classified as negative.
- A sentence qualifies as positive if it incorporates at least one slang term, as exemplified by کرو Chill.
- For a statement to be categorized as positive, it must convey a positive sentiment, exemplified by اچھا جواب تھا یہ, devoid of any negativity across its key features.
- If a reader is likely to agree with the assertions presented in the sentence, it is deemed positive, as illustrated by مجھے آپ کے جواب میں کہنا مطفیق ہوں۔.
- Reviews are designated negative if the language consistently reflects a negative attitude.
- When a user remark contains more insults than compliments, it is automatically classified as negative.
- Negative sentences must contain unequivocal criticism to be considered negative.
- Sentences including the words "no," "not," or "never" are identified as negative.
- Reviews commencing with a negative statement and concluding with a positive term are considered negative.
- Statements that embarrass the subject are classified as negative.
- Sentences stating facts are categorized as neutral.
- Statements expressing theories, beliefs, or opinions are classified as neutral.

IV. PROPOSED METHODOLOGY

The experimental details are discussed in this section, in which numerous machine learning and deep learning models are used. We implemented machine learning models (RF, LR, SVM, XGBoost, and DT) and deep learning models (LSTM, CNN1D+LSTM). Additionally, we fine-tuned state-of-the-art transformer models (GPT2 and XLM-R) for sentiment analysis of Urdu text. Additionally, the proposed LUCSA-23 corpus is the foundation for analyzing these models. The overall architecture of the proposed methodology is shown in Fig. 1.

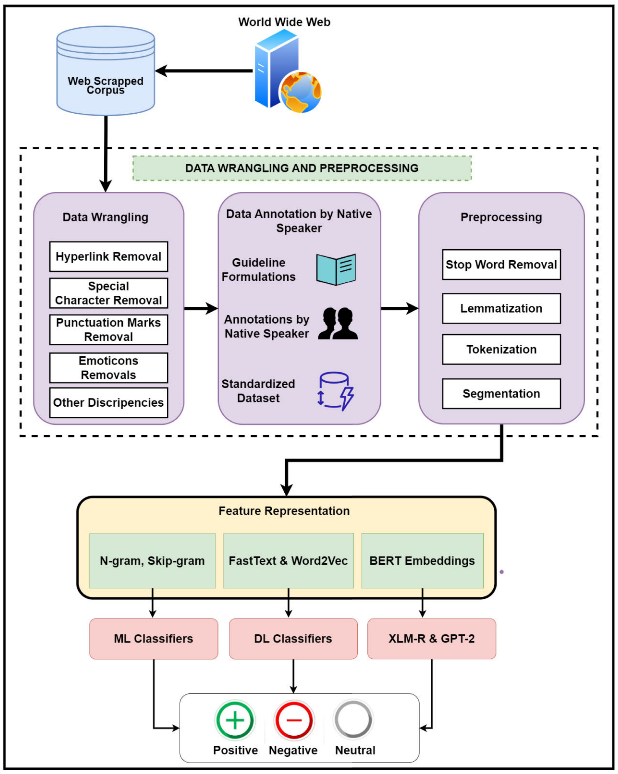

FIGURE 1. Overview of the Proposed Methodology.

A. NORMALIZATION

The problem of inaccurate text encoding is addressed using the normalization technique. Mapping Arabic and Urdu characters to the right Unicode characters is essential for accurately representing text for these two languages. Each Urdu character is mapped to its corresponding Unicode character in the hexadecimal range during normalization to provide consistent text representation and prevent encoding problems. Furthermore, normalization is used to avoid concatenating various Urdu words.

B. STOP-WORDS REMOVAL

Words that are used to complete the sentences are called stop words. The words like تھا and ہم، تم are often used in the Urdu dialect as stop-words. Despite this, it is difficult to eliminate stop-words automatically in Urdu because of the structure of the language dialect and the lack of resources. A list of mostly used Urdu stop-words is created, and those words are

TABLE 2. Few Examples from LUCSA-23.

Sentiment	Sentences/Comments
Positive	میں اس سافٹ ویئر کے استعمال سے بہت خوش ہوں۔
	I am very happy with the use of this software.
	میں اس پارٹی کی حمایت کرتا ہوں کیونکہ اس نے ملک کے لئے بہت کچھ کیا ہے۔
	I support this political party because it has done a lot for the country.
	میں کرکٹ کے خیالات سے بہت متاثر ہوں۔
	I am very impressed with the thoughts of cricket.
Negative	یہ سافٹ ویئر بہت آہستہ ہے اور کام بھی نہیں کر رہا۔
	This software is very slow and not working properly.
	میں سیاست سے نفرت کرتا ہوں کیونکہ یہاں کے لوگوں کو صرف اپنی مفاد کے لئے استعمال کیا جاتا ہے۔
	I hate politics because people here are only used for their own benefit.
	میں ٹینس کھیلنے سے بہت نفرت کرتا ہوں۔
	I hate playing tennis.
Positive	میں نے اس سافٹ ویئر کو استعمال نہیں کیا۔
	I haven't used this software.
	میں سیاست کے بارے میں کچھ نہیں جانتا۔
	I don't know anything about politics.
	میں اس کھیل میں دلچسپی نہیں رکھتا۔
	I am not interested in this sport.

TABLE 3. Statistics of LUCSA-23.

Features	Count
Total Sentences/Comments	65670
Total Positive	23657
Total Negative	20453
Total Neutral	21560
Minimum Words in a Sentence	3
Maximum Words in a Sentence	121
Total Tokens of Dataset	845710
Average Tokens in a Sentence	14

removed from the file using UNLT and UrduHack[1] Python libraries.

C. LEMMATIZATION

Lemmatization reduces words to their root forms to perform sentiment analysis on Urdu text. It contributes to standardizing words, eliminating inflectional variances, and enhancing correctness. For example, the word “"محبتیں"” (loves) is lemmatized to “"محبت"” (love), and similarly “"تلواریں "” (swords) into “"تلوار"” (sword). Lemmatization enables consistent analysis by preserving the core meaning of a word, which helps in determining the correct sentiment of the text. We exploit the UrduHack Library for this task.

D. SEGMENTATION

Segmentation is used to ascertain and identify the Urdu word boundaries. The structure of the Urdu dialect renders the gaps between words meaningless. Therefore, it is vital to identify the word boundaries in the Urdu language. Urdu word segmentation has two primary challenges: space insertion and space omissions. The word library can be written in Urdu as کتابخانہ which is incorrect after space insertion at specific points. The new Urdu word “کتاب خانہ” is semantically and syntax-wise correct. On the other hand, in Urdu, there are multiple strings in many words, such as “خوش اخلاق,” Which means that etiquette is a two-string Unigram. If you omit a space between two strings during typing, then it will become “خوشاخلاق,” which is wrong syntactically and semantically.

E. TOKENIZATION

Tokenization is the process of splitting a text into smaller units called tokens. Compared to languages that employ the Latin alphabet, the tokenization of Urdu, a right-to-left script language, maybe a little more difficult. We split the text into words. In Urdu, punctuation marks or spaces are used to separate words. We used white spaces or punctuation marks as delimiters to split the text into words. However,

[1] https://github.com/urduhack/urduhack

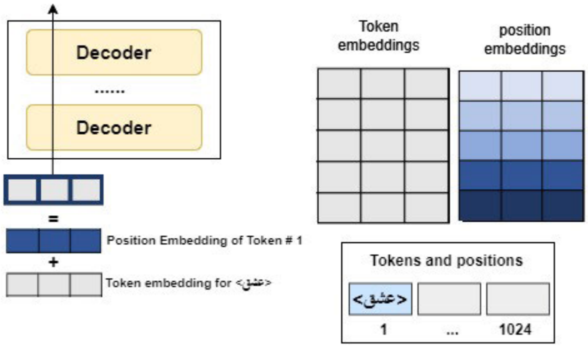

FIGURE 2. Urdu WTE and WPE in GPT2.

we make sure that compound words are intact during tokenization.

F. FEATURE REPRESENTATION

In NLP tasks like text classification, text is often represented as a vector of weighted features. n-gram model, i.e., uni-gram, bi-gram, and tri-grams are utilized to assign the probability to a series of words. In an Urdu sentence " میں آپ کو سمجھتا ہوں " uni-grams are میں (meaning: "I"), آپ (meaning: "you"), کو (meaning: "to/for"), سمجھتا (meaning: "understand"), ہوں (meaning: "am"). Bi-grams are: میں آپ (meaning: "I you"),آپ کو (meaning: "you to/for"), کو سمجھتا (meaning: "to/for understand"), سمجھتا ہوں (meaning: "understand am"). Similarly, tri-grams would be: میں آپ کو (meaning: "I you to/for"), آپ کو سمجھتا (meaning: "you to/for understand"), سمجھتا ہوںکو (meaning: "to/for understand am"). Recent studies show that the pre-trained word embeddings have outperformed previous systems in NLP-related tasks. These word embedding models are trained on massive text data and used for particular purposes. The self-trained FastText embedding model is trained using Wikipedia and Common Crawl (CC) data. FastText model has been trained to understand more than 150 dialects, including Urdu. We use the FastText word vector model in our proposed LSTM and XLM-R model. On the other hand, we use the GPT2 embeddings, which are the sum of word token embeddings (WTE) and word position embeddings(WPE). The GPT2 embeddings for Urdu text are illustrated in Fig. 2.

G. PROPOSED TECHNIQUES

The proposed collection of Machine/Deep Learning (M/DL), GPT2, and XLM-R models are described in detail in this section. The collection of machine learning algorithms, i.e., RF, LR, SVM, XGBoost, and DT, deep learning models, i.e., LSTM and CNN1D+LSTM with FastText embedding model, are used to classify the Urdu text. Furthermore, the state-of-the-art transformer models, GPT2 and XLM-R, with BPE and BERT word embeddings, are used for sentiment analysis. The overall architecture is shown in Fig. 1.

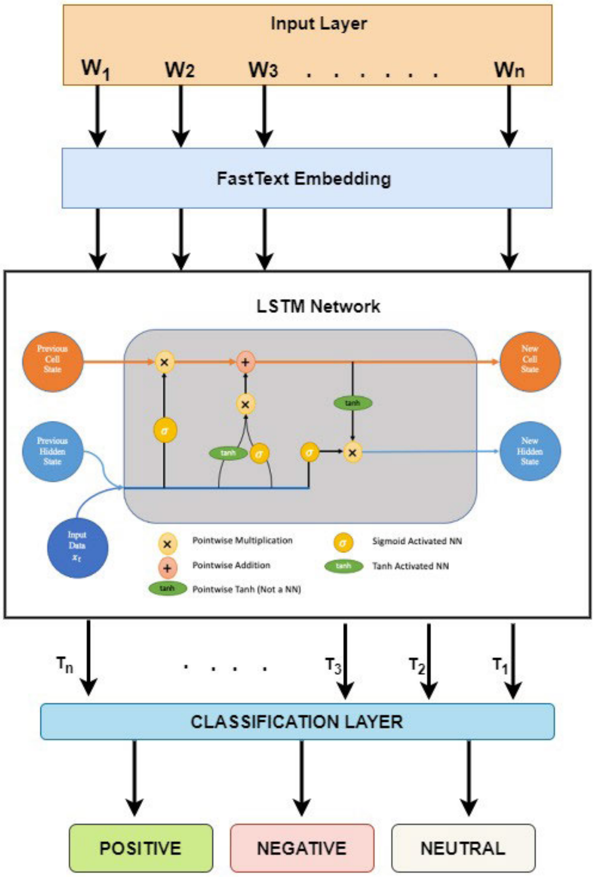

FIGURE 3. Overview of LSTM Classifier.

1) EXPERIMENT ENVIRONMENT SETTINGS

This study employed the Google Cloud Platform (GCP) compute Engine, Tesla T4 GPU for XLM-R and Tesla 100 GPU for the GPT2 model. Both models are employed on the 512 GB RAM. On the other hand, the same hardware settings are used for the ML models for faster training and testing.

2) M/DL CLASSIFICATION MODELS

We employ ML classification models, i.e., RF, LR, SVM, and XGBoost, and DL classification models, i.e., LSTM with FastText embeddings and hybrid CNN-1D with LSTM, to identify the efficiency of our proposed LUCSA-23 corpus for the classification of the Urdu text.

a: HYBRID CNN-1D & LSTM

CNN model is widely used in computer vision. However, we use CNN-1D because recent findings show that CNN-1D performs well in classification tasks. A CNN-1D is extensively useful for capturing new attributes from short sentences of the overall dataset; however, it doesn't matter where the feature is located. To aid with sequence prediction, CNN extracts features from data. Features are initially captured using a 128-layered CNN layer with a kernel size of

4 using max pooling and then fed to an LSTM layer. Results show that the hybrid model tends to overfit after the 6th epoch. Therefore, L2 regularisation is used to reduce weight. The weight matrix values decrease because a regularisation is added to the cost function. Meanwhile, the L2 value for CNN is "3e-3".

$$cf = loss + \frac{\lambda}{2m} \times \sum ||w||^2 \tag{1}$$

where "λ" is the value of regularization, "cf" is the cost function, and "w" is weight.

b: LSTM

LSTM is a Recurrent Neural Network (RNN) architecture providing cutting-edge sequential data results. LSTM is made to preserve the long-term dependencies between text. The LSTM model gets its input from the current word at each time step and its output from the previous or last word, which is then utilized to feed the next state. The previous state generated by the hidden layer is then used for the classification task. Fig. 3 depicts the high-level system architecture of an LSTM network with FastText embedding. Input gate, forget gate, memory cell, and output gate are the four major parts of an LSTM network. The representation of LSTM in mathematical form is as follows:

$$h_{ti} = f(Wx_{ti} + Uh_{t-1} + b) \tag{2}$$

$$i_{ti} = \sigma(W^i x_{ti} + U^i h_{ti-1} + b^i) \tag{3}$$

$$f_{ti} = \sigma(W^f x_{ti} + U^f h_{ti-1} + b^f) \tag{4}$$

$$o_{ti} = \sigma(W^o x_{ti} + U^o h_{ti-1} + b^o) \tag{5}$$

$$g_{ti} = \sigma(W^g x_{ti} + U^g h_{ti-1} + b^g) \tag{6}$$

$$c_{ti} = f_{ti}\theta c_{ti-1} + i_{ti}\theta g_{ti} \tag{7}$$

$$h_{ti} = O_{ti}\theta tanh(c_{ti}) \tag{8}$$

where σ and θ are known as sigmoid function and element-to-element multiplications. For input gate i, Wi, and Ui are two weight matrices, and bi is the bias vector. Similarly, o, c, h, f, and ti represent the output gate, memory cell, hidden state, forget gate, and time, respectively. At the end of the output gate, a classification layer with softmax function is deployed for multiclass sentiment classification of Urdu text.

The LSTM model is trained on the LUCSA-23 dataset with 20-fold cross-validation. In each fold, 80% of the dataset is used for training and 20% as a validation test set. Hyperparameters of the LSTM model are shown in Table 4. Keras NN library is used to implement these two models to evaluate Urdu text sentiment analysis on the proposed LUCSA-23 dataset.

3) TRANSFORMER MODELS

Recent studies show that state-of-the-art transformer-based deep learning language models showcased superior performance in text classification, generation, comprehension, and other NLP tasks. This study evaluated XLM-R and GPT-2 for Sentiment analysis of Urdu Text.

TABLE 4. Hyper-parameters of LSTM Classifier.

Hyper-parameter	Value
LSTM Units	128
Max input Sequence Length	90
Dropout Rate	0.1
Learning Rate	0.001
Batch Size	32
Activation Functions	Sigmoid, Tanh
Gradient Clipping	3.0

XLM-R: Robustly Optimized BERT (XLM-R) and mBERT are the cross-lingual language models, but XLM-R makes several enhancements that allow it to outperform mBERT. In our proposed approach XLM-R leverages much larger CommonCrawl web data across 100 languages and uses a larger 250k SentencePiece vocabulary to improve subword coverage of Urdu Text. Beyond masked language modeling, XLM-R is pre-trained with additional objectives like permutation language modeling and translation language modeling to develop strong cross-lingual representations.

On the other hand, we evaluate the bidirectional quality of the transformer model for Urdu text sentiment analysis in which XLM-R provides contextualized representations capturing word use across different contexts. Before feeding text data into a deep learning model, it needs to be converted into a list of indices, where each index corresponds to a specific token or word in the model's vocabulary and XLM-R model format. Long sentences in the dataset are split into multiple samples. This splitting allows for better handling of long sequences and prevents model capacity and memory constraint issues. To achieve the optimal performance of the model, samples must be converted into integer sequences and the necessary transformations or splitting must be applied for optimal performance.

We employ the XLM-RBase model, which encompasses approximately 355 million parameters. This model comprises 24 layers, 1,027 hidden states, 4,096 feed-forward hidden states, and 16 attention heads. The default maximum sequence length is set to 512 tokens, meaning it can process input sequences of up to 512 tokens and generate corresponding representations. Every sequence's initial token is always [CLS], as the special classification embedding. The final state, denoted as "v," associated with this token, is utilized as the overall representation of the sequence, particularly for classification purposes. To predict the probability of a specific label "l," a straightforward softmax classifier is appended atop the XLM-R model. This classifier incorporates a task-specific parameter matrix, "S," which can be shown in the following equation.

$$(l|v) = softmax\ (Sv) \tag{9}$$

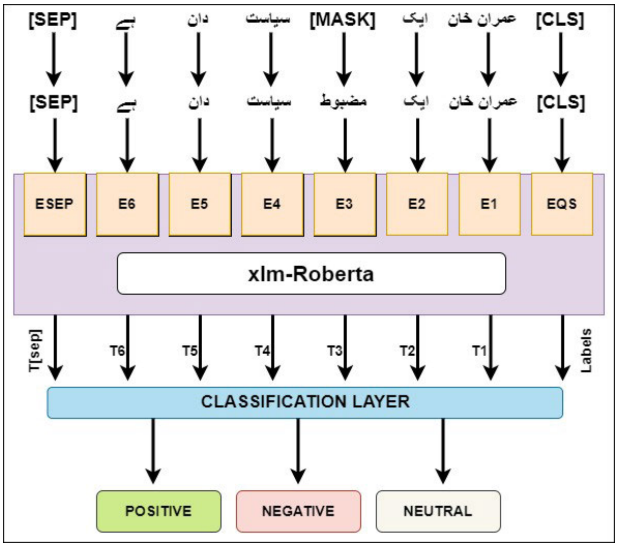

FIGURE 4. Overview of XLM-R Classifier.

TABLE 5. Hyper-parameters of XLM-R Classifier.

Hyper-parameter	Value
Learning rate	2.00E-04
Hidden layers	768
Parameters	350M+
Batch Size	16, 32
Epochs	15
Attention Heads	12

During the fine-tuning process, we simultaneously optimize all the parameters from both XLM-R and the task-specific parameter matrix "S". This is achieved by maximizing the log probability of the correct label. The fine-tuning process involves updating the parameters based on the training data to enhance the model's ability to predict the correct labels for the given task. The detailed architecture of XLM-R for Urdu text sentiment analysis is presented in Fig. 4, and its hyper-parameters are presented in Table 5.

Notably, we first trained XLM-R as the MLM training objective for BERT without the NSP task. It only uses MLM, where tokens are randomly masked, and the model is trained to predict these masked tokens based on the context of the sentence. Secondly, XLM-R is trained with larger batch sizes and longer sequences than BERT. Thirdly, the proposed XLM-RoBERTa Model with a token classification head on top (a linear layer on top of the hidden-states output), e.g., for Named-Entity-Recognition (NER) tasks, is employed for proper classification at the token level.

GPT2: Generative Pretrained Transformer 2 GPT-2 utilizes a multi-layer transformer-based architecture using only the decoder block to facilitate auto-regressive language modeling. The model is composed of multiple identical decoder blocks, each containing a masked self-attention layer followed by a feedforward network which restricts the decoder's ability to extract information from the sentence's earlier words by the use of obfuscation masking of the remaining word locations (plus the word itself).

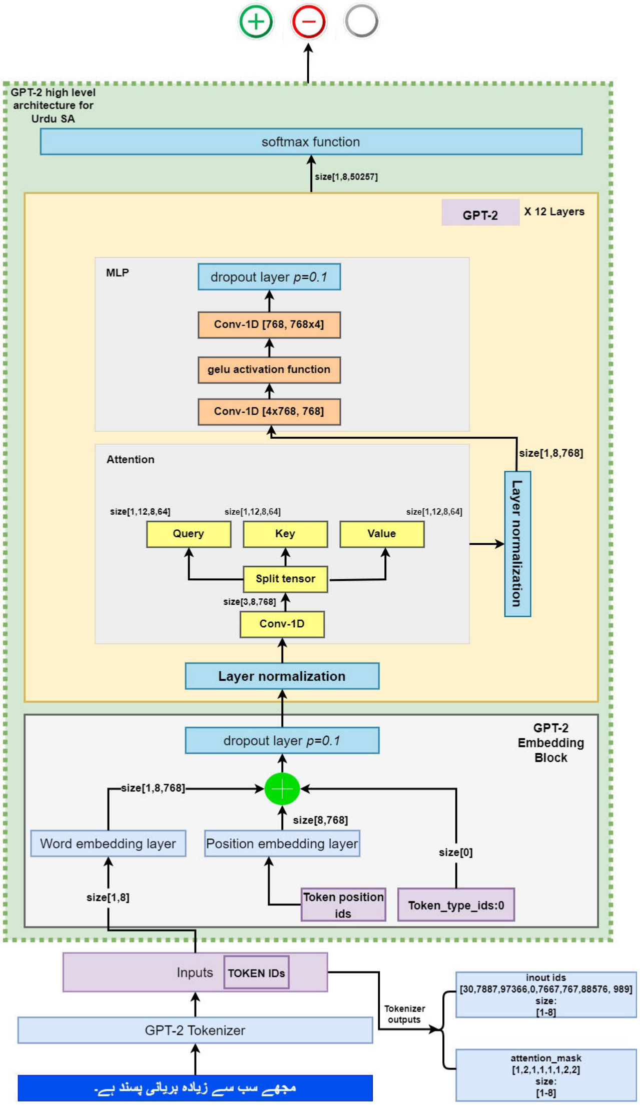

FIGURE 5. Overview of GPT-2 Classifier.

Masked self-attention: The BERT model heavily relies on the self-attention mechanism. One problem with the self-attention in the BERT model is that by only using a single set of trained matrices Q, K, and V, the self-attention could be dominated by just one or a few tokens, and thereby not being able to pay attention to multiple places that might be meaningful. Therefore, by using multi-heads, we aim to linearly combine the results of many independent self-attention computations, thereby expanding the self-attention layers ability to focus on different positions. More concretely, we use multiple sets of mutually independent (Q), (K), and (V) matrices, each being randomly initialized and indepen-

TABLE 6. Performance of ML Models using Uni-gram, Bi-gram, and Tri-gram features.

Feature	Classifier	Precision	Recall	F1 Score	Accuracy
Uni-gram	**VM**	**68.01**	**74.22**	**71.46**	**72.81**
	RF	66.07	71.12	68.40	67.80
	XGBoost	64.90	72.01	66.35	66.70
	LR	67.45	74.19	70.73	72.70
	DT	65.31	72.34	68.64	70.23
Bi-gram	SVM	63.01	70.02	67.28	68.37
	RF	63.97	64.10	64.49	63.58
	XGBoost	62.10	66.98	63.97	65.24
	LR	**65.05**	**69.20**	**66.05**	**67.39**
	DT	63.21	67.04	64.06	65.73
Tri-gram	**SVM**	**64.10**	**72.78**	**70.29**	**69.42**
	RF	52.00	69.21	57.31	57.39
	XGBoost	51.09	68.80	59.89	58.00
	LR	55.05	75.24	64.11	63.24
	DT	52.88	72.04	60.19	62.13

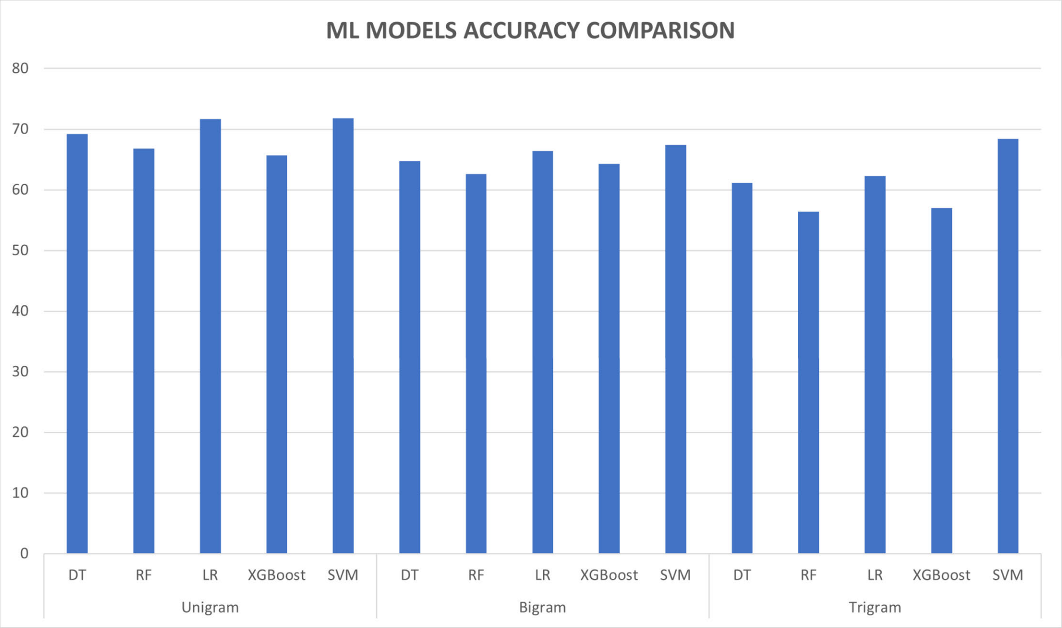

FIGURE 6. Accuracy Comparison of ML Classifiers.

dently trained. With multiple (Q), (K), and (V) matrices, we end up with multiple resulting vectors for every input token vector. Nonetheless, the feedforward neural network in the next step is designed only to accept one vector per word input. To combine those vectors, we concatenate them into a single vector and then multiply it with another weight vector trained simultaneously. This multi-head attention is defined as the multi-head attention mechanism that applies attention pooling in parallel across different representation subspaces. Denoting the query, key, and value vector inputs as Q, K, and V, respectively, the multi-head computation can be expressed as:

$$\mathrm{MultiHead}(Q, K, V) = \mathrm{Concat}(\mathrm{head}_1, \ldots, \mathrm{head}_h)W^O \tag{10}$$

where

$$\mathrm{head}_i = \mathrm{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{11}$$

and W_i^Q, W_i^K, W_i^V are projection matrices for the i^{th} attention head, and W^O projects the concatenated heads into the final representations. This expresses the essential mathematical flow of projecting the inputs into distinct key, value, and query subspaces, applying scaled dot-product attention in each subspace, concatenating the results, and projecting to the final dimensionality.

Residual connections are employed around each block, along with layer normalization on the inputs. GPT-2 uses specialized position embeddings for each token to retain ordering information. The token embeddings themselves represent the meaning of each input word. In self-attention calculations, previous context tokens are represented as context vectors that attend to the current token being processed.

The self-attention mechanism, which is the sole part that increases quadratically with the length of the sequence, becomes a primary focus. Although numerous studies have suggested methods to make attention patterns more sparse and decrease the computational burden of self-attention, these approaches are frequently constrained by implementation issues and result in imposition a basic and unchanging structure on the attention matrix. On the other hand, incorporating a more flexible sparse attention approach sometimes leads to considerably slower execution times than calculating the whole attention using the Flash method of Dao et al. (2022). We enhance the capabilities GPT2 by using FlashAttention to incorporate a wide range of attention sparsity patterns, including key/query dropping and hashing-based attention.

The topmost layer uses a softmax function to output a probability distribution predicting the next token. For the largest version of GPT-2, the architecture contains 24 decoder blocks for 1.5 billion parameters. In the typical transformer design, the encoder generates both a word embedding and a context vector, which are supplied to the decoder together. The context vector is initialized to zero for the first word embedding in GPT-2. The overall design of the GPT2 model for Urdu text sentiment analysis is shown in Fig. 5.

V. RESULTS

This section explores the detailed analysis of experimental results, demonstrating the significance and efficiency of machine learning, deep learning, and transformer language models for sentiment analysis of Urdu text.

The results obtained from ML techniques with various features on our proposed LUCSA-23 corpus are shown in Table 6. The findings demonstrate that the overall accuracy of the SVM model marginally surpasses the other machine learning algorithms when evaluated on the LUCSA-23 dataset, achieving an accuracy of 72.81%. The comparison of machine learning techniques in terms of accuracy allows us to gauge their effectiveness in classifying sentiments within the LUCSA-23 corpus. It indicates that SVM, among the evaluated algorithms, demonstrates high accuracy in sentiment classification for the proposed dataset. The overall comparison of the accuracy of ML models on different character N-grams is shown in Fig. 6.

The machine learning classifier RF is achieving 57.39% accuracy when utilizing tri-gram features. Interestingly, this results in the lowest accuracy among all the ML classifiers evaluated in the study. Furthermore, when comparing the performance of different feature types, it is observed that all ML classifiers exhibit better results when using uni-gram word features compared to bi-gram and tri-gram word features. This finding suggests that including tri-gram features may introduce additional complexity or noise into the classification process, leading to decreased accuracy. On the other hand, using uni-gram word features, which capture individual words in isolation, seems more effective for sentiment analysis in the given context. These results highlight the importance of feature selection and demonstrate that the choice of feature type can significantly impact the performance of ML classifiers. It suggests that the classifiers benefit from focusing on individual words rather than incorporating higher-order word features like bi-grams or tri-grams.

The results obtained by DL classifiers (CNN1D+ LSTM and LSTM) are superior to those obtained by ML classifiers and classical rule-based techniques. This shows that the DL classifiers (CNN1D+LSTM and LSTM) surpass the baseline results of ML classifiers, as shown in Table 7. Findings show that just after the 11 epochs of experiments, the LSTM model shows indications of overfitting. This is discovered when the process of experimenting is carried out. Consequently, the LSTM model's training process is terminated. After 15 epochs, the CNN1D+LSTM hybrid model produces some useful results. It is discovered that the CNN1D+LSTM hybrid model can potentially be beneficial for Urdu sentiment analysis. This is particularly true when we compare it to other standard ML algorithms. While we compare using a single-layer LSTM, the classification performance is significantly enhanced when employing two stacked LSTM layers. Similarly, the outputs of the proposed model reveal a modest improvement when a two-layer LSTM is used across all the utilized datasets. Because LSTMs can capture information in both the forward and backward directions, two LSTM layers are appropriate for generating more comprehensive feature representations of Urdu phrases. This results in easier classification being possible. In addition, it has been discovered that the LSTM and CNN1D models produce somewhat better outcomes when we utilize an attention layer instead of a max-pooling (MP) layer in the architecture.

According to the results in Table 8, transformer-based classifiers surpass both ML and DL classifiers. The findings indicate that the proposed GPT-2 classifier is superior to the XLM-R classifier in terms of accuracy, precision, recall, and F1 measure of 95%, 94.09%, 95.05%, and 94.49%, respectively.

We also compute confusion matrices to evaluate the accuracy of the proposed classifier for Urdu text sentiment analysis. Fig. 7 and Fig. 8 show confusion matrices of XLM-R and GPT-2 classifiers using the LUCSA-23 Urdu corpus, respectively.

From Fig. 7 and Fig. 8, it can be concluded that the positive phrases are correctly classified as positive with 95.4% accuracy for the GPT-2 classifier. However, only 2.38% and 3.08% of positive phrases are mistakenly classified as negative and neutral, respectively. Moreover, findings show that 92.76% of negative phrases are correctly classified as negative. However, only 3.08% and 4.16% phrases are incorrectly classified as positive and neutral, respectively. Similarly, findings show that 93.8% sentences

TABLE 7. Performance of DL classifiers.

Embeddings	Classifier	Accuracy	Precision	Recall	F1 Score
-	**CNN1D+LSTM**	**73.1**	**69.79**	**75.7**	**72.82**
FastText	LSTM	68.09	67.53	71.7	69.69

TABLE 8. Performance of Transformer-based Classifiers.

Classifier	Accuracy	Precision	Recall	F1 Score
XLM-R	89.6	87.67	80.56	83.45
GPT-2	**95.0**	**94.09**	**95.05**	**94.49**

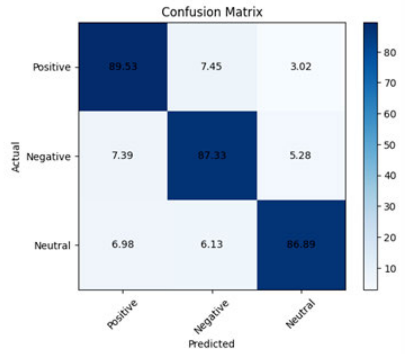

FIGURE 7. Confusion Matrix of XLM-R.

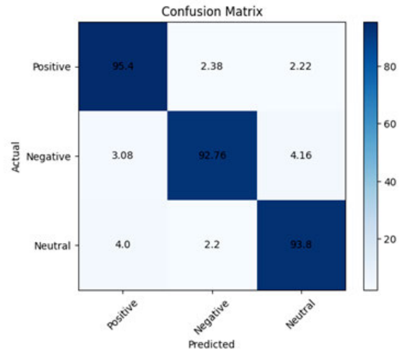

FIGURE 8. Confusion Matrix of GPT-2.

are correctly classified as neutral. However, 4.00% and 2.20% are incorrectly classified as positive and negative, respectively. Such findings demonstrate how effectively the proposed classifier classifies the neutral reviews.

VI. DISCUSSION

This study aimed to analyze the findings of sentiment analysis of Urdu text using state-of-the-art Large language models, deep learning and machine learning approaches. Our findings indicate that the Generative pre-trained transformer model (GPT2), with an accuracy of 95%, outperformed other models, such as XLM-RoBERTa and CNN1D+LSTM, in classifying the sentiment of Urdu text. Using BERT word embeddings for text representation significantly contributed to the improved performance of the large language models. The comprehensive comparison between DL and the proposed transformer classifiers regarding accuracy is depicted in Fig. 9. Urdu stands out among languages due to its exceptionally unique and remarkably complex morphological structure. A fusion of various languages, including Sanskrit, Hindi, Arabic, Turkish, and Persian, contributes to the presence of loan words, adding to its linguistic richness. However, these diverse linguistic elements often challenge algorithms, leading to errors in categorization. The normalization of Urdu text remains imperfect, further complicating the process. Tokenizing Urdu text necessitates adjusting word boundaries, as they are not always apparent, and altering the sequence of words within a phrase may not alter its underlying meaning. Moreover, the manual annotation of user evaluations introduces another potential source of classification errors. The paper aims to analyze sentiment analysis results of Urdu text using advanced Large language models, deep learning, and machine learning approaches. Our findings indicate that the Generative pre-trained transformer model (GPT2), with an accuracy of 95%, outperformed other models, such as XLM-RoBERTa and CNN1D+LSTM, in classifying the sentiment of Urdu text. Using BERT word embeddings for text representation significantly contributed to the improved performance of the large language models. Creating a new dataset, LUCSA-23, for the low-resource language Urdu is one of the main outcomes of our study. Over 65,000 user reviews from a range of areas, including Pakistani politics, sports, entertainment, and food, are included in this dataset. Preprocessing the Urdu text and BERT embeddings, then using these embeddings as input for the proposed technique. Our findings showed that, with an amazing accuracy of 95%, the proposed classifiers surpassed current state-of-the-art techniques. We build upon the results of previous studies by focusing on the notable deficiency in sentiment analysis for languages with limited resources. While earlier studies have primarily concentrated on the English language, our research focuses on Urdu, offering a valuable dataset and showcasing the effectiveness of transformer models in this specific context. Our study can provide substantial benefits to future scholars in this field. The LUCSA-23 dataset is a significant resource for training and assessing novel models. It helps save time and effort by eliminating the need for data gathering and annotation. Our methodological insights in preprocessing, embedding generation, and classifier building provide a robust framework for similar challenges. Future researchers could investigate the transfer of models between languages, specifically examining how well models trained in Urdu can be adapted to other languages with limited resources.

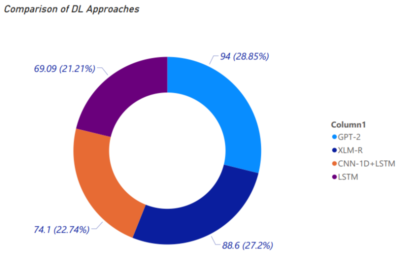

FIGURE 9. **Comparison of DL and Transformer-based Classifiers.**

Exploring the use of more recent or hybrid transformer models can improve sentiment classification accuracy. The resilience and practicality of sentiment analysis can be enhanced by utilizing our models in practical scenarios, such as monitoring social media and analyzing customer comments in Urdu. Additionally, creating techniques to effectively handle intricate textual components such as sarcasm and idioms would further enhance the performance of sentiment analysis. These instructions can enhance the field and expand the influence of NLP technology in various linguistic circumstances.

VII. CONCLUSION AND LIMITATIONS

Several restrictions are encountered during the experiments. One significant limitation is the unbalanced nature of the data, particularly in sentiment distribution. Specifically, the number of positive sentences exceeds the number of negative sentences. This class imbalance can impact the classifier's ability to accurately classify sentiments, as it may be more biased towards predicting the majority class.

Another limitation relates to the tokenization process. The Urdu language contains many compounds, which are not considered in this study due to restrictions imposed by the employed tokenization technique. This limitation could affect the classifier's performance in capturing the nuances and meaning of compound words, potentially leading to reduced accuracy in sentiment analysis. For example, the word ضرورت مند can be written with and without spaces as ضرورتمند.

Furthermore, the variability in writing styles within Urdu poses a challenge. The presence of diverse writing styles can introduce inconsistencies in the data and impact the classifier's ability to generalize across different text samples. This variability in writing style may lead to lower accuracy and pose a hurdle in achieving robust sentiment analysis results.

One limitation mentioned earlier in the study is the tokenization restriction, which causes compound words in Urdu language to be split into separate tokens. This splitting of compound words can reduce their effectiveness and hinder accurately representing their intended meaning. However, it is worth noting that advancements in tokenization techniques are continually being made to address this issue. In the future, one potential solution to this problem is the application of multiword tokenization. Multiword tokenization involves treating compound words as single tokens, preserving their integrity and capturing their inherent meaning. By adopting this approach, the tokenizer could recognize and represent compound words more effectively, leading to improved performance in sentiment analysis tasks.

As large amounts of useful information for many purposes are created on social media sites, evaluating public opinion of a product or service requires sentiment analysis. Few experiments have been conducted in Urdu sentiment analysis using classical machine-learning approaches. The available data corpus is limited, consisting of just two data classes. In contrast, we develop a benchmark for Urdu sentiment analysis using various M/DL classifiers. Furthermore, while using our proposed XLM-R and GPT-2 classifiers on the LUCSA-23 dataset, we achieve an F1 score of 83.45 percent and 94.49 percent, respectively. This article paves the way for future deep learning studies to focus on developing resource-constrained language-independent models. Our research shows that deep learning using pre-trained word embeddings and multi-lingual transformer models effectively handles difficult and under-resourced languages like Urdu.

We intend to exploit state-of-the-art transformer-based classifiers to improve the results and expand this research to 5-7 classes, such as sad, anger, happy, etc. To that end, we make the created dataset accessible to the public, hoping it might significantly contribute to Urdu sentiment analysis.

REFERENCES

[1] I. Mogotsi, C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008, p. 482.

[2] J. Hutchins, "The first public demonstration of machine translation: The Georgetown-IBM system, 7th January 1954," Comput. Sci. Linguistics, U.K., Tech. Rep. 132677, 2006.

[3] F. Neri, C. Aliprandi, F. Capeci, M. Cuadros, and T. By, "Sentiment analysis on social media," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining*, Aug. 2012, pp. 919–926.

[4] S. Bird, E. Klein, and E. Loper, *Natural Language Processing With Python: Analyzing Text With the Natural Language Toolkit*. Sebastopol, CA, USA: O'Reilly Media, 2009.

[5] G. F. Simons and C. D. Fennig. (2017). *Ethnologue: Languages of the World*. SIL Int., Dallas, TX, USA. [Online]. Available: http://www.ethnologue.com

[6] T. Kaneda, C. Greenbaum, and K. Patierno, *World Population Data Sheet*. Washington, DC, USA: Population Reference Bureau, 2015.

[7] A. Julka, "From Hindi to Urdu: A social and political history," *Strategic Anal.*, vol. 36, no. 5, pp. 829–830, Sep. 2012.

[8] A. Daud, W. Khan, and D. Che, "Urdu language processing: A survey," *Artif. Intell. Rev.*, vol. 47, no. 3, pp. 279–311, Mar. 2017.

[9] N. Asher, F. Benamara, and Y. Y. Mathieu, "Appraisal of opinion expressions in discourse," *Lingvisticæ Investigationes*, vol. 32, no. 2, pp. 279–292, Dec. 2009.

[10] N. Asher, F. Benamara, and Y. Y. Mathieu, "Distilling opinion in discourse: A preliminary study," in *Proc. 22nd Int. Conf. Comput. Linguistics (COLING)*, 2008, pp. 7–10.

[11] S. Somasundaran, J. Ruppenhofer, and J. Wiebe, "Discourse level opinion relations: An annotation study," in *Proc. 9th SIGdial Workshop Discourse Dialogue (SIGdial)*, 2008, pp. 129–137.

[12] M. Taboada, K. Voll, and J. Brooke, "Extracting sentiment as a function of discourse structure and topicality," Simon Fraser Univ. School Comput. Sci., Comput. Sci. Linguistics, U.K., Rep. 15080426, 2008, pp. 1–22.

[13] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," 2011, *arXiv:1103.2903*.

[14] A. Esuli and F. Sebastiani, "SENTIWORDNET: A publicly available lexical resource for opinion mining," in *Proc. 5th Int. Conf. Lang. Resour. Eval.* Genoa, Italy: European Language Resources Association (ELRA), 2006, pp. 417–422.

[15] C. Hutto and E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. Int. AAAI Conf. Web Social Media*, vol. 8, May 2014, pp. 216–225.

[16] E. Cambria, R. Speer, C. Havasi, and A. Hussain, "SenticNet: A publicly available semantic resource for opinion mining," in *Proc. AAAI Fall Symp., Commonsense Knowl.*, 2010, pp. 1–5.

[17] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining Text Data*. Boston, MA, USA: Springer, 2012, pp. 415–463.

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

[19] R. Sharma, S. Nigam, and R. Jain, "Polarity detection at sentence level," *Int. J. Comput. Appl.*, vol. 86, no. 11, pp. 29–33, Jan. 2014.

[20] A. Kornai, "Digital language death," *PLoS ONE*, vol. 8, no. 10, Oct. 2013, Art. no. e77056.

[21] H. Arif, K. Munir, A. S. Danyal, A. Salman, and M. M. Fraz, "Sentiment analysis of Roman Urdu/Hindi using supervised methods," in *Proc. ICICC*, vol. 8, 2016, pp. 48–53.

[22] J. Barnes, P. Lambert, and T. Badia, "MultiBooked: A corpus of Basque and Catalan hotel reviews annotated for aspect-level sentiment classification," 2018, *arXiv:1803.08614*.

[23] A. Nawaz, S. Asghar, and S. H. A. Naqvi, "A segregational approach for determining aspect sentiments in social media analysis," *J. Supercomput.*, vol. 75, no. 5, pp. 2584–2602, May 2019.

[24] S. Haque, T. Rahman, A. K. Shakir, M. S. Arman, K. B. B. Biplob, F. A. Himu, D. Das, and M. S. Islam, "Aspect based sentiment analysis in Bangla dataset based on aspect term extraction," in *Proc. 2nd EAI Int. Conf. ICONCS Cyber Security Comput. Sci.*, Dhaka, Bangladesh. Cham, Switzerland: Springer, 2020, pp. 403–413.

[25] Y. Zhao, B. Qin, and T. Liu, "Creating a fine-grained corpus for Chinese sentiment analysis," *IEEE Intell. Syst.*, vol. 30, no. 1, pp. 36–43, Jan. 2015.

[26] R. Kumar, A. K. Ojha, S. Malmasi, and M. Zampieri, "Benchmarking aggression identification in social media," in *Proc. 1st Workshop Trolling, Aggression Cyberbullying*, 2018, pp. 1–11.

[27] K. V. S. Prasad, S. Virk, J. Camilleri, K. Angelov, K. Kaljurand, O. Caprotti, and A. Ranta, "Computational evidence that Hindi and Urdu share a grammar but not the lexicon," in *Proc. 3rd Workshop South Southeast Asian Natural Language Process. (SANLP)*, vol. 7427, 2012, pp. 1–12.

[28] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, A. Mehmood, and M. T. Sadiq, "Document-level text classification using single-layer multisize filters convolutional neural network," *IEEE Access*, vol. 8, pp. 42689–42707, 2020.

[29] M. N. Asim, M. U. G. Khan, M. I. Malik, A. Dengel, and S. Ahmed, "A robust hybrid approach for textual document classification," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 1390–1396.

[30] M. Abdul-Mageed, M. Diab, and S. Kübler, "SAMAR: Subjectivity and sentiment analysis for Arabic social media," *Comput. Speech Lang.*, vol. 28, no. 1, pp. 20–37, Jan. 2014.

[31] Y. Kim, H. Lee, and K. Jung, "AttnConvnet at SemEval-2018 task 1: Attention-based convolutional neural networks for multi-label emotion classification," 2018, *arXiv:1804.00831*.

[32] S. Vashishtha and S. Susan, "Inferring sentiments from supervised classification of text and speech cues using fuzzy rules," *Proc. Comput. Sci.*, vol. 167, pp. 1370–1379, Jan. 2020.

[33] L. Zhu, Z. Zhu, C. Zhang, Y. Xu, and X. Kong, "Multimodal sentiment analysis based on fusion methods: A survey," *Inf. Fusion*, vol. 95, pp. 306–325, Jul. 2023.

[34] P. D. Mahendhiran and S. Kannimuthu, "Deep learning techniques for polarity classification in multimodal sentiment analysis," *Int. J. Inf. Technol. Decis. Making*, vol. 17, no. 03, pp. 883–910, May 2018.

[35] A. Agarwal, A. Yadav, and D. K. Vishwakarma, "Multimodal sentiment analysis via RNN variants," in *Proc. IEEE Int. Conf. Big Data, Cloud Comput., Data Sci. Eng. (BCD)*, May 2019, pp. 19–23.

[36] Y. Yao, V. Pérez-Rosas, M. Abouelenien, and M. Burzo, "MORSE: Multimodal sentiment analysis for real-life settings," in *Proc. Int. Conf. Multimodal Interact.*, Oct. 2020, pp. 387–396.

[37] I. O. Hussien, K. Dashtipour, and A. Hussain, "Comparison of sentiment analysis approaches using modern Arabic and Sudanese Dialect," in *Proc. 9th Int. Advances Brain Inspired Cogn. Syst.*, Xi'an, China. Cham, Switzerland: Springer, Jul. 2018, pp. 615–624.

[38] Q. Portes, J. Carvalho, J. Pinquier, and F. Lerasle, "Multimodal neural network for sentiment analysis in embedded systems," in *Proc. 16th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2021, pp. 387–398.

[39] K. Shakeel, G. R. Tahir, I. Tehseen, and M. Ali, "A framework of Urdu topic modeling using latent Dirichlet allocation (LDA)," in *Proc. IEEE 8th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2018, pp. 117–123.

[40] S. Rahmani, S. Hosseini, R. Zall, M. R. Kangavari, S. Kamran, and W. Hua, "Transfer-based adaptive tree for multimodal sentiment analysis based on user latent aspects," *Knowl.-Based Syst.*, vol. 261, Feb. 2023, Art. no. 110219.

[41] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Gener. Comput. Syst.*, vol. 115, pp. 279–294, Feb. 2021.

[42] C. Baziotis, N. Athanasiou, A. Chronopoulou, A. Kolovou, G. Paraskevopoulos, N. Ellinas, S. Narayanan, and A. Potamianos, "NTUA-SLP at SemEval-2018 task 1: Predicting affective content in tweets with deep attentive RNNs and transfer learning," 2018, *arXiv:1804.06658*.

[43] S. Hassan, S. Shaar, and K. Darwish, "Cross-lingual emotion detection," 2021, *arXiv:2106.06017*.

[44] C. Gan, L. Wang, and Z. Zhang, "Multi-entity sentiment analysis using self-attention based hierarchical dilated convolutional neural network," *Future Gener. Comput. Syst.*, vol. 112, pp. 116–125, Nov. 2020.

[45] N. Majumder, R. Bhardwaj, S. Poria, A. Gelbukh, and A. Hussain, "Improving aspect-level sentiment analysis with aspect extraction," *Neural Comput. Appl.*, vol. 34, no. 11, pp. 8333–8343, Jun. 2022.

[46] A. Altaf, M. W. Anwar, M. H. Jamal, S. Hassan, U. I. Bajwa, G. S. Choi, and I. Ashraf, "Deep learning based cross domain sentiment classification for Urdu language," *IEEE Access*, vol. 10, pp. 102135–102147, 2022.

[47] N. Azam, B. Tahir, and M. A. Mehmood, "Sentiment and emotion analysis of text: A survey on approaches and resources," *Language Technol.*, vol. 87, pp. 1–8, Feb. 2020.

[48] S. M. Ghulam and T. R. Soomro, "Twitter and Urdu," in *Proc. Int. Conf. Comput., Math. Eng. Technol. (iCoMET)*, Mar. 2018, pp. 1–6.

[49] M. Y. Khan, T. Ahmed, S. Wasi, and M.-M. S. Siddiqui, "Enhancing sarcasm and sentiment analysis with cognitive relationship: A context-aware approach for Urdu—A resource poor language," in *Computational Intelligence and Neuroscience*, vol. 8. Hoboken, NJ, USA: Wiley, 2022.

[50] M. A. Chhajro, A. Arshad, K. Luhana, A. Wagan, M. Muneed, and A. I. Umrani, "Electronic ledger management: A mobile-enabled sentiment reviews analysis of Urdu language," *J. Tianjin Univ. Sci. Technol.*, vol. 55, no. 6, pp. 133–141, 2022.

[51] K. Iqbal Malik, "Urdu news content classification using machine learning algorithms," *Lahore Garrison Univ. Res. J. Comput. Sci. Inf. Technol.*, vol. 6, no. 1, pp. 22–31, Mar. 2022.

[52] M. Z. Ali, K. Javed, E. Ul Haq, and A. Tariq, "Sentiment and emotion classification of epidemic related bilingual data from social media," 2021, *arXiv:2105.01468*.

[53] M. F. Bashir, A. R. Javed, M. U. Arshad, T. R. Gadekallu, W. Shahzad, and M. O. Beg, "Context-aware emotion detection from low-resource Urdu language using deep neural network," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 22, no. 5, pp. 1–30, May 2023.

[54] N. Ashraf, L. Khan, S. Butt, H.-T. Chang, G. Sidorov, and A. Gelbukh, "Multi-label emotion classification of Urdu tweets," *PeerJ Comput. Sci.*, vol. 8, pp. 237–249, Apr. 2022, Art. no. e896.

[55] S. Shams, B. Sadia, and M. Aslam, "Intent detection in Urdu queries using fine-tuned BERT models," in *Proc. 16th Int. Conf. Open Source Syst. Technol. (ICOSST)*, Dec. 2022, pp. 1–6.

[56] A. Mehmood, M. S. Farooq, A. Naseem, F. Rustam, M. G. Villar, C. L. Rodríguez, and I. Ashraf, "Threatening Urdu language detection from tweets using machine learning," *Appl. Sci.*, vol. 12, no. 20, p. 10342, Oct. 2022.

[57] P. Vyas, M. Reisslein, B. P. Rimal, G. Vyas, G. P. Basyal, and P. Muzumdar, ‘‘Automated classification of societal sentiments on Twitter with machine learning,’’ *IEEE Trans. Technol. Soc.*, vol. 3, no. 2, pp. 100–110, Jun. 2022.

[58] 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, pp. 122–135, Dec. 2021.

[59] A. Altaf, M. W. Anwar, M. H. Jamal, and U. I. Bajwa, ‘‘Exploiting linguistic features for effective sentence-level sentiment analysis in Urdu language,’’ *Multimedia Tools Appl.*, vol. 82, no. 27, pp. 41813–41839, Nov. 2023, doi: 10.1007/s11042-023-15216-0.

MUHAMMAD REHAN ASHRAF received the M.Sc. and M.Phil. degrees from Quaid-i-Azam University Islamabad, Pakistan. He is currently an Assistant Professor with the Department of Computer Sciences, COMSATS University Islamabad, Vehari Campus, Pakistan. His research interests include machine learning, data mining, and digital image processing.

MUZAMMAL HUSSAIN received the master’s degree in computer science from COMSATS University Islamabad, Pakistan. He is currently a Lecturer with the Government College University Faisalabad, Sahiwal Campus, Pakistan. His research interests include natural language processing (NLP), machine learning, and deep learning.

M. ARFAN JAFFAR received the M.Sc. degree in computer science from Quaid-i-Azam University, Islamabad, Pakistan, in March 2003, and the M.S. and Ph.D. degrees in computer science from the FAST National University of Computer and Emerging Sciences, in 2006 and 2009, respectively. He received a postdoctoral research fellowship in South Korea and carried-out research at the top raking Korean University, ‘‘Gwangju Institute of Science and Technology, Gwangju, South Korea,’’ from 2010 to 2013. He was an Assistant Professor with Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia, from March 2013 to August 2018. He is currently the Dean of the Faculty of Computer Science and Information Technology, Superior University, Lahore, Pakistan. He is also the Director of the Intelligent Data Visual Computing Research (IDVCR). His research interests include image processing, data science, machine learning, computer vision, artificial intelligence, and medical images. He is a Reviewer of 30 reputed international journals, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Industrial Electronics, *Pattern Recognition*, and *Knowledge and Information Systems*.

WAHEED YOUSUF RAMAY received the Ph.D. degree from the University of Science and Technology Beijing (USTB), China. He is currently an Assistant Professor with Air University, Islamabad. His academic and clinical focus is on using algorithms (deep learning, machine learning, and big data analysis), advanced text analysis techniques, and sentiment analysis.

MUHAMMAD FAHEEM (Member, IEEE) received the B.Sc. degree in computer engineering from the Department of Computer Engineering, University College of Engineering and Technology, Bahauddin Zakariya University, Multan, Pakistan, in 2010, the M.S. degree in computer science from the Faculty of Computer Science and Information Systems, Universiti Teknologi Malaysia, Johor Bahru, Malaysia, and the Ph.D. degree in computer science from the Faculty of Engineering, Universiti Teknologi Malaysia, in 2021. Previously, he was a Lecturer with the COMSATS Institute of Information and Technology, Pakistan, from 2012 to 2014. He was an Assistant Professor with the Department of Computer Engineering, Abdullah Gül University, from 2014 to 2022, Tükiye. He is currently a Researcher with the School of Computing (Innovations and Technology), University of Vaasa, Vaasa, Finland. His research interests include cybersecurity, blockchain, artificial intelligence, smart grids, and smart cities. He has authored several papers in refereed journals and conferences and served as a Reviewer for numerous journals in IEEE, Elsevier, Springer, Willey, Hindawi, and MDPI.