IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017. Doi Number

Urdu sentiment analysis with deep learning methods

LAL KHAN[1], AMMAR AMJAD[2], NOMAN ASHRAF[3], HSIEN-TSUNG CHANG[4567], (Member, IEEE) ALEXANDER GELBUKH[8]

[1] Department of Computer Science and Information Engineering, Chang Gung University, Taoyuan, Taiwan, (e-mail: lal.khan.buzdar@gmail.com)
[2] Department of Computer Science and Information Engineering, Chang Gung University, Taoyuan, Taiwan (e-mail: ammar.amjad12@gmail.com)
[3] CIC, Instituto Politécnico Nacional, Mexico (e-mail: nomanashraf@sagitario.cic.ipn.mx)
[4] Bachelor Program in Artificial Intelligence, Chang Gung University, Taoyuan, Taiwan (e-mail: smallpig@widelab.org)
[5] Department of Computer Science and Information Engineering, Chang Gung University, Taoyuan, Taiwan
[6] Department of Physical Medicine and Rehabilitation, Chang Gung Memorial Hospital, Taoyuan, Taiwan
[7] Artificial Intelligence Research Center, Chang Gung University, Taiwan
[8] CIC, Instituto Politécnico Nacional, Mexico (e-mail: gelbukh@gelbukh.com)

Corresponding author: Hsien-Tsung Chang (e-mail: Smallpig@widelab.org).

ABSTRACT Although over 169 million people in the world are familiar with the Urdu language and a large quantity of Urdu data is being generated on different social websites daily, very few research studies and efforts have been completed to build language resources for the Urdu language and examine user sentiments. This study is focused on Urdu sentiment analysis of user reviews. After collecting Urdu user reviews about different genres from different websites, Urdu user reviews were annotated by three human experts. The primary objective of this study is twofold: (1) develop a benchmark dataset for resource-deprived Urdu language for sentiment analysis and (2) evaluate various machine learning and deep learning sentiment analysis models. Six machine learning and two deep learning classifiers, random forest (RF), naïve Bayes, support vector machine (SVM), AdaBoost, multilayer perceptron (MLP), logistic regression, LSTM, and CNN1D, are implemented. The results of all machine learning models are compared on the basis of different N-gram feature models. We implement all the above mentioned machine learning classifiers with unigram, bigram, trigram, uni-bigram, and uni-trigram features and deep learning models with the FastText word embedding model. Finally, the results of all classifiers are analyzed. The logistic regression model outperforms all other models in terms of accuracy of 0.8194, precision of 0.7995, recall of 0.8426, and an F1 measure of 0.8205 with the uni-trigram feature.

INDEX TERMS Machine Learning, Natural Language Processing (NLP), Urdu Sentiment Analysis, Word Embedding, Deep Learning

I. INTRODUCTION

In recent years, with the remarkable increase in the use of hand-held devices and the internet, the use of social media such as Twitter, Facebook, and blogs has been equally increasing by individual users to express their emotions and sentiments [1-3]. Currently, people want to publicly share their opinions, feedback, reviews, and feelings about products, politics, any viral news or any activity. Businesses and institutes are searching for useful information from social media views [4-7]. Therefore, there is a need for artificial intelligence systems such as sentiment analyzers, which can convert raw social media user data into useful information. To recognize and detect emotions and for sentiment analysis, languages such as English, French, German, and other European languages are considered rich languages in terms of tool accessibility. Nevertheless, languages such as Urdu, Punjabi, and Hindi are judged resource deprived [8]. Urdu is very different from other languages due to its morphological structure as it starts from right to left. Due to its morphological structure, the Urdu script is not very common; therefore, a standard dataset or corpora must perform natural language processing tasks.

Sentiment analysis of the Urdu language is equally essential, as it is important in other languages, as it assists non-Urdu speakers in grasping the basic feelings, emotions, and opinions of any user behind a text. Urdu is one of the most common, official, national languages in Pakistan, and it is also spoken in many state and union tertiaries of India. In regard to social

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

media, several native Urdu speakers use Urdu script on platforms such as Twitter, Facebook, and YouTube to express their emotions, feelings, and opinions. Therefore, it is equally important to detect and recognize sentiments to understand the opinions and feelings of native Urdu speakers.

There are many problems with Urdu sentiment analysis, such as a shortage of recognized lexical resources [9], [10], [11]. Mostly, Urdu websites are developed in a descriptive arrangement rather than a proper text encoding structure; due to this hurdle, it is challenging to create a benchmark corpus in Urdu. Urdu sentiment analysis has not yet been investigated completely even after its considerable use; most of the existing literature studies are focused on different aspects of language processing [12], [13].

Therefore, in this paper, the primary focus is to contribute a benchmark corpus for Urdu sentiment analysis, hereafter named the UCSA-20 dataset, and recognize sentiment using machine and deep learning.

This study is motivated by the following. Urdu is the national and official language of Pakistan and commonly spoken medium in many states and union tertiaries of India; more than 169 million people can speak Urdu. In recent times, due to the tremendous use of social media and web applications, Urdu data have been increasing. Therefore, examining the sentiment and feelings of consumers, and people about any product is essential.

The main contributions of this research are as follows:

- Our primary goal is to contribute a manually annotated important benchmark Urdu dataset for sentiment analysis tasks named the UCSA-2020 corpus and make it publicly accessible to the research community. To create the proposed dataset, 9,601 user reviews were collected from various genres. Initially, three human experts manually annotated each review into a positive or negative class.
- Our second contribution is comparing different state-of-the-art machine learning models (SVM, naïve Bayes, RF, MLP, logistic regression, and AdaBoost) with different word level features to display the effectiveness of the UCSA-2020 corpus on the Urdu sentiment analysis task.
- To the best of our knowledge, no research study shows the use of deep learning models with pre-trained word embedding such as fastText for Urdu sentiment analysis; therefore, another main contribution is to study the effectiveness of word embedding in resource-deprived languages such as Urdu in sentiment analysis.
- As Urdu is a resource-deprived language, the authors are confident that the UCSA-20 dataset and model Python code will be beneficial in advancing Urdu sentiment analysis research, permitting an instant evaluation of existing advanced sentiment analysis methods in the Urdu language and building and examining new methods in Urdu sentiment analysis.

The rest of the paper is organized into six sections. Section II presents the background related work. In section III, the description of the data collection and corpus generation is presented. In section IV, the overall methodology of the paper is explained. Section V contains details of the experiments and results. Section VI concludes the paper.

II. BACKGROUND AND RELATED WORK

This section first discusses famous sentiment analysis corpora for user sentiment and supervised machine learning classification techniques. In the second part of this section, deep learning models with word embedding's for different domains are explained. Sentiment analysis in the Urdu language is discussed in the third part of this section.

A. OVERVIEW OF SENTIMENT ANALYSIS AND DATASETS

To create a benchmark dataset for sentiment analysis, SemEval contests are considered one of the most noticeable literature efforts. In the series of SemEval competitions to examine sentiment analysis, researchers perform distinct tasks using different datasets. These datasets have been developed in Arabic and English [14]. Generally, these corpora contain user tweets from Twitter, SMS, and different products, such as laptops, and service reviews, such as hotels. The SemEval corpus 2013 edition consists of Twitter data and SMS data; Twitter corpora were divided into training, development and testing data 9,728, 1,654, and 3,813, respectively. However, SMS corpora were used for testing, which contains 2,093 messages. Similarly, the 2014 version of the SemEval Twitter dataset contains 1,853 user tweets and 1,142 live journal datasets [15]. The 2016 and 2017 versions of the SemEval datasets were split into training, development, and testing data for each subtask in the competition. There were five A, B, C, D, and E subtasks in this edition [16]. In addition to the SemEval efforts, Korean, German, and Indonesian languages have also been investigated for sentiment analysis. A Korean dataset was created (KOSAC) that includes 332 news articles. The study's primary aim was to examine sentiments in Korean; Korean Subjectivity Markup Language was used as an annotator [17]. Another dataset was developed by [18], which contains customer reviews about various Amazon products. Amazon review parser1 was used for data collection. Human experts annotated each review according to their semantic meaning. A total of 63,067 reviews were collected about different products. Another effort was made to develop an Indonesian tweet corpus. The Twitter Streaming API was used to collect tweets. To clean the data into Indonesian dialect tweets, geo location was used. The Indonesian dataset contains 5.3 million tweets [19].

B. SENTIMENT ANALYSIS USING DEEP LEARNING AND WORD EMBEDDINGS MODELS

Recently, a deep learning method was implemented to investigate text representation and to overcome the problem of sentiment classification on a large social network corpus [34],

[35], and [36]. A unique new method, improved word vectors (IWVs), was recommended for word embedding in the domain of sentiment analysis [37].

A study [38] was performed on the sentiment analysis of social network data on the subject to support intelligent transportation systems. Data were gathered from various social networking (Facebook, Twitter, TripAdvisor) websites. After removing irrelevant data, they applied a word embedding approach for sentiment analysis, and their system achieved 93% accuracy.

Another research study [39] used word embedding techniques with deep learning models for intelligent healthcare monitoring of wearable sensors and social networking data. A Bi-LSTM deep learning model was used to classify the data to predict abnormal patient conditions and drug side effects.

In another research study [40], based on social network data, a real-time observation framework was suggested to detect traffic accidents and analyze traffic conditions by using Bi-LSTM. In the first step, data were collected from social networks by a query-based engine. Finally, the FastText word embedding technique and Bi-LSTM with softmax were trained. The proposed system achieved 97% accuracy for traffic event detection analysis.

C. URDU DATASETS SENTIMENT ANALYSIS

Although a considerable quantity of data is available on internet research on sentiment analysis, Urdu is still at the initial level compared to other resource-rich languages such as English. A large quantity of data is required to create a benchmark dataset for sentiment analysis. The drawbacks of existing corpora are that they are too small or contains data about limited genres.

In this [20] study, the authors collected user reviews to create two corpora to find their model's efficiency. The first corpus contains 322 positive and 328 negative movie reviews. The second corpus contains reviews about electronic appliances. This dataset contains 650 user reviews, among which 322 are positive and 328 are negative. In this study, the grammatical-based approach was used and focused on sentence grammatical structure. Their model achieved 82.5 percent accuracy. The authors did not mention any data annotating technique. The corpus used in study [20] is not publicly available. To use machine learning applications, a large quantity of data is required for training purposes, but very little data is used in this study.

Another study was performed on Urdu sentiment analysis. The authors of the research study [21] extracted Urdu text from Urdu news channel websites, e.g., bbcurdu.com, downnnews.com, blog.jang.com.pk, on a particular topic for corpus generation. The authors used a lexicon-based architecture and assigned polarity to each token according to its sentiment. To find the model efficiency, they performed experiments only on 124 Urdu comments, which they extracted from different websites. The lexicon-based model reveals an overall accuracy of 66 percent.

The most significant effort to build an Urdu sentiment analysis corpus was made by the authors of study [22]. This study method began with a collection of Urdu blogs of different genres. A total of 6,025 Urdu sentences were gathered from 151 different blogs. Three human experts annotated these collected sentences into positive, negative, and neutral classes. After applying basic natural language preprocessing tasks such as stop word removal, the authors used Lib-SVM, decision tree (J48), and K-NN algorithms for classification. Out of the three machine learning classifiers, K-NN achieved the highest accuracy of 67.01 percent. The corpus used in study [22] is not publicly available.

Note that Urdu is a resource-deprived language, linguistically and technically. According to the existing literature, many of the procedures applicable to sentiment analysis of other languages are not relevant to the Urdu language due to morphological structure [41], [42].

Additionally, the deficiency in linguistic and language resources such as lexicons and corpora also makes it difficult to implement the currently available sentiment analysis methods cited in the literature review, such as the availability of lexicons and datasets.

Moreover, accessible annotated datasets are not sufficient for implementing useful sentiment analysis. In addition, datasets and sentences generally belong to the same or limited genres. To reduce this deficiency, this study emphasizes building an Urdu dataset containing sentences fitted to six different domains. To execute sentiment analysis, we implemented machine learning and deep learning models on our constructed corpus UCSA-20, which has not yet been studied fully for the sentiment analysis of Urdu data.

III. DATASET GENERATION

This section describes the token procedures to create an annotated Urdu dataset for sentiment analysis. The stages included for building the Urdu corpus are collecting user reviews from the internet, preparation of annotation rules, manual annotation, and final version of the corpus.

A. COLLECTING REVIEWS FROM THE INTERNET

To build a benchmark dataset for Urdu sentiment analysis, user reviews contain information about various services, products, games, and politics from different websites that allow users to post their Urdu views. Urdu is a resource-deprived language; therefore, the authors decided to collect data about different genres from internet repositories that are easily accessible to construct a standard Urdu language text corpus. Consumer reviews contain information about politics, movies, Urdu drama, TV talk shows and sports. Four individuals were hired for manual data collection. They were native Urdu speakers. They were aware of the objective. They took 3 months for data collection. Initially, data was gathered in an Excel sheet.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

B. ANNOTATIONS RULES

This section explains the annotation process that the authors used in manual corpus generation. This step includes preparing the rules or guidelines for annotation, manual annotation of the complete dataset by native Urdu speakers, and IAA (inter annotation agreement). We design rules for sentiment analysis from different currently available datasets prior to corpus annotation. TABLE 1 shows examples of user reviews fitting to the positive and negative classes.

TABLE 1
EXAMPLES OF USER REVIEWS SUITABLE TO POSITIVE AND NEGATIVE CLASSES

Negative Review Examples	Positive Review Examples
سافٹ ویئر قابل عمل نہیں ہے (ye software kabil-e-amal nahi hai) This software is not workable	یہ خوش کن ہے (ye Khush kun hai) It is nice
demagnetizing factor	کوشش کی ہے کہ یہ نسخہ بہت اچھا ہے (Main ne koshish ki hai ye nuskha baohut acha hai)
مجھے ایسے لوگوں سے نفرت ہے... ایسے لوگوں سے نفرت ہے & میں نے (Mujhy eisay lougon se nafrat hai ….eisay lougon se nafrat hai) I hate such people... I hate such people	I have tried this recipe, it is very good

1. A sentence is labeled as positive if it conveys an overall positive sentiment [23].
2. A sentenced is marked as positive if it expresses both positive and neutral.
3. The sentence is classified as positive if it contains agreement approval [24].
4. Sentences with words such as congratulations and admiration were marked as positive [24].
5. A sentence is labeled as negative if it conveys an overall negative sentiment [25].
6. A review or sentence is considered negative if it has more negative words than the other sentiments.
7. If any sentence shows any disagreement, then the sentence is classified as negative [24].
8. If a sentence has terms such as ban, penalizing, and assessing, it is labeled negative [24].
9. If a sentence comprises a negative word with a positive adjective, it is classified as negative [26].

C. DATASET FEATURES

The Urdu dataset was manually annotated by three (X, Y, and Z) human experts to create a benchmark dataset hereafter named Urdu Corpora for Sentiment Analysis (UCSA-20). Native Urdu speakers annotated all the user reviews, and all were master graduates in the Urdu language. The annotators were aware of sentiment analysis and annotation rules, as discussed above. Experts X and Y annotated each sentence either in positive or negative classes, considering the above-discussed rules. The conflicts between X and Y were resolved by Z by labeling the review. We obtained an IAA of 73.91 percent and a Cohen Kappa score of 0.597 (moderate) for the whole UCSA-20. IAA and moderate scores revealed that the annotators followed the annotation guidelines during the labeling phase. UCSA-20 contains 9,601 user reviews, of which 4,843 are positive and the remaining are negative reviews, as shown in TABLE 2. From the statistics in TABLE 3, it can be clearly seen that our corpus is class balanced. Very few scholars in the existing literature have made efforts to create datasets for carrying out experiments. Nevertheless, unluckily, most of the currently available datasets are very small and are from specific genres or cover very few genres rather than different genres. The corpora [20], [25] are small and contain user reviews to specific fields.

TABLE 2
DATASET STATISTICS

Dataset Properties	Stats
Total number of reviews	9,601
Total number of negative reviews	4,758
Total number of positive reviews	4,843
Total number of tokens	1,141,716
Minimum length of review	1 word
Average tokens per review	118.91

IV. METHODOLOGY

This section focuses on the experimental details of our machine learning and deep learning models, the support vector machine (SVM) model, naïve Bayes, random forest, AdaBoost, multilayer perceptron (MLP), logistic regression, LSTM and CNN1D. All these machine and deep learning models have been implemented on our proposed UCSA-20 corpus. FIGURE 1 represents the overall architecture of the system.

A. TEXT CLEANING AND PREPOSSESSING

The preprocessing of Urdu text is essential to make it easy and useful for NLP tasks. To enhance our model's accuracy, emoji's, URLs, email addresses, phone numbers, numerical numbers, numerical digits, currency symbols, and punctuation marks were replaced with a special character. Additionally, the following text preprocessing steps were performed to increase our model's effectiveness for Urdu text.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

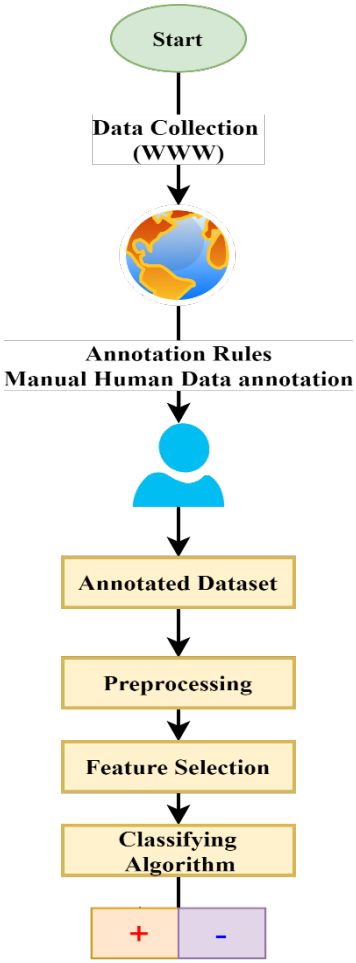

FIGURE 1. High-Level System Architecture for Urdu Sentiment Analysis.

B. REMOVING URDU STOP WORDS

The words used to complete sentences are called stop words. Words such as are (ہیں), and we (ہم) are commonly used words that are removed from the corpus. Similar to the other NLP tasks, sentiment analysis also requires some preprocessing steps to accomplish the task. Like other natural languages, the Urdu language has stop words. Nevertheless, due to the Urdu language's morphological structure and poor resources, it is challenging to remove stop words automatically. The following diagram (FIGURE 2) explains the flowchart of the Urdu stop word removal step. All commonly used Urdu stop words were collected in a file, and then all those were eliminated from the corpus.

C. TOKENIZATION

Tokenization of Urdu text is equally useful as for any other language to solve any NLP-related tasks. Tokenization is a process to separate sentences from each other and separate each word from sentences

D. NORMALIZATION OF URDU TEXT

The normalization of Urdu text is essential to make it advantageous for NLP-related tasks. This step is used to solve the issue of correct encoding for the Urdu characters.

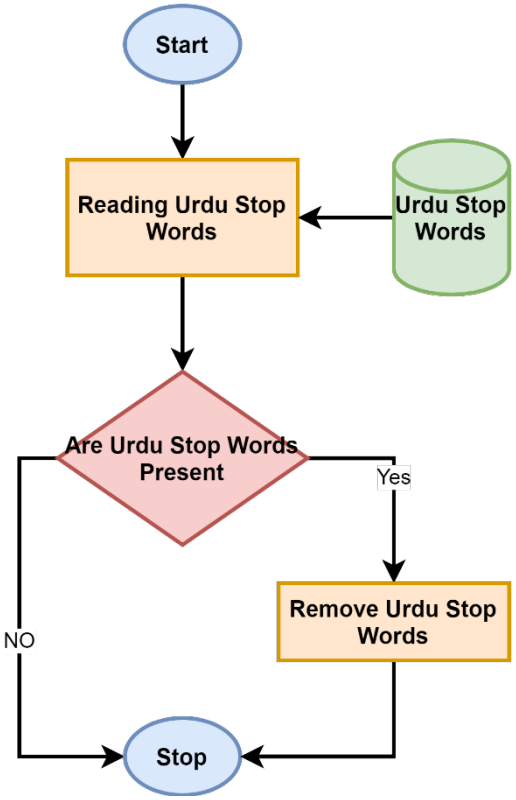

FIGURE 2. Flowchart of the Urdu Stop Word Removal Module.

Normalization is used to obtain all the characters in the required Unicode range (0600-06FF) for Urdu text. This step is also used to avoid the concatenation of different Urdu words. For example, "خوش باش" (khush bash, happy) is one word (unigram) with two different strings. These two strings (khush and bash) are part of the same word concerning syntax and semantics. If the space between two strings is omitted, then we obtain "خوشباش" (khushbash), which is an incorrect word in the Urdu language. With the help of normalization, authors attempt to minimize this effect.

E. FEATURE SELECTION

In natural language processing tasks such as text classification, the text is generally denoted as a vector of weighted features. In this study, different n-gram models are used; these are the models that allocate probabilities to a sequence of words. An n-gram is a sequence of N words; a unigram is a model that contains a sequence of one word such as "homework"; similarly, a bigram is a sequence of two words such as "your homework," and a 3-gram or trigram model contains a sequence of three words such as "complete your homework." A uni-bigram, and uni-trigram are combinations of words. In this paper, unigram, bigram, trigram, uni-bigram, and uni-trigram feature models are used.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

F. WORD EMBEDDING

Recently, pre-trained word vector models have been applied in many natural processing tasks and have shown state-of-the-art results. The basic concept behind these pre-trained models is to train these models on very large corpora and fine tune these models for specific tasks. FastText [26] is a word vector model trained on Wikipedia and common crawl datasets. This model is trained for a total of 157 languages, including Urdu. This is the motive behind using the FastText word embedding model for this task with deep learning models. The FastText model was trained using skip-gram [27] and continuous bag-of-words (CBOW) [28]. CBOW and skip-gram are both trained using a single hidden layer. The CBOW model is used to guess the central word in a given circumstance. The skip-gram model objective is the opposite of CBOW. The FastText model is an extension of skip-gram that breaks down the unigram (words) into bags of character n-grams (sub-words) and allocates a vector value to individual character n-grams. Therefore, each single word is represented by the summation of its related n-gram vectors. Additionally, the FastText word embedding model with n-gram characters can also give vector values for those words that are not available in the training corpus.

G. CLASSIFICATION MODELS

Various machine learning algorithms with different features and deep learning models with FastText pre-trained model, namely, RF, naïve Bayes, SVM, AdaBoost, MLP, logistic regression, CNN1D, and long short-term memory (LSTM), are used to find the effectiveness of our corpus and achieve state-of-the-art results. We do not explain these conventional machine learning models here because these models are prevalent and famous. LSTM [29] is an accepted recurrent neural network architecture and shows state-of-the-art results for sequential data. Basically, LSTM is designed to capture the long-term dependencies between text data. For each time step, the LSTM model obtains the input from the current word, and the output from the previous or last word produces an output, which is used to feed to the next state. The hidden layer from the previous state (and sometimes all hidden layers) is then used for classification. The high-level system architecture of an LSTM network with FastText embedding is shown in FIGURE 3. A typical LSTM network contains four main components: input gate, forget gate, memory cell, and output gate. Basically, these gates are used to flow in and out of the data at the existing time step. More formally, these cells are explained in equations 1, 2, 3, 4 and 5.

$$f_t = \sigma\left(w_f[x_t, h_{t-1}] + b_f\right) \quad (1)$$

$$I_t = \sigma(w_I[x_t, h_{t-1}] + b_I) \quad (2)$$

$$O_t = \sigma(w_O[x_t, h_{t-1}] + b_O) \quad (3)$$

$$C_t = f_t * C_{t-1} + I_t * \tanh(W_C[x_t, h_{t-1}] + b_C) \quad (4)$$

$$h_t = O_t * \tanh(C_t) \quad (5)$$

In the above equations, f_t is the forget gate, I_t is the input gate, and O_t represents the output gate, where σ represents the sigmoid function and tanh denotes the hyperbolic tangent function. w_f, w_I and w_O, b_f, b_I, b_O are the weight matrix and bias scalars that are assessed during model training. h_t Denotes the hidden layer.

Convolutional neural networks are primarily used in the computer vision domain. However, a 1-dimensional CNN is mainly used in the sequential processing of text data. A 1-d CNN was used by [30] for semantic role labeling, [31] used a 1-d CNN for sentiment classification similarly, and [32] used a 1-d CNN for question type classification.

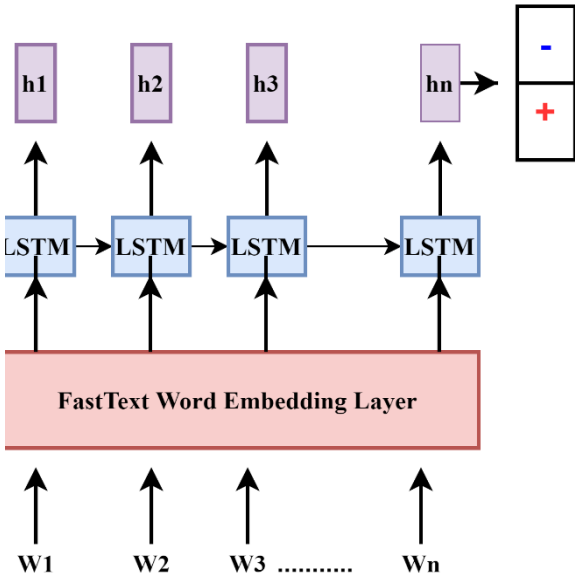

FIGURE 3. Proposed System Architecture of an LSTM Network with a FastText Word Embedding Layer for Urdu Sentiment Classification.

V. EXPERIMENTS AND RESULTS

We performed our experiments on UCSA-20, which is publicly available to the research community. UCSA-20 contains 9,601 Urdu reviews, which belong to political, dramas, movies, TV talk shows, sports, and software domains. These views were manually collected from different websites through the internet. Every review in UCSA-20 falls into a positive class, denoted by 1, or into a negative class, denoted by 0. The dataset is split into training, which contains 80 percent of user reviews, and testing, which contains 20 percent. A Python NLP library named UrduHack is used for preprocessing tasks.

A. EVOLUTION METHODS

Overall, four performance metrics, testing accuracy, precision, recall, and F1 measure, were used to evaluate our model. Overall accuracy is measured as accurately categorized subjects divided by the total number of subjects, as explained in the following equations.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}} \quad (6)$$

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (7)$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (8)$$

$$\text{F1 Score} = 2 \times \frac{Precision \times recall}{Precision + recall} \quad (9)$$

B. ANALYSIS OF RESULTS

Each of the six machine learning classifiers is run on the UCSA-20 dataset using unigram, uni-bigram, uni-trigram, bigram, and trigram features discretely. All the revealed results are carefully examined to improve the results and identify the finest machine learning classifier with features that achieve better results than the other five concerning the accuracy, precision, recall, and F1 score. By witnessing the TABLE 3 results, all the machine learning classifiers' performances are quite poor with the trigram feature. Generally, there are discriminative models (SVM, logistic regression, etc.) and generative classification models (naïve Bayes).

Both SVM and logistic regression achieve satisfactory results, as both classifiers belong to discriminative models. Logistic regression is a supervised machine learning algorithm that is used when problems are categorical in nature. Logistic regression is the most commonly used classifier when the data in question have two classes, either positive or negative.

Logistic regression is a very efficient and effective supervised machine learning classification algorithm, so it is used for many binary classification problems. This is the reason why out of all six machine learning classifiers, the logistic regression achieves the highest accuracy of 81.94% with the trigram feature. Overall, the highest accuracy of 81.94%, precision of 79.95%, recall of 84.26%, and F1 score of 82.05% were achieved by logistic regression with the uni-trigram feature model. The SVM classifier achieved the 2nd highest accuracy, precision, recall and F1 score, which were 81.47%, 80.32%, 82.36%, and 81.47%, respectively, with the unigram feature model.

The worst accuracy out of all classifiers was 55.25% gain by random forest with trigram features. All classifiers show better performance with the bigram feature model than with the trigram feature model. Again, logistic regressions achieve the highest accuracy of 72.32% with the bigram feature model. The overall results using different machine learning models with different features are explained in TABLE 3.

The mean square error (MSE) is used as a loss function in CNN1D and LSTM. Adam is used as an optimizer function for both deep learning algorithms. The number of epochs was set to 25. According to the deep learning results, the models are shown in TABLE 4. LSTM achieves slightly better results than the CNN1D model in terms of accuracy, which is 75.96 for LSTM with FastText and 0.7573 for CNN1D. LSTM performs better only because it is an accepted recurrent neural network for sequential data.

As previously stated, a lack of research using machine learning algorithms in Urdu sentiment analysis is seen. Very few studies are found regarding this context. These authors have used different machine learning classifiers on a very insignificant dataset. Our dataset contains more user reviews as a camper to previous studies. The results of our study reveal that each model in our study performs better than existing models. A comparison of our study with existing studies is presented in TABLE 5.

TABLE 3
OVERALL ACCURACY PRECISION, RECALL AND F1 SCORE OF DIFFERENT MACHINE LEARNING MODELS WITH DIFFERENT FEATURES

Model Name	Feature	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
Random Forest	unigram	0.7477	0.7128	0.8099	0.7475
	uni-bigram	0.7485	0.7136	0.813	0.74857
	uni-trigram	**0.7527**	0.7136	0.8134	0.74857
	bigram	0.6564	0.6004	0.8933	0.6564
	trigram	0.5525	0.524	0.9419	0.5525
Naïve Bayes	unigram	0.7723	0.7547	0.79303	0.7723
	uni-bigram	**0.7899**	0.768	0.8183	0.7899
	uni-trigram	0.7894	0.7641	0.8247	0.7894
	bigram	0.6983	0.6753	0.7402	0.6983
	trigram	0.6037	0.5674	0.8035	0.6037
SVM	unigram	0.8147	0.8032	0.8236	0.8147
	uni-bigram	0.8116	0.7965	0.8268	0.8116
	uni-trigram	**0.8153**	0.8016	0.8278	0.8153
	bigram	0.7113	0.6927	0.7381	0.7113
	trigram	0.6078	0.5729	0.7835	0.6078
AdaBoost	unigram	0.7837	0.7729	0.7909	0.7818
	uni-bigram	0.7909	0.7723	0.7723	0.7921
	uni-trigram	**0.7982**	0.7798	0.8194	0.7991
	bigram	0.6958	0.6615	0.7761	0.7142
	trigram	0.5835	0.5449	0.9028	0.6796
Multilayer Perceptron (MLP)	unigram	0.7972	0.7905	0.7972	0.7939
	uni-bigram	0.8034	0.7943	0.8078	0.8013
	uni-trigram	**0.8044**	0.8036	0.7951	0.7993
	bigram	0.6564	0.6004	0.8933	0.6564
	trigram	0.6083	0.5768	0.7529	0.6532
Logistic Regression	unigram	0.4868	0.4874	0.9197	0.6371
	uni-bigram	0.8184	0.7985	0.8416	0.8195
	uni-trigram	**0.8194**	0.7995	0.8426	0.8205
	bigram	0.7232	0.6939	0.7782	0.7336
	trigram	0.6223	0.5839	0.7972	0.6741

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

FIGUREs 4, 5, 6, 7 and 8 describe the comparison of each model in terms of accuracy, precision, recall and F1 measure with selected n-gram features.

TABLE 4
DEEP LEARNING MODELS RESULTS WITH FASTTEXT

Model Name	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
LSTM with FastText	0.7596	0.7776	0.7783	0.7752
CNN1D with FastText	0.7573	0.7843	0.7678	0.7557

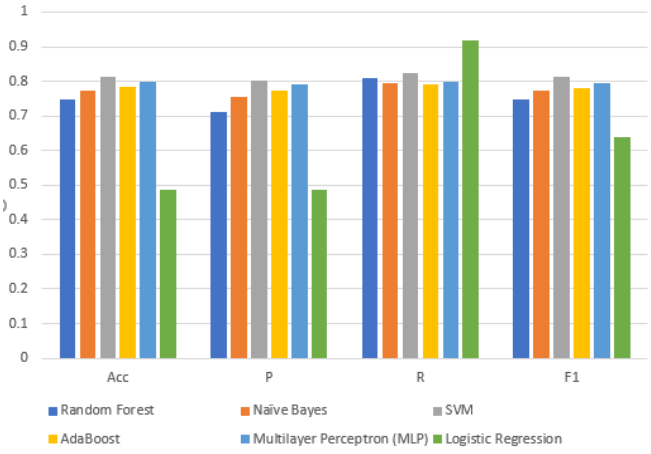

FIGURE 4. Performance Comparison of all Classifiers with Unigram Features.

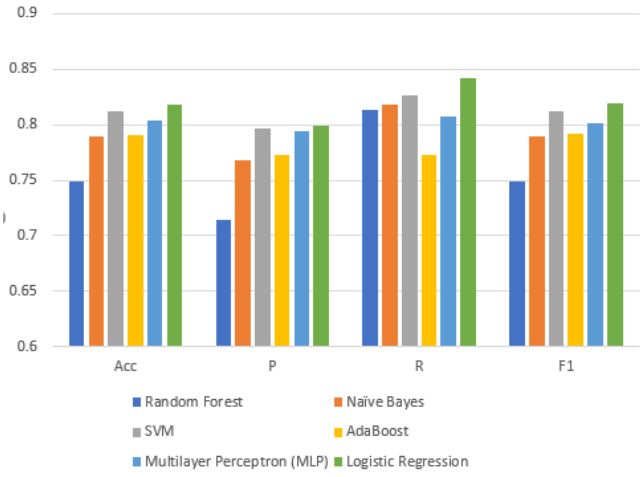

FIGURE 5. Performance Comparison of all Classifiers with Uni-Bigram features.

TABLE 5
COMPARISON WITH EXISTING MODELS

Reference	Model Name	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
[33]	k-NN	67.0185	0.674	0.6703	0.6703
[33]	Lib SVM	65	0.6743	0.65	0.6457
[19]	Lexicon-Based	0.66	0.69	0.79	0.73
Proposed study	SVM	0.8147	0.8032	0.8236	0.8147
Proposed study	Random Forest	0.7527	0.7136	0.813	0.74857
Proposed study	Naïve bays	0.7899	0.768	0.8183	0.7899
Proposed study	AdaBoost	0.7982	0.7798	0.8194	0.7991
Proposed study	MLP	0.8044	0.8036	0.7951	0.7993
Proposed study	Logistic Regression	0.8194	0.7995	0.8426	0.8205
Proposed study	CNN1D +FastText	0.7573	0.7843	0.7678	0.7557
Proposed study	LSTM +FastText	0.7596	0.7776	0.7783	0.7752

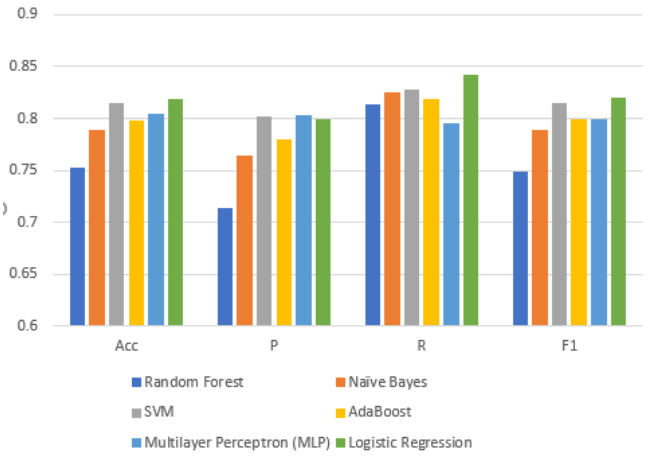

FIGURE 6. Performance Comparison of all Classifiers with Uni-Trigram Features.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3093078, IEEE Access

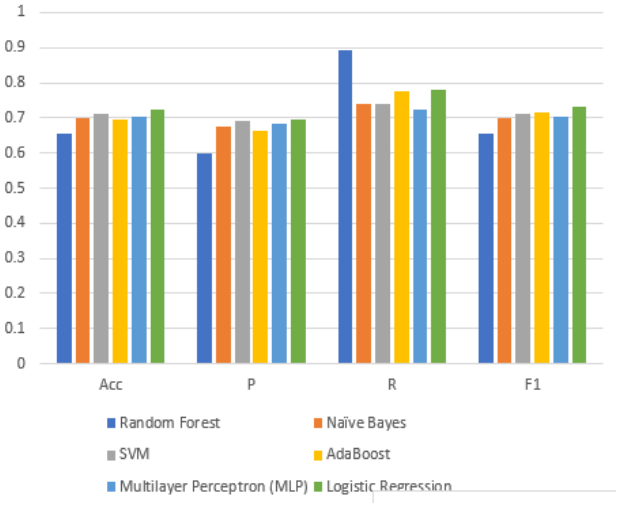

FIGURE 7. Performance Comparison of all Classifiers with Bigram Feature.

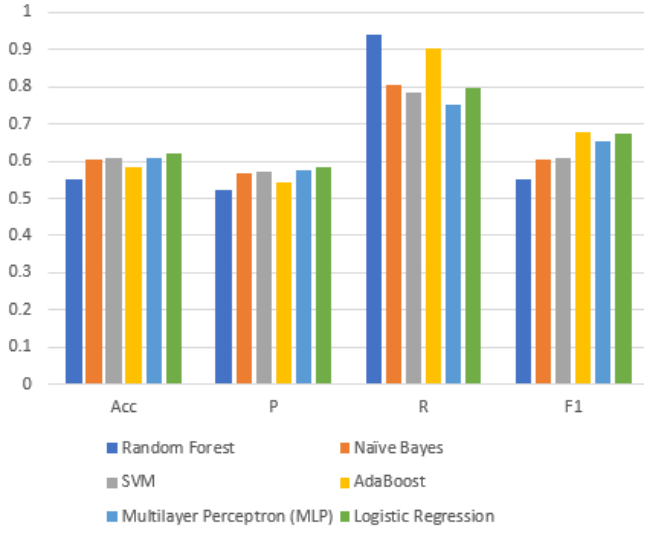

FIGURE 8. Performance Comparison of all Classifiers with Trigram Features.

VII. CONCLUSION

Few research studies have been reported in the Urdu sentiment analysis domain. In this paper, high classification accuracy has been achieved on different machine learning and deep learning models. After performing various experiments based on different N-gram feature models, logistic regression achieved the highest performance in accuracy, precision, recall, and F1 score. The SVM classifier is the 2nd highest performer, but its average performance is better than all other classifiers.

This study exposed a new domain for future research using machine learning and deep learning to build models for resource-deprived languages. One of the limitations of this study is that it includes only positive and negative classes; our future work will include a neutral class in our dataset. Room still exists for enhancing the results of the above machine learning classifiers. In the future, some deep learning models with word embedding techniques such as BERT, GPT, and ELMO can be applied to achieve the highest performance in terms of accuracy, precision, recall, and f-measures.

REFERENCES

[1] S.U.Hassan, "Identifying important citations using contextual informationfrom full text," in ACM/IEEE Joint Conference on Digital Libraries(JCDL), Toronto, ON, Canada,2017, pp. 1–8.
[2] Liu, "Identifying social roles using heterogeneous features in online socialnetworks," Journal of the Association for Information Science and Tech-nology., vol. 70, no. 7, pp. 660–674, Jan. 2019.
[3] Luo, "Knowledge empowered prominent aspect extraction from productreviews," Information Processing & Management. vol. 56, no. 3, pp. 408–423, May.2019.
[4] F.Anwaar, "A hybrid framework to integrate content embeddings in recom-mender systems for cold start items," Journal of computational science.,vol. 13, pp. 9–18, Nov.2018.
[5] R.Nawaz, "Identification of Manner in Bio-Events," Lrec., vol.6, pp.3505–3510, May.2012.
[6] H.Qadir, "An optimal ride sharing recommendation framework for car-pooling services," IEEE Access., vol. 13, pp. 62296–62313, Oct.2018.
[7] M.Z. Asghar, "Creating sentiment lexicon for sentiment analysis in Urdu:The case of a resource-poor language," Expert Systems., vol. 36, no. 3, pp.e12397, Jun.2019.
[8] A. Z.Syed and M.Aslam, "Lexicon based sentiment analysis of Urdutext using SentiUnits," in Mexican International Conference on Artificial Intelligence, Berlin, Heidelberg, Germany, 2010, pp.32–43..
[9] W.Ijaz and S.Hussain, "Corpus based Urdu lexicon development," in theProceedings of Conference on Language Technology (CLT07), Peshawar, University of Peshawar, Pakistan, 2007, pp.1–12.
[10] W Anwar, X Wang, "A Survey of Automatic Urdu language processing, "in International Conference on Machine Learning and Cybernetics, New Jersey, USA, 2006, pp.4489–4494.
[11] A.Daud, W.Khan and D.Che, "Urdu language processing: a survey," in Artificial Intelligence Review., vol. 47, no.3, pp.279–311, Mar.2017.
[12] S.Kiritchenko,S., Mohammad, and M.Salameh, "Semeval-2016 task 7:Determining sentiment intensity of English and Arabic phrases," In Proceedings of the 10th international workshop on semantic evaluation(SEMEVAL-2016) ., vol. 47, no.3, ,pp.42–51, Jun.2016.
[13] J. Villena-Román, J.García-Morera, "DAEDALUS at SemEval-2014 Task9: comparing approaches for sentiment analysis in Twitter," in In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval2014) ., vol. 47, no.3, ,pp.218–222, Aug.2014.VOLUME 4, 2016
[14] P.Nakov, "SemEval-2016 task 4: Sentiment analysis in Twitter," in arXivpreprint arXiv:1912.01973 ., Dec.2019.
[15] H.Jang,,M.Kim, and H.Shin, "KOSAC: A full-fledged Korean sentiment analysis corpus," in Proceedings of the 27th Pacific Asia Conference on Language, Information, and Computation (PACLIC 27), Taipei , Taiwan, Nov. 2013, pp.366–373.
[16] L.S.Chen, "A neural network-based approach for sentiment classification in the blogosphere," Journal of Informatics., vol. 5, no. 2, pp. 313–322,April.2011.
[17] A.F. Wicaksono, C.Vania, and B.Distiawan, "Automatically building a10 corpus for sentiment analysis on Indonesian tweets," in Proceedings of the28th Pacific Asia Conference on Language,

Information and Computing, Phuket, Thailand, Dec. 2014, pp.185–194.

[18] A.Z.Syed, M.Aslam and A.M.Martinez-Enriquez, "Associating targetswith SentiUnits: a step forward in sentiment analysis of Urdu text," in Artificial Intelligence Review ., vol. 41, no.4, ,pp.535–561, Feb.2014.

[19] Z.U.Rehman, and I.S.Bajwa, "Lexicon-based sentiment analysis for Urdulanguage," in sixth international conference on innovative computing technology (INTECH), Dublin, Ireland, Aug. 2016, pp.497–501.

[20] N.Mukhtar, "Urdu sentiment analysis using supervised machine learning approach," International Journal of Pattern Recognition and Artificial Intelligence., vol. 32, no. 2, pp. 1851001, July.2017.

[21] M.Pontiki, D.Galanis, H.Papageorgiou, "Semeval-2016 task 5: Aspect based sentiment analysis," in Proceedings of the 10th international work-shop on semantic evaluation (SemEval-2016), San Diego, California, USA, Jun. 2016, pp.19—30.

[22] A.Mageed, and M.T.Diab,"A Multi-Genre Corpus for Modern Standard Arabic Subjectivity and Sentiment Analysis," LREC., vol. 515, pp. 3907–3914, July.2012.

[23] D.Maynard and K.Bontcheva, "Challenges of evaluating sentiment anal-ysis tools on social media," Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), Portorož,Slovenia, May.2016, pp.1142—1148.

[24] M.Ganapathibhotla, and B.Liu, "Mining opinions in comparative sentences," Proceedings of the 22nd International Conference on Computational Linguistics, Manchester, UK, Aug.2008, pp.241—248.

[25] Z. U.Rehman., and I.S. Bajwa, "Lexicon-based sentiment analysis for Urdu language," sixth international conference on innovative computing technology (INTECH), Dublin, Ireland, Aug.2016, pp.497–501.

[26] E.Grave, "Learning word vectors for 157 languages," in arXiv preprintarXiv:1802.06893 ., Mar.2018.

[27] Bojanowski and Piotr, "Enriching word vectors with sub word information. "Transactions of the Association for Computational Linguistics Elect. Eng. Res. Lab., MIT. Massachusetts, USA, Sep.2017, pp.135-146.

[28] T.Mikolov, I.Sutskever, K.Chen, "Distributed representations of words and phrases and their compositionality," Conference on Neural Information Processing Systems (NIPS), NEVADA, USA, Dec.2013, pp.3111-3119.

[29] S. Hochreiter, "Long short-term memory," Neural computation. vol. 9, no.8, pp. 1735–1780, Nov.1997.

[30] R Collobert, "Natural language processing (almost) from scratch," Journal of machine learning research. vol. 9, no. 8, pp. 2493–2537, Nov.1997.

[31] N.Kalchbrenner, "A convolutional neural network for modelling sentences," in arXiv preprint arXiv:1408.5882 ., Apr.2014.

[32] Y. Kim, "Convolutional neural networks for sentence classification," inarXiv preprint arXiv: 1408.5882., Sep.2018.

[33] J N.Mukhtar, "Urdu sentiment analysis using supervised machine learning approach," International Journal of Pattern Recognition and Artificial Intelligence., vol.32, no. 2, pp. 1851001, Feb.2018.

[34] Zhao, Wei, et al, "Weakly supervised deep embedding for product review sentiment analysis", IEEE Trans on Knowledge and Data Engineering., vol. 30, no.1, pp. 185-197, Sep. 2017.

[35] Kamkarhaghighi, Mehran, and Masoud Makrehchi. "Content tree word embedding for document representation." Expert Systems with Applications. vol. 30, no.90, pp. 241-249, December. 2017.

[36] Hu, Zhongkai, et al. "Review sentiment analysis based on deep learning", 12th International Conference on E-Business Engineering, BEIJING, CHINA, October 2015, pp. 87–94.

[37] Rezaeinia, Seyed Mahdi, Ali Ghodsi, and Rouhollah Rahmani. "Improving the accuracy of pre-trained word embeddings for sentiment analysis." In arXiv preprint arXiv: 1711.08609., Nov 2017.

[38] Ali F, Kwak D, Khan P, El-Sappagh S, Ali A, Ullah S, Kim KH, Kwak KS. Ali, Farman, et al. "Transportation sentiment analysis using word embedding and ontology-based topic modeling." in Knowledge-Based Systems., Vol. 174, pp.27-42, Jun 2019.

[39] Ali, Farman, et al. "An intelligent healthcare monitoring framework using wearable sensors and social networking data." in Future Generation Computer Systems., vol. 114, pp.23-43, Jan.2021.

[40] Ali, Farman, et al. "Traffic accident detection and condition analysis based on social networking data" in Accident Analysis & Prevention., vol. 151, pp. 105973, Mar. 2021.

[41] Masroor, Hafsa, et al. "Transtech: development of a novel translator for Roman Urdu to English Heliyon., vol. " In 5, e01780, May.2019.

[42] Rafae, Abdul, et al. "An unsupervised method for discovering lexical variations in Roman Urdu informal text." In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing., Lisbon, Portugal, Sep.2015, pp. 823-828.

LAL KHAN was born in D G Khan, Punjab, Pakistan in 1990. He is a Ph.D. scholar in the Department of Computer Science and Information Engineering, Chang Gung University, Taiwan. He received a M.S. Degree in computer science from the Federal Urdu University of Arts, Science and Technology, Islamabad in 2017. His research interests include machine learning, deep learning, natural language processing (NLP) and speech recognition. He is currently working in NLP task for resource-deprived languages.

AMMAR AMJAD received a master's degree in computer science from the National College of Business Administration and Economics, in March 2017. He is currently pursuing the Ph.D. degree in electrical engineering with the Division of Computer Science and Information Engineering, Chang Gung University, Taiwan. His main research interests include speech processing, language learning, speech analysis, speech synthesis, voice pathologies, auditory neuroscience and machine learning.

Hsien-Tsung Chang Hsien-Tsung Chang obtained his M.S. and Ph.D. degrees in the Department of Computer Science and Information (CSIE) from National Chung Cheng University in July 2000 and July 2007, respectively. He joined the Faculty of Computer Science and Information Engineering Department at Chang Gung University and served as an associate professor. He is also a member of the Artificial Intelligence Research Center at Chang Gung University and the Department of Physical Medicine and Rehabilitation at Chang Gung Memorial Hospital. Prof. Chang's research areas focus on artificial intelligence, natural language processing, information retrieval, big data, web services, and search engines. Prof. Chang is the Director of the Web Information and Data Engineering Laboratory (WIDE Lab).

Noman Ashraf received his master's degree in computer science from the National University of Computer and Emerging Sciences, Islamabad, Pakistan in 2017. He is a Ph.D. student at the IPN Computing Research Center, Mexico, and his specialization area are natural language processing. His main research interests are natural language processing, machine learning and deep learning.

Alexander Gelbukh is a Research Professor and Head of the Natural Language Processing Laboratory of the Center for Computing Research of

the Instituto Polit\'ecnico Nacional, Mexico, and Honorary Professor of the Amity University, India. He is a member of the Mexican Academy of Sciences, founding member of the Mexican Academy of Computing, and National Researcher of Mexico (SNI) at excellence level 3 (highest). He is an author or co-author of more than 500 publications in computational linguistics, natural language processing, and artificial intelligence, recently with a focus on sentiment analysis and opinion mining. He is an editor-in-chief, associate editor, or member of editorial board for more than 20 international journals, and he has been a chair or program committee chair of over 50 international conferences.