Contextualized Satire Detection in Short Texts Using Deep Learning Techniques

Ashraf Kamal[1], Muhammad Abulaish[2,*] and Jahiruddin[3]

[1]*PayPal, Chennai–600119, India*
[2]*Department of Computer Science, South Asian University, New Delhi–110068, India*
[3]*Department of Computer Science, Jamia Millia Islamia (A Central University), New Delhi–110025, India*
E-mail: ashrafkamal.mca@gmail.com; abulaish@sau.ac.in; jahiruddin@jmi.ac.in
[*]*Corresponding Author*

Received 18 November 2023; Accepted 27 January 2024;
Publication 27 March 2024

Abstract

Satire is prominent in user-generated content on various online platforms in the form of satirical news, customer reviews, blogs, articles, and short messages that are typically of an informal nature. As satire is also used to disseminate false information on the Internet, its computational detection has become a well-known issue. Existing work focuses primarily on formal document- or sentence-level textual data, whereas informal short texts have gotten less attention for satire detection. This paper presents a new model called BiLSTM self-attention (BiSAT) for detecting satire in informal short texts. It consists of various components such as input, embedding, self-attention, and two bi-directional long short-term memory (BiLSTM) layers for learning crucial contextual information pertaining to the satire present in the texts. The input layer uses the text as input to create an input vector, which is then given to the embedding layer to create the appropriate numeric

Journal of Web Engineering, Vol. 23_1, 27–52.
doi: 10.13052/jwe1540-9589.2312

vector. The output of the embedding layer is passed on to the first BiLSTM layer, which extracts contextual information-based sequences in the opposite direction. Between the first and second BiLSTM layers, a self-attention layer is employed to draw attention to the important satirical information that is acquired by the hidden layer of the first BiLSTM. The BiSAT model also takes a classic feature engineering approach, employing a 13-dimensional auxiliary feature vector comprised of features from four separate feature categories: *sentiment*, *punctuation*, *hyperbole*, and *affective*. The proposed BiSAT model is empirically evaluated on two benchmark datasets and a newly created dataset called Satire-280. It outperforms existing research and baseline methods by a significant margin. The Satire-280 dataset along with code can be downloaded from GitHub repository: https://github.com/Ashraf-Kamal/Satire-Detection.

Keywords: Information retrieval, online social media, figurative language detection, satire detection, deep learning.

1 Introduction

Online social media (OSM) is a leading source of short, informal, and unstructured data. The fast propagation of information and the high rise in the number of users over OSM are the key reasons behind the generation of such an overflowing amount of unstructured data [1]. As a result, OSM has become an important source for numerous applications, such as Web surveillance, product endorsement, digital marketing, recommendation systems, and prediction models. Moreover, over the last decade, figurative language (FL) has been used extensively on OSM platforms. One of the main reasons behind the tremendous expansion of FL is the availability of informal and non-literal components in user-generated unstructured data. Consequently, it causes FL proliferation in several forms of user-generated data, including customer reviews, news, blogs, and tweets. On the other hand, the computational linguistics research community has also shown remarkable interest in the automatic FL detection of FL, and it has become the primary task for the reliable functioning of the existing sentiment analysis and recommender system. If FL is not properly identified, then receiving high accuracy is a quite cumbersome task for such applications. Hence, the detection of satire, which is a particular category of FL, is a non-trivial, significant, and important research problem.

1.1 Satire in Online Social Media

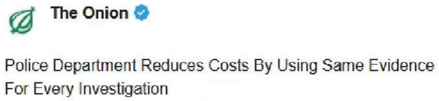
Satire is one of the prevalent FL categories, which is rapidly increasing in online user-generated data. Interestingly, it also includes other categories of FL (*irony* and *humor*). It is an important source to expose and highlight flaws in particular customs, policies, events, and legitimate orders [2]. It is defined as "a way of criticizing people or ideas humorously, or a piece of writing or play that uses this style".[1] Consider the satirical headline, "new law makes it legal for atheist doctors and nurses to refuse care to religious patients," taken from online satire news website '`The Science Post`'.[2] In this satirical headline, satire is applied in a humorous way to target doctors and nurses.

Interestingly, satire has been a well-known topic in psycholinguistics, philosophy, and cognitive science [2]. Usually, satirical posts are seen in indirect speech, wherein sophisticated content is used to convey implicit semantics. Nowadays, satire has been seen extensively over OSM platforms to propagate misinformation [3]. Mostly, satire is used in the form of news (*aka* satirical news) on the Internet. Satirical news belongs to a genre of deceptive news, and it is framed as legitimate news articles in online data, wherein ridicule and criticism remarks based on fictionalized stories are composed to target a specific entity [4]. However, it is quite different from fake news, in which fictionalized narratives are exhibited to mislead and harm people. It is framed with the intent that it seems *fake*, but in reality, it is not *fake* like fake news. The main purpose of the satirical news is to get the readers' attention towards the news as true.

Expansion of satire can seen on a large scale in customer reviews, blogs, and micro-blogging platforms like `Twitter` [2]. Therefore, satire detection on the Internet seems important to control the overall expansion of false stories across online platforms [4]. It is highly beneficial to numerous areas, such as affective computing and sentiment analysis where an in-depth knowledge of the metaphorical attributes of language is required. It is also necessary to establish a strongly connected human–computer interaction. It assists in enhancing the process of computers to understand and respond the human emotional factors [5].

[1] https://dictionary.cambridge.org/dictionary/english/satire (last accessed on 31 December 2023).

[2] https://bit.ly/2tF6ElN (last accessed on 31 December 2023).



The Onion

Police Department Reduces Costs By Using Same Evidence For Every Investigation

Figure 1 A satirical tweet posted from the official `Twitter` account of `The Onion`.

1.2 Satire on Twitter

Since its inception, `Twitter` has become an important fact-finding source for people across the globe [28, 29]. Users share and receive a broad range of ideas and information in the form of tweets in a short bandwidth of only 280 characters. Tweets are comprised of various informal and non-literal data [26, 27]. As a result, figurative language, such as *sarcasm*, *humor*, and *satire* are embedded within them. Users consider such FL categories in tweets to express their feelings, emotions, attitudes, or evaluations on a specific target in a non-literal fashion. In the last few years, tweets have become an important source to propagate satire. An enormous number of satirical tweets are published in a huge volume very rapidly. In addition, various official satirical `Twitter` accounts are available which generate satirical data on any ongoing events [6]. For example, Figure 1 presents a satirical tweet posted from the official `Twitter` account of (`The Onion`) (a popular online satirical newspaper). In this example, there is a satirical dig on the police department, exposing their investigation procedure and collected evidence. Moreover, satirical tweets also include deceptive and misinformation, and that originates a step towards generating rumors on the Internet [4]. Hence, research towards the computational detection of deceptive and misinformation in the form satire on `Twitter` is increasing rapidly.

1.3 Our Contributions

Satire detection tasks are performed mostly on large documents (news articles) or at the sentence level; however, short textual data have not received much attention, particularly when using a deep learning-based approach. Moreover, linguistic context is employed when satire is created by the users [7]. Taking this into account, in this paper we present a deep learning model for binary classification, called BiLSTM with self-attention (`BiSAT`), for detecting satire in Twitter and news headline data. It extracts important contextual information based on satire from short textual data. `BiSAT` consists of an input, an embedding, a self-attention layer, and two bidirectional long short-term memory (BiLSTM) layers. It extracts important linguistic contextual information from the input text useful for detecting satire. It begins by converting the input textual data into a nominal vector. The embedding layer generates respective numeric vectors corresponding to the nominal vectors that are then presented to the first BiLSTM layer, which captures long-term temporal dependencies and linguistic contextual sequences from the input text in the opposite direction. The output of the first BiLSTM is fed to a self-attention layer, which gives the contextual information-based sequences of the first BiLSTM a different focus. Thereafter, the result of the self-attention layer is passed to the second BiLSTM, which generates an encoded vector by combining forward and backward contextual sequences.

The main intent behind the architecture of the proposed `BiSAT` model is to combine the strength of BiLSTM, which captures bidirectional contextual information, with the self-attention mechanism, which can capture relationships between tokens irrespective of their positions in the sequence. The second BiLSTM layer then delivers a more dense contextual representation of the linguistics tokens available in the input text after the self-attention stage and that leads to determining whether the input text is satirical or non-satirical. It helps in understanding satirical context and relationships among tokens and is particularly effective when dealing with tasks that require capturing long-range dependencies and understanding nuanced relationships within sequential data.

In addition, a 13-dimensional auxiliary feature vector consisting of shallow and deep linguistics features is also generated and concatenated with the encoded vector of the second BiLSTM to improve model performance. The learned contextual representation of the input text is forwarded to a dense layer, followed by a sigmoid function for satire detection.

In short, the key contributions of this paper can be summarized as follows:

- Exploring the problem of satire detection in short textual data and its role in spreading misinformation on the Web.
- Development of a BiLSTM with self-attention (`BiSAT`) model for satire detection in short textual data.
- Extraction of 13 auxiliary features which are employed in the `BiSAT` model to enhance its classification accuracy.
- Generation of a new dataset called Satire-280 which can be used as a benchmark for the satire detection problem.

The rest of the paper is organized as follows. Section 2 presents the existing studies for satire detection. Section 3 presents the functional details of the proposed satire detection approach. It also presents the auxiliary features applied in the proposed approach. Section 4 presents the datasets, different experimental settings, and empirical evaluation of the proposed approach, followed by its comparative analysis. Section 5 presents a critical discussion based on the varying parameters. Finally, Section 6 concludes the paper and highlights possible future research directions.

2 Related Work

This section presents the related studies for satire detection, which is considered a binary classification problem and is mostly based on the English language. The extracted features are based on the underlying concepts of linguistic clues, incongruity, offensive information, profane content, polarity reversal, and unexpectedness for satire detection. Burfoot and Baldwin [8] extracted lexical features, such as profanity, headlines, and slang. They considered a support vector machine (SVM) classifier to detect satire. Rubin et al. [9] applied SVM and used Canadian and US newspapers for satire and non-satire categories. Reganti et al. [10] considered several feature groups (lexical, literary device, sentiment amplifier, speech act, sensicon, and sentiment continuity disruption) for satire detection. They highlighted that the ensemble classifier performs better than SVM and random forest classifiers. Barbieri et al. [5] detected satirical news from tweets and extracted features, such as word-based, frequency, sentiments, etc.

Barbieri et al. [6] detected satirical news advertisements in Spanish tweets. They extracted features, such as slang, frequency, sentiments, characters, synonyms, parts-of-speech (POS), and ambiguity. They applied binary classification experiments for satire detection. Salas-Zárate et al. [11] collected datasets from Mexican and Spanish accounts from Twitter based on satirical and non-satirical news. They considered psycholinguistic features and considered seven groups of classifiers (Bayesian, functions, lazy, meta-classifiers, rules, miscellaneous, and trees) to perform the classification task. Thu and New [3] highlighted emotional features for satire detection from three online sources and applied SVM and bagging classifiers.

Stöckl [12] proposed satire detection from satirical websites and news articles from politics, business, technology, etc., using SVM and logistic regression classifiers. Ravi and Ravi [2] detected satire and irony and highlighted common characteristics in both FL categories. They mentioned that LIWC is a useful tool for the detection of these two FL categories. Dutta and Chakraborty [13] detected satire using both linguistics and machine learning tools and collected datasets, such as newswire documents, satire news articles, and NewYork Times articles. They extracted lexical (headlines, profanity, and slang) and semantic validity-based features. Sinha et al. [7] proposed considered images from a photo-sharing online platform `Flickr` and applied a dynamic autoencoder-based unsupervised clustering approach for satire detection.

Sarkar et al. [4] applied hierarchical deep neural networks using CNN, LSTM, and GRU for satire detection at both sentence- and document-level. They highlighted that the last sentence is important for satire detection. Yang et al. [14] proposed a hierarchical network and attention mechanism for satire detection. They also incorporated a few paragraph-level linguistic features along with the neural network and attention mechanism to detect satire. Sharma et al. [15] proposed Bangla satirical news detection using CNN. They detected satirical news using a hybrid approach, wherein the traditional term frequency-inverse document frequency and `Word2Vec` are combined to detect satirical information. Recently, Horvitz et al. [16] proposed a satirical news headlines generation task using a context-driven approach. They introduced a new approach for considering satirical news headlines in a real-world context along with an information retrieval pipeline.

The preceding discussion demonstrates that a variety of machine learning-based approaches and techniques are for automatic satire detection, particularly in news documents. In addition, few works address deep learning-based approaches for satire detection at the document-, paragraph-, and sentence-level and emphasize the significance of context in satirical data. Researchers have not paid much attention to detecting satire using deep learning-based approaches in short textual data, such as tweets and news headlines. Consequently, investigating satirical context for satire detection in short textual data is a challenging and worthwhile research problem.

3 Proposed Approach

This section describes the functional aspects of the proposed method, BiLSTM with self-attention (`BiSAT`), for satire detection. Figure 2 illustrates the workflow depicting the interactions of various functional components, such as the data crawling and data pre-processing modules, as well as the architecture of the `BiSAT` model.

3.1 Data Crawling and Pre-processing

In this study, we consider a total number of three datasets for the proposed approach, which includes two benchmarks and one newly created dataset. Out

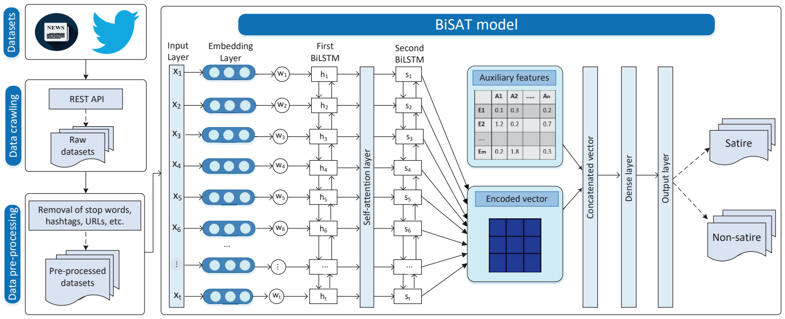


Figure 2 Workflow of the proposed satire detection approach.

of the two benchmark datasets that are taken from the existing literature, one dataset is based on news tweets. However, authors distributed only tweets ids due to the privacy policy of `Twitter`. Therefore, a crawler in `Python` 2.7 is implemented to crawl tweets through a popular `Python` library called `tweepy`.[3] It fetches tweets against its tweet ids by accessing the `Twitter` `REST` API. Further, these tweets are stored in a local repository. However, deleted and protected tweets are not crawled against their tweet ids. Moreover, a new `Twitter` dataset is also generated which is based on satirical and non-satirical hashtags. Further, Section 4.1 presents the details and statistics of all datasets.

Post crawling of all tweets/instances, several data cleaning steps are performed as a part of pre-processing to receive a refined dataset. It includes the elimination of `Twitter`-specific markers like mentions, retweets, digits, unusable white spaces, quotes, symbols, acronyms, ampersands, commas, stop words, non-ASCII characters, and emoticons. Finally, all duplicate tweets/instances are removed and converted into lower-case.

3.2 `BiSAT` Model

This section presents the layer-wise functional details of the proposed `BiSAT` model for the satire detection approach, which is motivated by the study presented in [18] in the following sub-sections.

3.2.1 Input layer

The input layer receives the pre-processed tweets/instances. Each input text carries t tokens and it is converted to an input vector. It is mapped with a

[3] https://www.tweepy.org/ (last accessed on 31 December 2023).

relative index value in the dictionary, i.e., $I \epsilon R^{1 \times t}$. Thereafter, a fixed-length padding is used for each input vector, i.e., $I \epsilon R^{1 \times p}$, where p is the maximum padding-length to eliminate the issue of different input lengths size.

3.2.2 Embedding layer

In neural network models, the embedding layer functions as a hidden layer. In this paper, pre-trained `Twitter`-specific *Global Vectors* (`GloVe`)[4] embedding is used. It is well-known pre-trained word embedding to obtain a vector representation of words. It is constructed from a large corpus and trained on global co-occurrence statistics using the word-pairs combination. This layer receives the input vector and converts each token into a distributional vector of size D-dimension. Hence, the input matrix is converted to $I \epsilon R^{p \times D}$.

3.2.3 First BiLSTM layer

In this study, we have used BiLSTM, which summarizes information, and captures long-term dependencies and contextual sequences in opposite directions of the input text. It moves from left to the right direction via a forward LSTM, captures the future information of the sequence, and generates a forward hidden state as $\overrightarrow{h_t}$. Also, it moves in the right to left direction, captures the historical latent sequential information via a backward LSTM, and generates a backward hidden state $\overleftarrow{h_t}$. Thereafter, a new hidden state is obtained by concatenation of these two hidden states, as given in Equation (1). Finally, the hidden state vector h_t encodes contextual knowledge related to satire from the input text.

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]. \tag{1}$$

3.2.4 Self-attention layer

Specific contextual words play a key role in the satirical text. Hence, extracting crucial information from such words is important to retrieve the overall satirical context from the input text. Using an attention mechanism [20], the proposed model learns to attend and target satirical words based on the outcome of the contextual sequences retrieved from the first BiLSTM layer from the input text. In this paper, we have used the self-attention layer (*aka* intra-attention) [21], a special kind of attention mechanism based on distinct positions of a single sequence to determine a representation of the same sequence. It learns the correlation between the tokens and their previous

[4] https://nlp.stanford.edu/projects/glove/ (last accessed on 31 December 2023).

sequence present in the input text. Positioned between the two BiLSTM layers, the self-attention mechanism allows the model to focus on important satirical token within the text sequence. It computes the attention weights that indicate the relevance/importance of different tokens to each other within the text sequence.

A sequence of input vector $X = [h_1, h_2, h_3, \ldots n]$ is generated where n is the sequence length and h_i refers to the ith vector. For each h_i, three vectors query (Q), key (K), and value (V) are transformed as $Q_i = X_i W_Q$, $K_i = X_i W_K$, and $V_i = X_i W_V$, where W_Q, W_K, and W_V are weight matrices. Thereafter, attention score, A_{ij}, as given in Equation (2), is calculated between x_i and x_j as dot products of their associated query and key vectors, where d_k is the dimensionality of the key vector.

$$A_{ij} = \frac{Q_i.K_j^T}{\sqrt{d_k}}. \tag{2}$$

The attention scores are normalized across the sequence length using the *Softmax* function to obtain attention weights, S_{ij}, as given in Equation (3).

$$s_{ij} = softmax(A_{ij}) = \frac{\exp(A_{ij})}{\sum_{J=1}^{n} \exp(A_{ij})}. \tag{3}$$

Equation (4) gives the weighted sum of the value vectors (V) via the obtained attention weights (s) generates the self-attention representation z_i for each vector h_i.

$$z_i = \sum_{j=1}^{n} s_{ij} h_i. \tag{4}$$

3.2.5 Second BiLSTM layer

The main purpose of the second BiLSTM is to deliver a more dense contextual representation of the linguistics tokens available in the input text, and that leads to determining whether the input text is satirical or non-satirical. The second BiLSTM receives the outcome of the self-attention layer as an input. It is propagated until the final hidden state (encoded vector) s_t is generated by combining both forward and backward directions of the second BiLSTM.

3.3 Auxiliary Features

This section gives the details of the hand-crafted 13 auxiliary features taken from the four feature groups (sentiment, punctuation, hyperbole, and affective). It generates a feature vector $f_v \, \epsilon \, R^d$ of d dimensions, where the value of d is 13. The main intent behind considering these rich set of hand-crafted linguistics features is to further enrich the satirical representation by fusing it into the proposed neural network-based models. Consequently, it helps in increasing the the overall classification performance.

3.3.1 Sentiment

Sentiment plays a crucial role in satirical texts while exposing particular customs and policies. To this end, the following sentiment-based features for satire detection are taken into consideration and it is extracted via a popular `Python` library called `TextBlob`.[5]

- Polarity score: This feature calculates the polarity score in a given input text.
- Subjectivity score: Subjectivity is a kind of sentiment that refers to a kind of personal opinion, emotion, or judgment about a topic or discussion. In this feature, we calculate the subjectivity score in a given input text.
- Count of positive words: This feature counts the total number of positive words in a given input text.
- Count of negative words: This feature counts the total number of negative words in a given input text.

3.3.2 Punctuation

Punctuation reflects the behavioral aspects of a satirical text. It is used mainly to show vocal tones and facial gestures, which are absent in textual data. To this end, the following punctuation-based features for satire detection are considered.

- Count of exclamation marks: This feature counts the total number of exclamation marks (!) in the input text. To this end, we search and count the occurrences of exclamation marks in a given input text.
- Count of dots: This feature counts the total number of dots (.) in the input text. To this end, we search and count the occurrences of dot in a given input text.
- Count of question marks: This feature counts the total number of question marks (?) in the input text. To this end, we search and count the occurrences of question marks in a given input text.

[5] https://textblob.readthedocs.io/en/dev/ (last accessed on 31 December 2023).

3.3.3 Hyperbole

Hyperbole reflects the exaggeration factor, which usually refers to the over-emphasis or extra-attention by satirical speakers. To this end, the following hyperbole-based features for satire detection are considered. The Natural Language Toolkit (*aka* NLTK),[6] a powerful `Python` package for natural language processing tasks, is used to tokenize a given input text and extract relative POS tags for these tokens, accordingly.

- Count of interjections: This feature counts the total number of interjections tag (i.e., UH) in a given input text.
- Count of adverbs: This feature counts the total number of adverbs tag (i.e., RB) in a given input text.
- Count of adjectives: This feature counts the total number of adjectives tag (i.e., JJ) in a given input text.

3.3.4 Affective

Affective contents are broadly used in satirical instances [11]. To this end, the following affective-based features for satire detection are considered. `ANEW` [22], a popular set of normative emotional rating resource is used to extract given below affective-based features.

- Valence score: This feature refers to the valence (pleasantness) of the emotions indicated by the word, such as going from sad to happy. The obtained valence score in a given input text is considered as an individual feature.
- Arousal score: This feature refers to the degree of arousal elicited by the word. The obtained arousal score in a given input text is considered as an individual feature.
- Dominance score: This feature refers to the dominance (power) of a word. The obtained score in a given input text is considered as an individual feature.

3.4 Concatenated Layers

The 13-dimensional feature vector, f_v, generated from the four groups of auxiliary features are combined with the generated encoded vector s_t of the second BiLSTM. Consequently, a concatenated vector, c_v, is generated, as given in Equation (5) which represents a dense and high-level contextual

[6] https://www.nltk.org/ (last accessed on 31 December 2023).

representation of the input text.

$$c_v = f_v \oplus s_t. \tag{5}$$

3.5 Dense and Output Layers

The fully connected dense layer receives the concatenated vector c_v. *Sigmoid*, a logistic regression function, is then used to classify the input text as either satire or non-satire.

4 Experimental Setup and Results

This section describes the datasets, experimental/parameter setttings, and the evaluation results of the proposed `BiSAT` model. It also provides a comparative analysis of `BiSAT` with a state-of-the-art and various baselines methods for satire detection.

4.1 Datasets

This section presents the description and statistics of three datasets. Tables 1 and 2 present the statistics of the crawled datasets and the final statistics of the datasets post pre-processing steps, respectively.

- Benchmark datasets: The first dataset is collected from Frain and Wubben [17], where satirical news and non-satirical news are considered from various online sources. Further, both satirical and non-satirical news contain *news headlines* and *news body*. We consider only *news headlines* for two reasons: (i) the proposed satire detection approach is based on short textual data, and (ii) *news headline* is an important indicator of satire [13]. The second dataset is collected from Thu and

Table 1 Statistics of the crawled datasets

Datasets ↓	#Satire	#Non-satire	Total
News headlines [17]	1704	1644	3348
News tweets [3]	9944	10056	20000
Satire-280	16374	25821	42195

Table 2 Final statistics of the datasets

Datasets ↓	#Satire	#Non-satire	Total
News headlines [17]	1656	1622	3278
News tweets [3]	9540	9989	19529
Satire-280	10596	22461	33057

New [3], where satirical and non-satirical tweets are collected from several satirical news and true news related accounts, respectively from Twitter.

- Satire-280 dataset: A new dataset called Satire-280 is created based on the 280 characters limit of a tweet. We collected hashtag-based annotated tweets during the period from October, 2018 to February, 2019. In the hashtag-based annotation technique, users self-label their posted tweets, wherein the hashtag works as a label for a particular tweet. In this paper, satirical tweets are crawled via #satire hashtag, and non-satirical tweets are crawled via #hate, #love, and #not hashtags.

4.2 Experimental Settings

In this paper, experimental tasks were implemented on a 2.00 GHz Intel processor machine along with 2 total cores, 85 flops, 4 threads, 4 GT/s bus speed, Windows 10 Pro (64-bit) operating system, DDR4-2133 memory types, and 8 GB RAM.Python 2.7 is used for data crawling and data preprocessing modules and Python 3.7 is used for implementation of the newly proposed BiSAT model. Keras,[7] a popular library in Python is used to execute the proposed BiSAT model.

4.3 Parameter Settings

Each dataset is divided into 20% for the training and 80% for the testing. Dropout of 0.4 is mainly taken to diminish over-fitting and improve generalization error. The first dropout is applied between the first BiLSTM and self-attention layer, whereas the second dropout is applied after the second BiLSTM layer. In the proposed model, *Adam* optimization algorithm, batch size of 64, binary cross-entropy loss function, and verbose of 2 are used. Early stopping, a kind of regularization is applied to avoid over-fitting. Glove 200 embedding dimensions and padding size value of 20 are used. A total of 300 neurons for each BiLSTMs (first BiLSTM and second BiLSTM) layer and 50 epochs are considered.

4.4 Evaluation Metrics

The proposed approach for satire detection is evaluated using four evaluation metrics which includes *Precision*, *Recall*, *F-score*, and *Accuracy* and defined

[7] https://keras.io/ (last accessed on 31 December 2023).

in Equations (6), (7), (8), and (9), respectively. These equations are defined via the four key terms (*true positive* (TP), *false positive* (FP), *true negative* (TN), and *false negative* (FN)). *TP* calculates the correctly identified satirical instances. *FP* calculates the wrongly identified satirical instances. *TN* calculates the correctly identified non-satirical instances. Finally, *FN* calculates the wrongly identified non-satirical instances.

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{6}$$

$$Recall\ (R) = \frac{TP}{TP + FN} \tag{7}$$

$$F\text{-}score\ (F) = \frac{2 \times P \times R}{P + R} \tag{8}$$

$$Accuracy\ (A) = \frac{TP + TN}{\#\,instances}. \tag{9}$$

4.5 Evaluation Results and Comparative Analysis

This section presents the performance evaluation results of the proposed `BiSAT` model in terms of *precision*, *recall*, and *f-score* on news headlines [17], news tweets [3], and Satire-280 datasets for training and testing data, as given in Tables 3 and 4, respectively. It also includes the comparative result analysis of the proposed `BiSAT` model with a relevant work and baseline methods in both Tables 3 and 4. It shows that the proposed model receives better results across all datasets on both training and testing data. News tweets [3] dataset receives significantly better results across all datasets.

We infer some interesting observation from these results presented in both tables that the proposed `BiSAT` model receives better results on both 140 and 280 character limits of tweet-length. Moreover, the proposed model receives

Table 3 Performance evaluation results on training data

Datasets →	News headlines [17]			News tweets [3]			Satire-280		
Methods ↓	P	R	F	P	R	F	P	R	F
Proposed `BiSAT` model	**0.75**	**0.77**	**0.76**	**0.85**	**0.80**	**0.83**	**0.73**	**0.71**	**0.72**
Yang et al. [14]	0.68	0.65	0.66	0.68	0.63	0.65	0.60	0.61	0.60
CNN	0.59	0.55	0.57	0.66	0.61	0.63	0.60	0.54	0.57
GRU	0.56	0.51	0.53	0.70	0.72	0.71	0.63	0.51	0.56
LSTM	0.61	0.62	0.62	0.77	0.75	0.76	0.65	0.63	0.64
BiGRU	0.58	0.60	0.59	0.71	0.75	0.73	0.62	0.60	0.61
DNN	0.54	0.50	0.52	0.54	0.50	0.52	0.47	0.42	0.44

Table 4 Performance evaluation results on testing data

Datasets →	News headlines [17]			News tweets [3]			Satire-280		
Methods ↓	P	R	F	P	R	F P	R	F	
Proposed BiSAT model	**0.79**	**0.83**	**0.80**	**0.86**	**0.90**	**0.88**	**0.75**	**0.68**	**0.71**
Yang et al. [14]	0.56	0.70	0.62	0.75	0.70	0.72	0.52	0.51	0.51
CNN	0.58	0.50	0.54	0.65	0.76	0.70	0.61	0.40	0.48
GRU	0.53	0.59	0.55	0.70	0.73	0.72	0.64	0.58	0.61
LSTM	0.69	0.61	0.65	0.78	0.77	0.77	0.71	0.60	0.66
BiGRU	0.58	0.52	0.55	0.77	0.75	0.76	0.60	0.55	0.57
DNN	0.57	0.59	0.58	0.58	0.51	0.54	0.65	0.60	0.63

Table 5 Training accuracy versus testing accuracy values over (a) news headlines [17], (b) news tweets [3], and (c) Satire-280 datasets

Datasets →	News headlines [17]		News tweets [3]		Satire-280	
Methods ↓	Training accuracy	Testing accuracy	Training accuracy	Testing accuracy	Training accuracy	Testing accuracy
Proposed BiSAT model	**0.75**	**0.71**	**0.83**	**0.80**	**0.78**	**0.75**
Yang et al. [14]	0.58	0.54	0.68	0.66	0.61	0.60
CNN	0.60	0.58	0.75	0.74	0.73	0.72
GRU	0.61	0.57	0.77	0.76	0.73	0.73
LSTM	0.65	0.62	0.78	0.77	0.75	0.74
BiGRU	0.59	0.59	0.73	0.70	0.71	0.69
DNN	0.62	0.60	0.60	0.63	0.63	0.69

better results on other short textual dataset (i.e., news headlines [17]). The proposed BiSAT model shows the highest *f-score* of 0.83 and 0.88 on training and testing data, respectively.

Moreover, Table 5 shows that the proposed BiSAT model is best fit in terms of training and testing accuracy values over all datasets on both training and testing data. The news tweets [3] dataset performs significantly better across all datasets. It receives training accuracy and testing accuracy of 0.83 and 0.80, respectively. Interestingly, the Satire-280 dataset also shows good results.

4.5.1 Comparison with a state-of-the-art method

This section compares the proposed BiSAT model with Yang et al. [14] study. As stated earlier in Section 2, they detected satire on both paragraph and document level. They considered a news corpus and applied CNN, BiGRU, and attention layers. In addition, they used four sets of linguistics features, such as psycholinguistic, writing stylistic, readability, and structural. They concatenated the generated feature vector from these feature sets with the outcome of the attention layer.

Tables 3 and 4 show that the BiSAT model outperforms Yang et al. [14] method over all datasets in terms of *precision*, *recall*, and *f-score* on both training and testing data. Table 5 shows that the proposed BiSAT model outperforms the Yang et al. [14] method across all datasets for both training and testing accuracy values. Also, it can be seen that the proposed BiSAT model outperforms the Yang et al. [14] method for the Satire-280 dataset.

4.5.2 Comparison with neural network-based baselines

The proposed BiSAT model for satire detection task is compared with the following neural network-based baseline methods.

- CNN: It performs continuous learning and updating mechanism. It includes convolution filters that retrieve hidden features. It extracts semantically rich features by the parameterized sliding window for text classification tasks.
- GRU: It belongs to the RNN family. It includes (*update gate* and *reset gate*.
- LSTM: As stated earlier, and unlike GRU, it consists of three digital gates.
- BiGRU: It is a special kind of GRU that moves in opposite directions.
- DNN: It consists of hidden nodes that takes input from data and a set of weights. The product of both input and weights are added and forwarded using an activation function.

Tables 3 and 4 show that the proposed BiSAT model outperforms baseline methods across all datasets on both training and testing data. Overall, LSTM received significantly better results across baseline methods. One of the main key reasons behind this is LSTM has a feedback loop, which helps in generating useful information and also captures dependencies between words. Similarly, Table 5 shows that the proposed BiSAT model outperforms baseline methods across all datasets. Overall, RNN models like LSTM, GRU, and BiGRU perform better across baseline methods for both training accuracy and testing accuracy, respectively. In addition, the proposed model shows better training accuracy and testing accuracy on the Satire-280 dataset.

5 Discussion

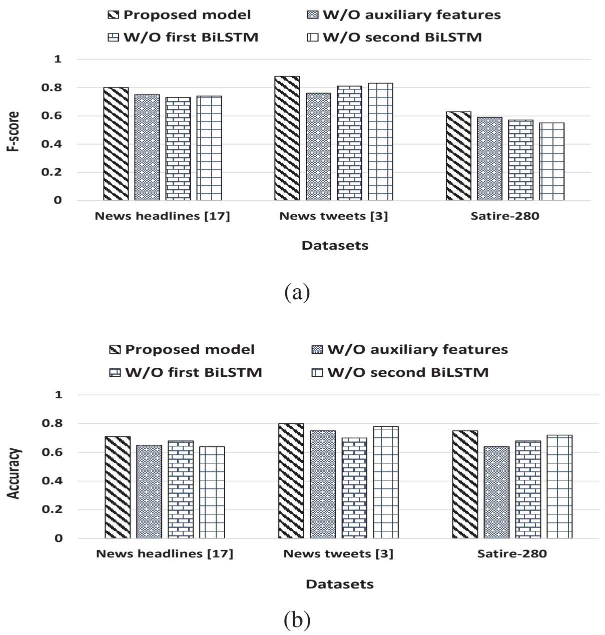
This section presents the ablation study of the proposed BiSAT model. It also includes the effects of embedding dimensions and parameters tuning in terms of *optimization algorithms* and *activation functions* on the proposed BiSAT model.

5.1 Ablation Study

We carried out an ablation study of the proposed BiSAT model to examine its layer-wise effect on *f-score* and accuracy values. To this end, the following ablation study on the proposed BiSAT model is performed: (1) without (W/O) auxiliary features, (2) W/O first BiLSTM, and (3) W/O second BiLSTM. Figure 3 presents the ablation study with the layer-wise effect of the proposed model over news headlines [17], news tweets [3], and Satire-280 datasets. It shows that the proposed BiSAT model with two BiLSTMs, self-attention, and hand-crafted auxiliary features performs significantly better as compared with its other combinations.

5.2 Effect of Embedding Dimensions

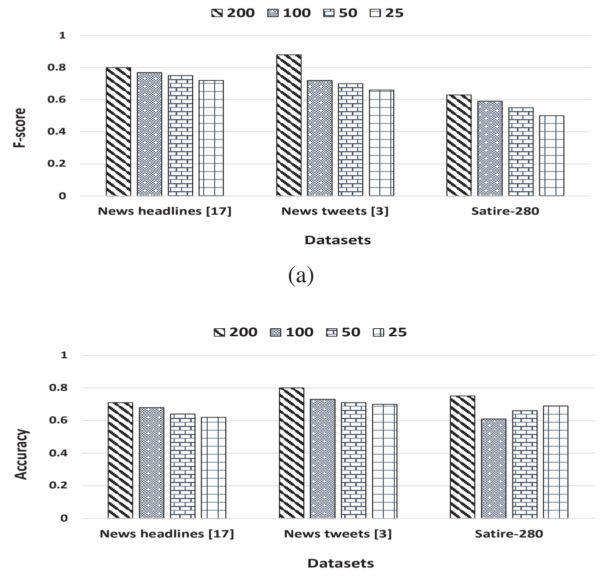
As stated earlier, we have used GloVe pre-trained embedding dimension of the size of 200. This section analyzes the effects of GloVe embedding on 100, 50, and 25 dimensions. Figure 4 presents the effect of GloVe embedding dimensions on the proposed model across all datasets for (a) *f-score* and (b)



(a)

(b)

Figure 3 Ablation study with layer-wise effect of the proposed BiSAT model. (a) *f-score* across all datasets, (b) *accuracy* across all datasets.



(a)

(b)

Figure 4 Effect of `GloVe` embedding dimensions on the proposed `BiSAT` model. (a) *f-score* across all datasets, (b) *accuracy* across all datasets.

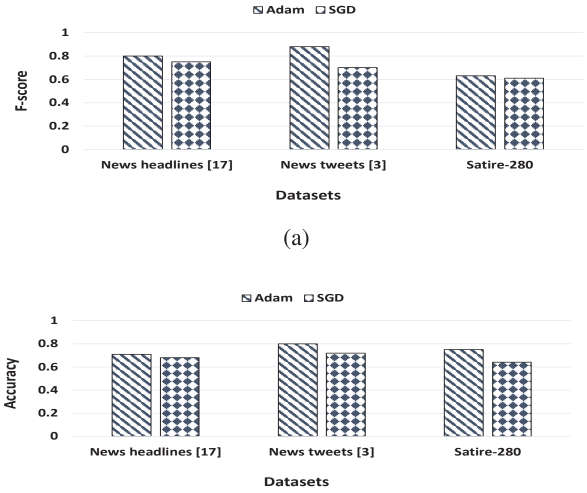
accuracy values. It can also be noticed from Figure 4 that news tweets [3] performs better for all `GloVe` dimensions across all datasets for both *f-score* and *accuracy* values. This infers that the higher dimension performs remarkably better in extracting features from the input text, and that also seems advantageous for the proposed model.

5.3 Effects of Parameters

In deep learning-based models, parameter tuning is an important factor. This section presents the effect on the classification results of the proposed `BiSAT` model for satire detection approach by analyzing two parameters (*activation functions* and *optimization algorithms*) across all datasets.

5.3.1 Optimization algorithms

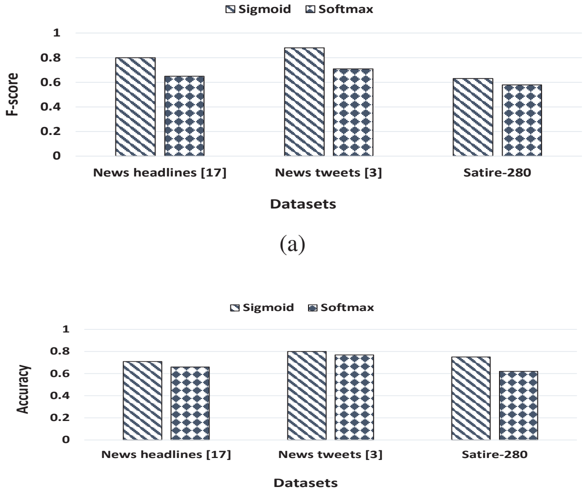
Selecting optimization algorithms is crucial for the classification model. In this section, we analyze the performance of the proposed `BiSAT` model on two different optimization algorithms (*Adam* and *SGD*) over all datasets for *f-score* and *accuracy* values. Figure 5 presents the classification effect results



(a)

(b)

Figure 5 Effect of two optimizers on the proposed `BiSAT` model. (a) *f-score* across all datasets, (b) *accuracy* across all datasets.



(a)

(b)

Figure 6 Effect of two activation functions on the proposed `BiSAT` model. (a) *f-score* across all datasets, (b) *accuracy* across all datasets.

of the proposed model on two optimization algorithms (*Adam* and *SGD*) over news headlines [17], news tweets [3], and Satire-280 datasets for (a) *f-score* and (b) *accuracy* values. It shows that the performance on *Adam* is better in comparison to *SGD* over all datasets.

5.3.2 Activation functions

Similar to selecting an optimization algorithm, choosing an activation function is an important factor for any classification model. Figure 6 presents the classification effect results of the proposed `BiSAT` model on two activation functions (*sigmoid* and *softmax*) over all datasets for (a) *f-score* and (b) *accuracy* values. It shows that the proposed `BiSAT` model performs remarkably better on *sigmoid* as compared to *softmax* over all datasets as it is suitable for any binary classification problem.

6 Conclusion

Satire is an important category of figurative language, and it is widely present in user-generated content in numerous forms, such as satirical news, consumer reviews, and tweets. In this paper, we have introduced a new `BiSAT` model for detecting satire in short textual data. The `BiSAT` model employs BiLSTM, which summarizes information and captures long-term dependencies and contextual sequences in opposite directions of the input text. We have also presented 13 auxiliary features that are integrated with `BiSAT` to boost its satire detection accuracy. The effectiveness of `BiSAT` is demonstrated on three datasets, including two benchmark datasets and one created specifically for this work for satire detection which is accessible to download from GitHub repository along with code: https://github.com/Ashraf-Kamal/Satire-Detection. The evaluation results are promising, outperforming several neural network-based baselines and a state-of-the-art method. Extending the proposed model to short multilingual data and taking into account multimodal data (e.g., text+visual, text+audio, and so on) seems significant future directions of research.

Acknowledgements

Supported by Visvesvaraya PhD Scheme, MeitY, Govt. of India, ‘MEITY-PHD-555’.

References

[1] Abulaish, M., Kamal, A., Zaki, M.J.: A survey of figurative language and its computational detection in online social networks. ACM Transactions on the Web 14(1): 1–52 (2020).

[2] Ravi, K., Ravi, V.: A novel automatic satire and irony detection using ensembled feature selection and data mining. Knowledge-Based Systems 120, 15–33 (2017).

[3] Thu, P.P., New, N.: Implementation of emotional features on satire detection. In: Proceedings of the 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Kanazawa, Japan, pp. 149–154, IEEE (2017).

[4] De Sarkar, S., Yang, F., Mukherjee, A.: Attending sentences to detect satirical fake news. In: Proceedings of the 27th International Conference on Computational Linguistics (COLING), Santa Fe, New Mexico, USA, pp. 3371–3380 (2018).

[5] Barbieri, F., Ronzano, F., Saggion, H.: Do we criticise (and laugh) in the same way? automatic detection of multi-lingual satirical news in Twitter. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI), Buenos Aires, Argentina, pp. 1215–1221 (2015).

[6] Barbieri, F., Ronzano, F., Saggion, H.: Is this tweet satirical? a computational approach for satire detection in spanish. Procesamiento del Lenguaje Natural (55): 135–142 (2015).

[7] Sinha, A., Patekar, P., Mamidi, R.: Unsupervised approach for monitoring satire on social media. In: Proceedings of the 11th Forum for Information Retrieval Evaluation (FIRE), Kolkata, India, pp. 36–41, ACM (2019).

[8] Burfoot, C., Baldwin, T.: Automatic satire detection: are you having a laugh?. In: Proceedings of the AAssociation for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP), Suntec, Singapore, pp. 161–164, ACL and AFNLP (2009).

[9] Rubin, V. L., Conroy, N., Chen, Y., Cornwell, S.: Fake news or truth? using satirical cues to detect potentially misleading news. In: Proceedings of the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), San Diego, California, pp. 7–17, ACL (2016).

[10] Reganti, A. N., Maheshwari, T., Kumar, U., Das, A., Bajpai, R.: Modeling satire in English text for automatic detection. In: Proceedings of the Workshop on Sentiment Elicitation from Natural Text for Information Retrieval and Extraction (SENTIRE, ICDMW), Barcelona, Spain, pp. 970–977, IEEE (2016).

[11] del Pilar Salas-Zárate, M., Paredes-Valverde, M. A., Rodriguez-García, M. Á., Valencia-García, R., Alor-Hernández, G.: Automatic detection of satire in Twitter: a psycholinguistic-based approach. Knowledge-Based Systems 128: 20–33 (2017).

[12] Stöckl, A. Detecting Satire in the News with Machine Learning. https://doi.org/10.13140/RG.2.2.17157.40164, pp. 1–5 (2018).

[13] Dutta, S., Chakraborty, A.: A deep learning-inspired method for social media satire detection. In: Wang J, Reddy GRM, Prasad VK, Reddy VS (eds). Soft Computing and Signal Processing, Springer, pp. 243–251 (2019).

[14] Yang, F., Mukherjee, A., Dragut, E.: Satirical news detection and analysis using attention mechanism and linguistic features. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Copenhagen, Denmark, pp. 1979–1989, ACL (2017).

[15] Sharma, A. S., Mridul, M. A., Islam, M. S.: Automatic detection of satire in bangla documents: a cnn approach based on hybrid feature extraction model. In: Proceedings of the 2nd International Conference on Bangla Speech and Language Processing (ICBSLP), Sylhet, Bangladesh, pp. 1–5, IEEE (2019).

[16] H, Zachary., Do, Nam., Littman, M. L.: Context-driven satirical headline generation. In: Proceedings of the 2nd Workshop on Figurative Language Processing (FLP), pp. 40–50, ACL (2020).

[17] Frain, A., Wubben, S.: SatiricLR: a language resource of satirical news articles. In: Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC), Portorož, (Slovenia), pp. 4137–4140 (2016).

[18] Ortega-Bueno, R., Rosso, P., Pagola, J. E.: UO UPV2 at HAHA 2019: bigru neural network informed with linguistic features for humor recognition. In: Proceedings of the Iberian Languages Evaluation Forum, co-located with 35th Conference of the Spanish Society for Natural Language Processing, CEUR Workshop, Bilbao, Spain, pp. 212–221 (2019).

[19] Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation 9(8): 1735–1780 (1997).

[20] Luong, M. T., Pham, H., Manning, C. D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Lisbon, Portugal, pp. 1412–1421, ACL (2015).

[21] Cheng, J., Dong, L., Lapata, M.: Long short-term memory-networks for machine reading. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Austin, Texas, USA, pp. 551–561, ACL (2016).

[22] Bradley, M. M., Lang, P. J.: Affective norms for English words (ANEW): instruction manual and affective ratings. Technical report C-1, the center for research in psychophysiology, University of Florida (1999).

[23] Kim, Y. Convolutional neural networks for sentence classification. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, pp. 1746–1751, ACL (2014).

[24] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, pp. 1724–1734, ACL (2014).

[25] Schmidhuber J. Deep learning in neural networks: An overview. Neural Networks 61, 85–117 (2015).

[26] Kamal, A., Abulaish, M.: Self-deprecating humor detection: a machine learning approach. In: Proceedings of the 16th International Conference of the Pacific Association for Computational Linguistics (PACLING), Hanoi, Vietnam, October; pp. 483–494, Springer (2019).

[27] Kamal, A., Abulaish, M.: An LSTM-based deep learning approach for detecting self-deprecating sarcasm in textual data. In: Proceedings of the 16th International Conference on Natural Language Processing (ICON), Hyderabad, India, December 18–21, 2019; pp. 201–210, ACL (2019).

[28] Abulaish, M., Kamal, A.: Self-deprecating sarcasm detection: an amalgamation of rule-based and machine learning approach. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, Chile; pp. 574–579, IEEE (2018).

[29] Kamal, A., Abulaish, M.: CAT-BiGRU: Convolution and Attention with Bi-Directional Gated Recurrent Unit for Self-Deprecating Sarcasm Detection. Cognitive Computation: 1–19 (2021).

Biographies

Ashraf Kamal received his Ph.D. degree in Computer Science from Jamia Millia Islamia (A Central University), New Delhi, India in 2021. Currently, he is a Machine Learning Engineer at PayPal, Chennai, India. He qualified UGC-NET in 2014 and his research interests include text mining, machine learning, and information retrieval. He was a recipient of the Visvesvaraya Ph.D. Fellowship from the Ministry of Electronics and Information Technology, Government of India to pursue his Ph.D. work. He has published over 10 research papers in reputed journals and conference proceedings, including two in IEEE/ACM Transactions.

Muhammad Abulaish (Senior Member, IEEE) received his Ph.D. degree in Computer Science from Indian Institute of Technology (IIT) Delhi in 2007. He is a Full Professor in the Department of Computer Science, South Asian University, New Delhi, India. His research interests include data analytics and mining, social computing, machine learning, and data-driven cyber Security. He has published over 139 research articles in international journals, books, and conference proceedings, including seven in IEEE/ACM Transactions.

He is an Associate Editor for the Social Network Analysis and Mining journal. He served as a Senior Program Committee member for CIKM'22. As a member of the Program Committee, he frequently serves prestigious international conferences such as SDM, CIKM, IJCAI-ECAI, PAKDD, Web Intelligence, and BIOKDD. He has also served as Publicity Co-chair for WI'19 and WI'20, as well as Workshop Co-chair for ASONAM'20. He is also a member of the editorial board and a reviewer for numerous reputable journals. He holds senior memberships with IEEE, ACM, and CSI. In addition, he is a lifetime member of ISTE, IETE, and ISCA.

Jahiruddin received his Ph.D. degree in Computer Science from Jamia Millia Islamia (A Central University), New Delhi, India in 2012. He is currently a Full Professor at the Department of Computer Science, Jamia Millia Islamia. His research interests include text mining, computational biology, and social network analysis. He has published over 20 research papers in various reputed journals and conference proceedings.