BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection

Article *in* Journal of King Saud University - Computer and Information Sciences · May 2022

DOI: 10.1016/j.jksuci.2022.05.006

7 authors, including:

Shakir Khan
Imam Mohammad ibn Saud Islamic University
139 PUBLICATIONS **3,564** CITATIONS
SEE PROFILE

Mohd Fazil
Imam Mohammad ibn Saud Islamic University
25 PUBLICATIONS **699** CITATIONS
SEE PROFILE

Vineet K Sejwal
Jamia Millia Islamia
11 PUBLICATIONS **279** CITATIONS
SEE PROFILE

Reemiah Muneer Alotaibi
Imam Mohammad ibn Saud Islamic University
16 PUBLICATIONS **671** CITATIONS
SEE PROFILE

Journal of King Saud University – Computer and Information Sciences xxx (xxxx) xxx

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection

Shakir Khan [a,*,1], Mohd Fazil [b,2], Vineet Kumar Sejwal [c,3], Mohammed Ali Alshara [a,4], Reemiah Muneer Alotaibi [a,5], Ashraf Kamal [d,6], Abdul Rauf Baig [a,7]

[a] College of Computer and Information Sciences in Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia
[b] Center for Transformative Learning, University of Limerick, Ireland
[c] Directorate of Education, New Delhi, India
[d] ACL Digital, Bengaluru, India

ARTICLE INFO

Article history:
Received 30 January 2022
Revised 18 April 2022
Accepted 10 May 2022
Available online xxxx

Keywords:
Hate speech detection
Deep learning
Social network security
Twitter data analysis
BiCHAT

ABSTRACT

Online social networks(OSNs) face the challenging problem of hate speech, which should be moderated for the growth of OSNs. The machine learning approaches dominate the existing set of approaches for hate speech detection. In this study, we introduce **BiCHAT**: a novel **Bi**LSTM with deep **C**NN and **H**ierarchical **AT**tention-based deep learning model for tweet representation learning toward hate speech detection. The proposed model takes the tweets as input and passes through a BERT layer followed by an attention-aware deep convolutional layer. The convolutional encoded representation further passes through an attention-aware Bidirectional LSTM network. Finally, the model labels the tweet as hateful or normal through a softmax layer. The proposed model is trained and evaluated over the three benchmark datasets extracted from Twitter and outperforms the state-of-the-art (SOTA) (Khan et al., 2022; Roy et al., 2020; Ding et al., 2019) and baseline methods with an improvement of 8%, 7% and 8% in terms of precision, recall, and f-score, respectively. BiCHAT also demonstrates good performance considering training and validation accuracy with an improvement of 5% and 9%, respectively. We also examined the impact of different constituting neural network components on the model. On analysis, we observed that the exclusion of the deep convolutional layer has the highest impact on the performance of the proposed model. We also investigated the efficacy of different embedding techniques, activation function, batch size, and optimization algorithms on the performance of the BiCHAT model.

* Corresponding author.
E-mail addresses: Sgkhan@imamu.edu.sa (S. Khan), mamalsharaa@imamu.edu.sa (M.A. Alshara), RMALotaibi@imamu.edu.sa (R.M. Alotaibi), abbaig@imamu.edu.sa (A.R. Baig).

[1] Currently Shakir Khan is an Associate Professor at College of Computer and Information Sciences in Imam Mohammad Ibn Saud Islamic University, Riyadh (Saudi Arabia).
[2] Currently Mohd Fazil is working as a Postdoctoral Researcher at the Center for Transformative Learning, University of Limerick, Ireland.
[3] Currently Vineet Kumar Sejwal is working as Post Graduate Teacher at Government of Delhi, NCT, New Delhi, India.
[4] Currently Mohammed Ali Alshara is working as an Assistant Professor and Vice Dean of quality at College of Computer and Information Sciences in Imam Mohammad Ibn Saud Islamic University, Riyadh (Saudi Arabia).
[5] Currently Reemiah Muneer Alotaibi is working as an Assistant Professor at College of Computer and Information Sciences in Imam Mohammad Ibn Saud Islamic University, Riyadh (Saudi Arabia).
[6] Currently Ashraf Kamal is working as a Data Scientist at ACL Digital, Bengaluru, India.
[7] Currently Abdulrauf Baig is working as a Professor at College of Computer and Information Sciences in Imam Mohammad Ibn Saud Islamic University, Riyadh (Saudi Arabia).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2022.05.006
1319-1578/

Please cite this article as: S. Khan, M. Fazil, Vineet Kumar Sejwal et al., BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection, Journal of King Saud University – Computer and Information Sciences, https://doi.org/10.1016/j.jksuci.2022.05.006

1. Introduction

Twitter, Facebook, Instagram, and other OSNs are modern communication media. These OSNs are part of users' routine life who spend considerable time on these platforms to discuss the current trends and debate different topics. A significant number of the world population (more than half) uses one or the other OSN. The activities of large OSN users generate a massive amount of data, which is analyzed to uncover hidden knowledge and gain insight into users' behavior. In the existing literature, researchers have investigated and modeled the user-generated content to solve the problem of sentiment analysis (Jain et al., 2021), event summarization, sarcasm detection (Abulaish et al., 2018; Kamal and Abulaish, 2019), keyword extraction (Abulaish et al., 2020; Abulaish et al., 2022), and so on. The user discourse on OSNs is not limited to personal news but to sharing opinions about national and international politics, current affairs, and financial updates. The Web has hundreds of OSN, but few are significantly popular among the users and have a large user base such as Facebook, Twitter, Instagram, WhatsApp, and Reddit. These OSNs have hundreds of millions of users, and most internet users use them.

The large user base, easy-to-use functionality, and anonymity have attracted anti-social elements and adversaries to perform illicit activities such as fake profile creation, trolling, abusing, and rumor propagation. Meantime, researchers are tackling these OSN-related security challenges by presenting novel methods towards socialbot detection (Fazil and Abulaish, 2018; Fazil et al., 2021; Davis et al., 2016), rumor prediction (Abulaish et al., 2019), clickbait detection (Potthast et al., 2016), stock market prediction (Akhtar et al., 2022). Among OSN-related security challenges, the problem of offensive language is significantly prevalent and threatening. Based on the target group and intent, an offensive language can be hate speech, cyberbullying, adult content, trolling, abuse, racism, or profanity. Hate speech is one of the threatening types of offensive language where the targeted group/individual is intimidated with an intent of harm, violence, or social chaos (Husain and Uzuner, 2021). There is a subtle difference between various types of offensive languages.

The existing literature has no consensus on a universally accepted definition of hate speech. Some define it as violence promoting content, and some as aggressive content. Hate speech on OSNs is an aggressive post targeted at specific groups based on certain identifying characteristics such as religion, sexual orientation, gender, and ethnicity. The existing studies show that the hate speech problem is in the upward direction, particularly with the rise of right-wing extremism.[8] The incidence of online hate speech and derogatory content rises sharply during the pandemic. Sometimes, hate speech on social media causes riots and social unrest in real life.[9] In the words of Davidson et al. (2017), hate speech expresses "hatred towards a targeted group or intended to be derogatory, to humiliate, or to insult the members of the group". In terms of Twitter, a tweet is hateful that "*promotes violence against or directly attacks or threatens other people based on race, ethnicity, nationality, sexual orientation, gender, religious affiliation, age, disability, or serious disease.*" Chinese people are facing aggressive hatemongering in online media spaces since the COVID-19 outbreak from Wuhan. Meanwhile, governments and OSN service providers are formulating policies to moderate this menace, but no usable and efficient solution exists.

[8] https://www.justice.gov/hatecrimes/hate-crime-statistics.

[9] https://www.dw.com/en/capitol-hill-riots-prompt-germany-to-revisit-online-hate-speech-law/a-56171516.

1.1. Our contributions

The existing literature has several deep learning models for hate speech detection. In this direction, researchers have presented deep learning models using different neural network components like LSTM, CNN (Vigna et al., 2017; Badjatiya et al., 2017; Park and Fung, 2017; Zhang et al., 2018). The existing deep models generally use one neural network component in addition to some user-defined features as extra features. Different deep learning components are effective over types of datasets. For example, recurrent neural networks are helpful for the sequential dataset, whereas CNN is useful for datasets organized in a grid-like format. In the same direction of research, Roy et al. (2020) introduced a deep learning framework using a deep CNN for hate speech classification and evaluated it over a Twitter dataset with an accuracy of 92%. The existing literature understudies the integration of BiLSTM and CNN with the attention mechanism along with contextual embedding for hate speech detection. To this end, this study introduces a deep neural network model, BiCHAT, a BERT employing deep CNN, BiLSTM, and hierarchical attention mechanism for hate speech detection. The proposed model first applies a stack of CNN layers to extract more complex spatial and position-invariant features (Roy et al., 2020). The deep CNN layer identifies both inter-tweet and intra-tweet dependencies.

We use the state-of-the-art transfer-based language model, BERT, to extract the contextual word representation. Further, we integrate the BiLSTM with deep CNN to incorporate the long-range dependencies among the semantically similar words. The BiCHAT also applies high-level attention over the encoded representation from deep CNN and a low-level attention mechanism on BiLSTM output. We use the two levels of attention mechanism to assign variable weights to features from both deep CNN and BiLSTM. The attention coefficient represents the discriminating power of encoded features. The high- and low-level attentions used in BiCHAT resemble a hierarchical structure, therefore, named hierarchical attention.

Thus, the proposed model integrates the strength of contextual word representation, deep CNN, BiLSTM, and hierarchical attention. The main contributions of this study are as follows:

- Introduce a novel deep model, BiCHAT, integrating the strength of contextual word representation, deep CNN, BiLSTM, and hierarchical attention to learn effective content representation for efficient detection of hateful content.
- Perform a detailed comparative performance evaluation of BiCHAT over three Twitter datasets against several SOTA and baselines methods to establish its efficacy.
- Perform the ablation analysis to analyze the impact of different constituting neural network components used in the proposed BiCHAT model.
- Perform the impact analysis of various neural network parameters on BiCHAT model.

The flow of the remaining paper is as follows. It starts with a review of existing literature on hate speech detection. Further, Section 3 introduces the proposed model with a detailed description of each of its components. Section 4 discusses the experimental settings and the underlying performance evaluation results of proposed model. This section also performs a comparative evaluation of BiCHAT with three SOTA and five baseline models. This section also examines the impact of various neural network components. Section 5 investigates the efficacy of neural network hyperparameters on the proposed BiCHAT model. Finally, Section 6 concludes the paper and presents future directions of research.

2. Related works

The existing literature has various approaches to study the different aspects of hate speech on OSNs. The hate speech detection approaches are based: (i) statistical analysis of textual content, (ii) pattern mining, and (iii) machine learning. Further, machine learning-based methods can be classified into two categories: (i) feature engineering-based classical machine learning methods and (ii) current deep learning-based methods.

In classical feature engineering-based approaches, the researchers first devise the features based on textual content, user profile, template-based patterns, and user-tweet networks. Further, they train and evaluate the machine learning classification models such as naïve Bayes, decision tree, random forest, XGBoost. Author in Warner and Hirschberg (2012) devised template, word-gram, and part of speech-based features to train the SVM^{light} classification model employing a linear kernel function. The authors evaluated the approach over two datasets and found that bigram and trigram patterns downgrade the model performance. Kwok and Wang (2013) also encoded the text using unigram-based features. The author further trained a naïve Bayes classification model to classify the racist content. They also used bigram, trigram, and sentiment-related features to train the machine learning model and reported an improved classification performance. Burnap and Williams (2015) used various n-gram (1–5) features and trained three machine learning models to classify hate speech from Twitter and concluded that voted ensemble classifier shows the best performance of 0.89, 0.69, and 0.77 considering accuracy, recall, and f-score, respectively. Waseem and Hovy (2016) annotated a collection of 16k tweets to construct a benchmark dataset of hate and genuine tweets and published it. The authors trained a logistic regression classifier using $1-4$-gram features. They experimented by adding gender and location features and reported the best results. In Davidson et al. (2017), authors discussed the subtle differences among different categories of offensive languages. They further devised unigram, bigram, POStag-based n-gram, sentiment score, and other linguistic features and evaluated the proposed approach using a logistic regression-based classification model for hate speech detection. In another multi-class classification approach, authors in Malmasi and Zampieri (2017) trained support vector machine employing character and word n-gram-based features and classified the hate, offensive, and neutral text. The authors concluded that the character 4-gram-based model performs best.

The classical machine learning models are not robust as adversaries circumvent the manually designed features by manipulating the behavior as per the devised features. Recently, deep learning techniques have replaced the classical machine learning models in various applications (Haq et al., 2021; Qaisar et al., 2021). Researchers proposed different deep learning architectures for automated hate speech detection (Djuric et al., 2015; Vigna et al., 2017; Badjatiya et al., 2017; Park and Fung, 2017; Zhang et al., 2018). Djuric et al. (2015) used the paragraph2vec (Le and Mikolov, 2014) language model for comment representation learning and further classified the hate comment using the logistic regression. Badjatiya et al. (2017) represented content using different word embedding techniques – GloVe, word2vec, and fastText and modeled the hate speech detection using CNN, LSTM, and DNN. Vigna et al. (2017) crawled Italian Facebook data and extracted morpho-syntactical, sentiment polarity, and word embedding-based features and trained over SVM and LSTM classification models to classify the hate speech. They also introduced a taxonomy of different hate categories considering the subject of the hate. In Park and Fung (2017), authors presented a convolutional neural network-based deep learning model and used logistic regression to classify the abusive content. The authors concluded that the convolutional network with logistic regression outperforms the SOTA and baseline models. In Zhang et al. (2018), Zhang et al. classified the hate speech using a CNN and GRU-based deep neural network model with the best performance of 0.94 considering an f-score. Roy et al. (2020) proposed a deep CNN-based framework and reported the best performance accuracy of 92% over a Twitter dataset. Recently, researchers also studied the hate content problem in code-mixed languages like Hinglish. Kamble and Joshi (2018) evaluated the efficacy of basic deep learning architecture-based models (CNN, LSTM, and BiLSTM) to code-mixed hate content classification employing domain-specific word embedding. On investigation, the authors concluded that the domain-specific embeddings are efficacious and outperform the pre-trained embeddings. The existing literature lacks benchmark datasets in regional languages, consequently hampering the hate speech detection research in low-resource languages. Researchers have made moderate progress in modeling low-resource language, but it is still in the infancy stage. In Pamungkas et al. (2021), Pamungkas et al. introduced a zero-shot learning-based model to detect hate in cross-lingual content. Researchers are also exploring the hate speech problem from different aspects of hate speech, including the analysis of social groups at the receiving end of hate (Mossie and Wang, 2020), analysis of the generalization of hate speech detection models over different datasets and categories of hate (Fortuna et al., 2021).

3. Proposed deep learning model

In this section, we describe each layer of the proposed BiCHAT model. It starts with a description of the datasets used in the evaluation of the proposed model including the data crawling and preprocessing procedures.

3.1. Data gathering and preprocessing

This study uses three publicly available benchmark Twitter datasets to evaluate the proposed model. Founta et al. (2018) provided the first dataset, whereas we received the second dataset from Kaggle. We received the third dataset by Davidson et al. (2017). We received the three datasets in raw format and then applied the pre-processing steps to filter the noisy and irrelevant content. All the duplicate tweets are filtered first because they do not feed any information to the model. In the pre-processing step, various Twitter-specific noises and symbols such as retweets (RT), mentions (@), hashtags (#), and URLs are filtered first. We also filter the username from the tweet. Further, the alphanumeric symbols like ampersands, dots, commas, non-ASCII characters, and stop words are filtered to avoid noisy content. Finally, the pre-processed tweets are converted to lower case to avoid ambiguity.

3.2. `BiCHAT` model

Fig. 1 presents a schematic representation of the BiCHAT model. It includes a BERT layer, an attention-aware deep convolutional network, and a bidirectional LSTM with an attention mechanism. The dense and output layers are the last two layers of the proposed BiCHAT model. The following subsections describe each of these layers.

3.2.1. BERT layer

Bidirectional Encoder Representations from Transformers (BERT) is a multi-layered bi-directional transformer-based language model. It is a pre-trained language model which uses bidirectional training of transformers for language modeling. In this

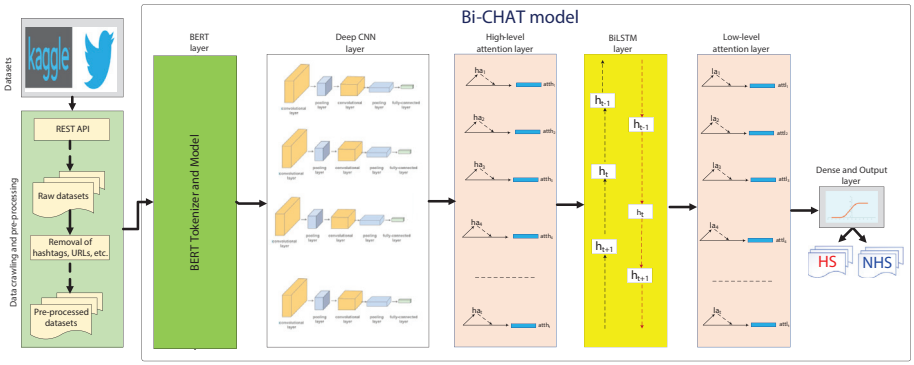

Fig. 1. Architecture of the proposed BiCHAT model for hate speech detection.

paper, we used the $BERT_{base}$ (Devlin et al., 2019) model with 12 self-attention heads to convert the tweet into vector representation format. BERT is pre-trained over BookCorpus (Zhu et al., 2015) and English Wikipedia using a masked language modeling objective. We use 768-dimensional word vector representation from BERT. The input to the BERT model is a tweet corpus with a maximum sequence length of 50. The encoded representation from the BERT layer is passed to the deep convolutional layer.

3.2.2. Deep convolutional layer

In the existing literature, researchers have used content to find hate speech signals employing different modeling techniques (Le and Mikolov, 2014). The proposed model processes the input tweets, encoded using BERT-based contextual embedding, using a deep convolutional neural network (CNN) to extract useful local and spatial features. CNN is a special class of neural network to process grid datasets like images and texts represented as matrix (Yin et al., 2017). It is useful in extracting local and position-invariant features. Initially, CNN was used to process images, but it is now actively used in text-related problems. In CNN, two operations – *convolution* and *pooling* are performed to extract local important features respectively. The convolution operation on the input data extracts high-level *feature map* by applying filters of different sizes and further pooling operation on *feature map* to extract important features. In a deep convolutional network, higher layers capture rich and complex features by applying convolutional operation on lower layers (Roy et al., 2020). The study applies 1D convolutional operation because word representation is a row vector. We use a deep CNN consisting of 6-layers to extract more efficient and complex features. Each of the six convolutional layers uses 256 filters of size 3 to extract the spatial and position-invariant features. We also perform max pooling operation of pool size 3 after the third and sixth CNN layers. In a CNN layer, all the 256 filters perform the convolutional operation on input text from top to bottom and extract the feature sequence as $f_n = [f_1, f_2, \ldots, f_{256}]$. The n^{th} feature sequence, f_n, generated from word window x_t is given in Eq. 1. Finally, the model concatenates the filter outputs to generate the resultant feature representation, which is injected into a high-level attention layer to assign variable weight to different features depending on their importance in the input text.

$$f_n = f(w_t \cdot x_t + b) \tag{1}$$

3.2.3. High-level attention layer

The model passes the encoded representation from the deep CNN layer to the high-level attention layer to assign an importance-based proportional score. The hidden representation, f_n, of a feature f is passed to a feed-forward neural network to learn f'_n, an encoded representation, using Eq. 2. Further, the dot product is applied between f'_n and a high-level context tensor v_h to compute the similarity. Finally, the attention score, α_f, of f is calculated using the softmax function as given in Eq. 3. The vertex tensor v_h is randomly initialized and jointly learned during the training process (Yang et al., 2016). Finally, attention-based representation, F_n, given using Eq. 4 of feature vector f_n is the weighted sum of hidden representations.

$$f'_n = \tanh(wf_n + b) \tag{2}$$

$$\alpha_f = \frac{exp(f'_n v_h)}{\sum_f exp(f'_n v_h)} \tag{3}$$

$$F_n = \sum_f (\alpha_f f_n) \tag{4}$$

3.2.4. BiLSTM layer

The output from the high-level attention layer is passed to a BiLSTM layer to learn the long range contextual dependencies. BiLSTM is a type of recurrent neural network and an improvement over LSTM network. It contains memory block to process the sequential information for temporal behavior modeling (Hochreiter and Schmidhuber, 1997). BiLSTM does not suffers with vanishing gradient problem. It contains memory cells to decide what to remember and what to forget and this functionality provide it the power to learn long range contextual information. An LSTM cell consists of *input gate* i_t, *forget gate* f_t, *output gate* o_t and a memory cell state c_t. The *input gate* at timestamp t, i_t, controls the flow of information in a cell and update its state to a new value using Eq. 5 whereas *forget gate* decides the amount of information to be erased at time t using Eq. 6. The Eq. 7 computes the candidate cell value, $\tilde{C}_t$. Similarly, the current cell state value C_t, output o_t from *output gate* and final output h_t of LSTM cell at time t is computed using Eqs. (8)–(10), respectively. In these equations, F_t, represents the input for the BiLSTM at time-stamp t obtained from the high-level attention, whereas W, b, σ and tanh represent the weight vector, bias vector, sigma function, and hyperbolic tangent function, respectively. Moreover, $\otimes$ performs element-wise multiplication.

$$i_t = \sigma(W_i \cdot [h_{t-1}, F_t] + b_i) \tag{5}$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, F_t] + b_f) \tag{6}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, F_t] + b_C) \tag{7}$$

$$C_t = F_t \otimes C_{t-1} + i_t \otimes \tilde{C}_t \quad (8)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, F_t] + b_o) \quad (9)$$

$$h_t = o_t \otimes \tanh(C_t) \quad (10)$$

We use BiLSTM rather than LSTM to capture the contextual information in both directions. BiLSTM has a pair of LSTM, wherein forward LSTM executes the sequence from left to right to capture the future/upcoming context and backward LSTM executes the sequential information from right to left to capture historical context. This process generates two hidden representations $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ as given in Eqs. 11 and 12 respectively. Further, BiLSTM concatenates the information from both the LSTM networks to compute the final representation as shown in Eq. 13. The proposed model uses a single BiLSTM layer to retrieve the backward (i.e., F_1 to F_{256}) and forward feature sequences (i.e., F_{256} to F_1). The encoded representation from BiLSTM incorporates the hate-related contexts from both directions. The proposed model passes this encoded information to an attention layer for variable weight assignment.

$$\overleftarrow{h_i} = \overrightarrow{LSTM}(F_n) \quad (11)$$

$$\overrightarrow{h}_i = \overleftarrow{LSTM}(F_n) \quad (12)$$

$$h_n = \left[\overleftarrow{h_i}, \overrightarrow{h}_i\right] \quad (13)$$

3.2.5. Low-level attention and dense layer

The output vector h_n from the BiLSTM layer is given to an attention layer to assign a weight to each feature component. The low level attention is applied using the Eqs. 14 and 15, where h'_n represents the encoded representation of BiLSTM layer output h_n following the passing from a feed-forward neural network, v_l represents the low-level context vector, and α_c represents the low-level attention score. The low-level attention mechanism generates the output vector $\mathscr{F}_n$, which is passed through a dense layer with a dropout of 0.3. Finally, the dense layer output passes through a sigmoid non-activation function to classify the input tweets into hate and non–hate classes.

$$h'_n = \tanh(wh_n + b) \quad (14)$$

$$\alpha_c = \frac{exp\left(h'^{T}_n v_l\right)}{\sum_{\beta} exp\left(h'^{T}_n v_l\right)} \quad (15)$$

$$\mathscr{F}_n = \sum_c \alpha_c h_n \quad (16)$$

4. Experimental setup and results

This section evaluates the proposed model over three benchmark datasets, including its description, experimental parameter setting, evaluation metrics, and underlying results. We also compare the proposed model with several existing SOTA and baseline methods.

4.1. Datasets

We evaluate the BiCHAT model over three Twitter-related benchmark datasets. The first dataset, HD1, was provided by Founta et al. (2018), who used the CrowdFlower platform to annotate a total of 80 k tweets into *abusive*, *hateful*, *spam*, or *normal* labels. This dataset releases only the tweet-id due to privacy issues. From the labeled dataset, we chose the given 3635 *hateful* and 52835 *normal* labeled tweet-ids and crawled the underlying tweets using a developed crawler to construct HD1. The crawler crawls only 2615 hateful and 35900 normal tweets because the remaining tweets are either deleted or not accessible due to privacy issues. Finally, 15% of the normal tweets – 5385 is selected to construct the final HD1 dataset, having 2615 and 5385 hateful and normal tweets, respectively. The second dataset, HD2, is constructed from a Kaggle competition dataset having 31962 tweets consisting of 2242 hate and 29720 normal tweets. The preprocessing filters the tweet having less than 5 words. As a result, we have 1421 hateful and 19150 normal content. Finally, we randomly select 50% of tweets from the normal category to construct HD2 having 1421 hate and 10575 normal tweets. The third dataset is provided by Davidson et al. (2017), who used CrowdFlower workers to annotate a sample of 24802 tweets into three categories – hate speech, offensive, and neither. The annotated dataset contains 1430 hate speech, 19190 offensive tweets, and 4163 normal tweets. Finally, we select the hate speech and normal tweets to construct the third dataset, HD3. Table 1 shows a brief statistic of HD1, HD2, and HD3, where #Hate tweet and #Normal tweet represent the number of hate and normal tweets, respectively. We can observe from the table that HD1 and HD3 are relatively balanced, whereas HD2 is unbalanced.

4.2. Experimental and hyperparameter settings

We performed all the experimental evaluations using Python 3.7.12 with Keras 2.7.0 on a Google Colab. We crawled the tweets using Tweepy, an in-built library. We evaluated the proposed model using 5-fold cross-validation where the dataset is divided into 5 equal parts. Further, the model is trained using four parts, and the remaining part evaluates the model. The whole process is repeated five times so that every instance of the data is used in both the training and validation. We trained and validated the model using the 20 epoch by computing the average value of evaluation metrics over all the epochs. During the experimentation, the *batch size* and *optimization algorithm* is adjusted at 32 and *Adam*, respectively. In the proposed model, the deep convolutional network has six layers wherein each layer has 256 filters of size 3 followed by a max-pooling operation of 3 after the third and last CNN layer. The proposed model also uses a BiLSTM layer with 256 memory cells. The sigmoid layer having 2 neurons takes the output from the dense layer for classification. The proposed model uses categorical cross-entropy as a loss function. Table 2 presents the values of all the hyperparameters used in the experimental evaluation.

4.3. Performance evaluation metrics

The performance of a machine learning model is generally evaluated using precision (Pr), recall (Rc), f-score (Fs), and accuracy (AC). In this study, we also evaluate the efficacy of the BiCHAT model using these four evaluation metrics. In the context of this paper, precision refers the percentage of correctly classified hate tweets to the total hate classified tweets as shown in Eq. 17. On

Table 1
Statistics of the datasets.

Datasets	#Hate tweet	#Normal tweet	Total
HD1 (Relatively balanced)	2615	5385	8000
HD2 (Unbalanced)	1421	10579	12000
HD3 (Unbalanced)	1430	4162	5592

Table 2
Hyperparameters used in the BiCHAT model.

Hyperparameter	Value
embedding dimension	768
Maximum sequence length	50
CNN filter size	3
# CNN Filters	256
# neurons in BiLSTM	256
Optimization algorithm	Adam
Dropout	0.3
batch size	16

the other hand, recall represents the percentage of correctly classified hate tweets from total labeled hate tweets. Its mathematical formulation is defined in Eq. 18. F-score is the harmonic mean of precision and recall as given in using Eq. 19. Finally, accuracy represents the percentage of correctly classified tweets from all the labeled tweets as defined in Eq. 20. We represent the training and validation accuracy using AC_T and AC_V

$$Pr = \frac{TruePositive}{TruePositive + FalsePositive} \tag{17}$$

$$Rc = \frac{TruePositive}{TruePositive + FalseNegative} \tag{18}$$

$$Fs = \frac{2 \times Pr \times Rc}{Pr + Rc} \tag{19}$$

$$AC = \frac{TruePositive + TrueNegative}{\#tweets} \tag{20}$$

4.4. Experimental results

This section presents the performance evaluation results of the proposed model over the three benchmark datasets. This section also discusses the comparative evaluation of the BiCHAT with three SOTA and five baseline methods. The first row of Table 3 presents the performance of BiCHAT over the three datasets considering Pr, Rc, and Fs, whereas the results considering training and validation accuracy are in the first row of Table 4. These experimental results indicate that the BiCHAT model shows the best performance over the unbalanced dataset than the balanced datasets. We can also observe that BiCHAT consistently shows comparative performance over the training and validation datasets, which is significant in a real-life scenario.

4.4.1. Comparison with state-of-the-art and baseline methods

We also evaluate the proposed model to three existing SOTA and six baseline methods for hate speech detection. Before examining comparative results, we can find a brief description of the SOTA and baseline models in the following paragraphs.

- HCovBi-Caps (Khan et al., 2022): In this paper, author present a BiGRU, CNN and capsule network based deep learning for hate speech detection and evaluated it over both balanced and unbalanced datasets.
- Ding et al. (2019): Authors integrated the stack of BiGRU with capsule network beating the SOTA result for date speech detection.
- Roy et al. (2020): In this paper, author used a deep convolutional neural network for hate speech detection.
- ANN: We also compare the BiCHAT with a simple artificial feed-forward neural network having two hidden layers. Each of the two hidden layers has 128 neurons. Finally, a sigmoid function classifies the tweets into hate and normal classes.
- CNN: We also use a simple convolutional neural network as a baseline method to compare with the proposed model. The baseline CNN has 128 filters, each of size 3.
- LSTM: This is the third baseline for comparison with BiCHAT and uses 128 neurons for tweet representation learning.
- BiLSTM: It is a recurrent neural network where representation learning incorporates the contextual information in preceding and succeeding directions. In this baseline BiLSTM, we use 128 neurons for experimental evaluation.
- GRU: This is the last baseline and has 128 neurons for tweet representation learning.

Tables 3 and 4 shows the comparative performance evaluation results of BiCHAT with SOTA and baseline models. The best performance among all the models considering a metric is in bold typeface. We can observe from the analysis of results in Table 3 that BiCHAT shows a significant performance improvement over SOTA and baseline methods. However, the difference in performance is low over the HD3. We can observe from Table 4 that considering AC_T and AC_V, BiCHAT also outperforms the SOTA and baselines over all the three datasets except for one HD1 where it performs equally to the comparative method, HCovBi-Caps (Khan et al., 2022), in terms of training accuracy. The results show that HCovBi-Caps reports the best performance among the comparison approaches, whereas other comparison approaches show significantly poor performance. The proposed BiCHAT model outperforms the best comparison model, HCovBi-Caps, by 8%, 7%, and 8% considering Pr, Rc, and Fs, respectively over HD1. BiCHAT also outperforms HCovBi-Caps by 9% in terms of AC_V but equally considering AC_T. Similarly, BiCHAT beats HCovBi-Caps by 6%, 7% 7%, 5% 7% considering Pr, Rc, and Fs, AC_T and AC_V respectively over HD2. A similar

Table 3
Experimental results of BiCHAT over HD1 and HD2 considering Pr, Rc, and Fs.

Datasets →	HD1 (balanced)			HD2 (unbalanced)			HD3 (balanced)		
Methods ↓	**Pr**	**Rc**	**Fs**	**Pr**	**Rc**	**Fs**	**Pr**	**Rc**	**Fs**
BiCHAT	**0.88**	**0.80**	**0.84**	**0.96**	**0.87**	**0.91**	**0.74**	**0.75**	**0.75**
HCovBi-Caps (Khan et al., 2022)	0.80	0.73	0.76	0.90	0.80	0.84	0.71	0.73	0.72
Roy et al. (2020)	0.60	0.40	0.48	0.61	0.58	0.59	0.64	0.68	0.66
Ding et al. (2019)	0.64	0.60	0.61	0.65	0.61	0.62	0.56	0.60	0.58
DNN	0.45	0.37	0.40	0.69	0.31	0.42	0.50	0.46	0.48
CNN	0.55	0.32	0.40	0.65	0.35	0.45	0.50	0.40	0.45
LSTM	0.37	0.34	0.35	0.72	0.45	0.55	0.62	0.71	0.66
BiLSTM	0.64	0.38	0.47	0.83	0.67	0.74	0.60	0.72	0.66
GRU	0.48	0.41	0.44	0.75	0.53	0.62	0.50	0.58	0.55

Table 4
Experimental results in terms of AC_T and AC_V over the Datasets.

Datasets →	HD1 (balanced)		HD2 (unbalanced)		HD3 (balanced)	
Methods ↓	**AC_T**	**AC_V**	**AC_T**	**AC_V**	**AC_T**	**AC_V**
BiCHAT	**0.87**	**0.89**	**0.98**	**0.97**	**0.76**	**0.73**
HCovBi-Caps (Khan et al., 2022)	0.87	0.80	0.93	0.90	0.74	0.70
Roy et al. (2020)	0.73	0.67	0.87	0.87	0.70	0.69
Ding et al. (2019)	0.67	0.65	0.88	0.88	0.63	0.65
ANN	0.67	0.65	0.90	0.88	0.68	0.68
CNN	0.67	0.66	0.89	0.86	0.58	0.52
LSTM	0.66	0.65	0.91	0.87	0.65	0.66
BiLSTM	0.80	0.70	0.92	0.89	0.66	0.65
GRU	0.68	0.64	0.90	0.86	0.63	0.66

Table 5
Ablation analysis considering **AC_T** and **AC_V** over the three datasets.

Datasets →	HD1 (balanced)		HD2 (unbalanced)		HD3 (balanced)	
Methods ↓	**AC_T**	**AC_V**	**AC_T**	**AC_V**	**AC_T**	**AC_V**
BiCHAT	0.83	0.89	0.98	0.97	0.76	0.73
BiCHAT (without deep CNN)	0.80	0.78	0.91	0.90	0.73	0.71
BiCHAT (without BiLSTM)	0.81	0.83	0.93	0.93	0.71	0.67
BiCHAT (without attention)	0.83	0.84	0.93	0.96	0.74	0.76

improvement in results is over HD3 from both tables. The analysis of the results among baseline approaches indicates that BiLSTM shows the best performance, whereas ANN and LSTM show poor performance over HD1 and HD2. However, over HD3, ANN and CNN perform poorly. Another observation among the baseline methods is that BiLSTM consistently outperforms over three datasets. It is due to the potential of bi-directional RNN models in retrieving efficient sequential features from both directions. The results of baseline approaches justify the use of BiLSTM in the proposed model. The improved performance of the proposed model signifies that the applied deep CNN layers extract efficient complex spatial and position-invariant features, useful in hate speech content classification. The following section evaluates the relative importance of each of the deep learning components.

4.4.2. Ablation analysis

The proposed deep learning model integrates three neural network components – deep CNN (6 layers), BiLSTM, and two attention layers. This section performs ablation analysis by excluding neural network components to analyze the impact of each of the three neural network components on the proposed model. In ablation analysis, we examine the efficacy of a particular deep learning component by excluding it from BiCHAT and analyze the change in the evaluation result. For example, to evaluate the effect of the CNN component, the deep CNN consisting of six layers is excluded, resulting in an updated model having a BERT layer followed by the high-level attention, BiLSTM, and low-level attention layers. The second row of Table 5 presents the evaluation results of the updated model. We follow the same procedure to construct two other deep learning models by removing BiLSTM and attention components. The removal of the BiLSTM layer builds a deep learning model having the BERT layer followed by six CNN layers and one attention layer. Similarly, the exclusion of high- and low-level attention layers creates a model having the BERT layer followed by six CNN layers & a BiLSTM layer. Table 5 presents the evaluation results for the constructed deep learning models over the three datasets. We can observe from the table that the exclusion of six CNN layers has the highest impact on model performance over HD1 and HD2. However, the exclusion of BiLSTM has the highest impact on HD3. Further, removing the attention mechanism shows minimal impact across all the datasets and even improves the validation accuracy over HD3. Based on the ablation analysis result, we conclude that each neural network component is vital in the proposed integrated BiCHAT model.

5. Impact analysis of hyperparameters

A deep learning model has various hyperparameters – batch size, embedding method, activation function, optimization algorithms, which affect the model performance. In this section, we present the evaluation results of experimental analysis to observe the impact of *embedding methods*, *activation functions*, *batch size*, and *optimization algorithms* on the performance of BiCHAT model over the three datasets. This evaluation is performed considering AC_T and AC_V.

5.1. Embedding method

In a deep learning model, a word/phrase/document is represented as a dense vector called embedding employing different language models like BERT (Devlin et al., 2019), GloVe (Pennington et al., 2014, word2vec Mikolov et al., 2013). The embedding or distributional representation of a word is a dense vector, incorporating the contextual semantics of the word. In an embedding vector, each value represents a latent concept based on word co-occurrence in the context. In this study, we use BERT to encode each word of the input text and further evaluate the performance of the BiCHAT over GloVe and word2vec. Table 6 presents the experimental results of BiCHAT using the three language models – BERT, GloVe, and word2vec over the three datasets. On analysis, we conclude that on all the three datasets, BERT shows the best performance among the three representation methods considering both AC_T and AC_V except for one instance of AC_T over HD3 where BERT and GloVe show equal performance. We can also observe that BERT shows the best performance, whereas word2vec shows the worst. The best performance using the BERT embedding ascertains the efficacy of context-incorporating representation and superiority of BERT over the context-independent embedding techniques like GloVe, word2vec. Hence, we used BERT-based contextual embedding representation in the proposed BiCHAT model.

Table 6
Experimental results of the BiCHAT model using different dimensions of GloVe embedding over the three datasets.

Datasets →	HD1 (balanced)		HD2 (unbalanced)		HD3 (balanced)	
Methods ↓	$\mathbf{AC}_T$	$\mathbf{AC}_V$	$\mathbf{AC}_T$	$\mathbf{AC}_V$	$\mathbf{AC}_T$	$\mathbf{AC}_V$
BERT	0.83	0.89	0.98	0.97	0.76	0.73
GloVe	0.74	0.84	0.94	0.93	0.76	0.71
word2vec	0.77	0.82	0.95	0.93	0.72	0.70

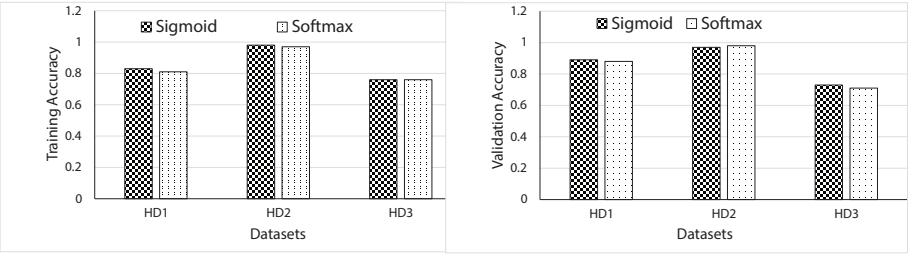

Fig. 2. Experimental results of BiCHAT using *sigmoid* and *softmax* over the three datasets considering (a) $\mathbf{AC}_T$ (b) $\mathbf{AC}_V$.

5.2. Activation functions

In a deep learning model, an activation function is responsible for neuron activation. Therefore, we evaluate the performance of the proposed model employing two activation functions – sigmoid and softmax, considering training and validation accuracy. Fig. 2 shows the underlying performance evaluation results over the three datasets considering both the activation functions. The figure shows that the performance of the BiCHAT is approximately similar using both the activation functions. However, BiCHAT performance using sigmoid is better in comparison to the softmax. The BiCHAT shows better performance using the sigmoid function because it is designed for binary classification systems, whereas softmax is better for multi-class classification systems. Hence, the proposed model uses the sigmoid as the activation function.

5.3. Batch size

The *batch size* refers to the number of samples propagated through a deep learning model at a time. Suppose a dataset has 1000 instances and the batch size is 50, then the first 50 instances will be taken and passed through the network to train it, then the next 50 will be passed, and this process continues till the dataset is exhausted. The *batch size* is also a hyperparameter and affects the performance of the underlying model. We evaluate the BiCHAT using 16, 32, 64, and 128 batch sizes to observe its impact on the performance of the BiCHAT model. Fig. 3 shows the corresponding experimental results over the three datasets. The figure shows that over the balanced dataset HD1, the BiCHAT shows the best performance considering validation accuracy using the 32 batch size. Over the HD2 and HD3 datasets, the BiCHAT outperforms using

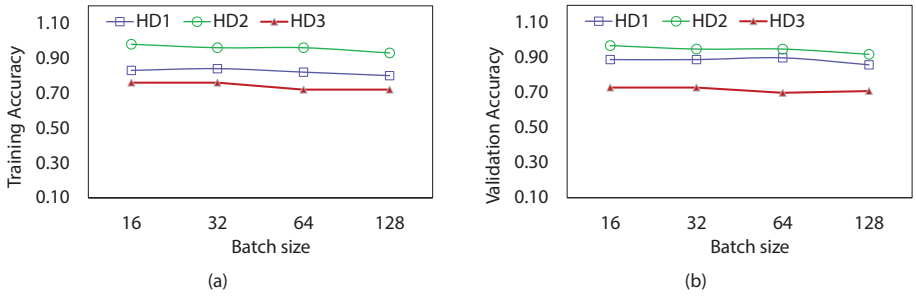

Fig. 3. Experimental results of BiCHAT using various batch sizes over the three datasets considering (a) $\mathbf{AC}_T$ (b) $\mathbf{AC}_V$.

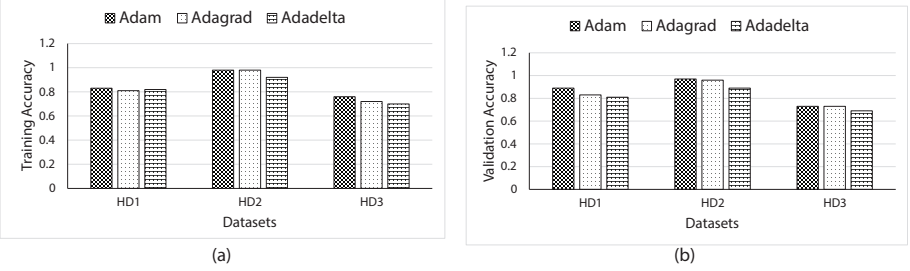

Fig. 4. Experimental results of BiCHAT using *Adam*, *Adagrad* and *Adadelta* over the three datasets considering (a) $\mathbf{AC}_T$ (b) $\mathbf{AC}_V$.

16 batch size in terms of both AC_T and AC_V. Thus, the performance evaluation over different batch sizes ascertains the use of 16 batch size in the BiCHAT model.

5.4. Optimization algorithms

The selection of an optimization algorithm is also significant among the various hyperparameters used in a neural network model. Thus, we will analyze the impact of various optimization algorithms on the performance of the BiCHAT model. To this end, we perform the experimental evaluation using three different optimization algorithms – Adam, Adagrad, and Adadelta. Fig. 4 shows the experimental results over the three datasets considering AC_T and AC_V. We can observe from the figure that Adam consistently shows the best performance over all the three datasets considering AC_T and AC_V. However, Adam and AdaGrad show equal performance in two instances - over HD2 in term of AC_T and over HD3 in terms of AC_V. The figure also indicates that Adam shows much better performance than AdaGrad and Adadelta considering AC_V than AC_T. Therefore, based on the analysis of the results, we choose Adam as an optimization algorithm in the proposed model.

6. Conclusion and future work

We proposed a novel deep learning model, BiCHAT, integrating the BERT-based contextual embedding with a deep convolutional network, BiLSTM, and hierarchical attention mechanism for hate speech detection. Unlike existing models, the BiCHAT incorporates the strength of context-incorporating embeddings, attention mechanism with deep CNN, and BiLSTM to learn the spatial features and long-range contextual dependencies. The proposed model has been evaluated over three benchmark Twitter datasets – HD1, HD2, and HD3. The experimental evaluation showed that the BiCHAT model outperformed the SOTA and baseline methods. We also performed an ablation study to investigate the efficacy of various neural network components used in the proposed model. We also investigated the impact of neural network hyperparameters on the performance of the BiCHAT model.

The BiCHAT model lacks sentiment, content, and other profile-related features. The evaluation of the BiCHAT over various datasets is also a fascinating research direction. The extension of the proposed model to a multi-modal method is another important direction of research. The extension of BiCHAT to classify the multilingual and code-mixed hate content is another direction of research. In terms of real-life application, the proposed model can be used by OSN service providers for hate speech detection, which will also establish the efficacy of the proposed model over the real dataset.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors extend their appreciation to the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University for funding this work through Research Group No. RG-21-07-08.

Consultant Works

We would also like to thank Prof. Ayub Khan for serving as a consultant to critically reviewed the study proposal and participated in technical editing of the manuscript.

References

Abulaish, M., Kamal, A., 2018. 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' 18), Santiago, Chile, IEEE. pp. 574–579.

Abulaish, M., Kumari, N., Fazil, M., Singh, B., 2019. A graph-theoretic embedding-based approach for rumor detection in twitter. In: Proc. of the WI. ACM, Thessaloniki, Greece, pp. 466–470.

Abulaish, M., Fazil, M., Anwar, T., 2020. A contextual semantic-based approach for domain-centric lexicon expansion. In: Proceedings of the 31st Australian Database Conference. Springer, Melbourne, Australia, pp. 216–224.

Abulaish, M., Fazil, M., Zaki, M.J., 2022. Domain-specific keyword extraction using joint modeling of local and global contextual semantics. ACM Trans. Knowl. Discovery Data 16 (4), 1–30.

Akhtar, M.M., Zamani, A.S., Khan, S., Shatat, A.S.A., Dilshad, S., Samdani, F., 2022. Stock market prediction based on statistical data using machine learning algorithms. J. King Saud Univ.- Sci. 34 (4), 1–15.

Badjatiya, P., Gupta, S., Gupta, M., Varma, V., 2017. Deep learning for hate speech detection in tweets. In: Proceedings of the 26th International Conference on World Wide Web Companion. ACM, Perth Australia, pp. 759–760.

Burnap, P., Williams, M.L., 2015. Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. Policy Internet 7 (2), 223–242.

Davidson, T., Warmsley, D., Macy, M., Weber, I., 2017. Automated hate speech detection and the problem of offensive language. In: Proceedings of the 11th International AAAI Conference on Web and Social Media, (ICWSM' 17), Montréal, Canada, AAAI, May 15–18. pp. 512–515.

Davis, C.A., Varol, O., Ferrara, E., Flammini, A., Menczer, F., 2016. Botornot: A system to evaluate social bots. In: Proc. of the WWW. ACM, Montreal, Canada, pp. 273–274.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT, ACL. pp. 4171–4186.

Ding, Y., Zhou, X., Zhang, X., 2019. Ynu_dyx at semeval-2019 task 5: A stacked bigru model based on capsule network in detection of hate. In: Proceedings of the 13th International Workshop on Semantic Evaluation, (SemEval' 19), Minneapolis, Minnesota, USA, ACL, June 6–7, 2019. pp. 535–539.

Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., Bhamidipati, N., 2015. Hate speech detection with comment embeddings. In: Proceedings of the 24th International Conference on World Wide Web Companion, ACM, Florence, Italy. pp. 29–30.

Fazil, M., Abulaish, M., 2018. A hybrid approach for detecting automated spammers in twitter. IEEE Trans. Inf. Forensics Secur. 13 (11), 2707–2719.

Fazil, M., Sah, A.K., Abulaish, M., 2021. Deepsbd: A deep neural network model with attention mechanism for socialbot detection. IEEE Trans. Inf. Forensics Secur. 16 (8), 4211–4223.

Fortuna, P., Soler-Company, J., Wanner, L., 2021. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? Inf. Process. Manage. 58 (3), 1–17.

Founta, A.-M., Djouvas, C., Chatzakou, D., Leontiadis, I., Blackburn, J., Stringhini, G., Vakali, A., Sirivianos, M., Kourtellis, N., 2018. Large scale crowdsourcing and characterization of twitter abusive behavior. In: Proceedings of the 12th International AAAI Conference on Web and Social Media, (ICWSM' 18), Stanford, California, USA, AAAI, June 25–28. pp. 491–500.

Haq, A.U., Li, J.P., Ahmad, S., Khan, S., Alshara, M.A., Alotaibi, R.M., 2021. Diagnostic approach for accurate diagnosis of covid-19 employing deep learning and transfer learning techniques through chest x-ray images clinical data in e-healthcare. Sensors 21 (24), 1–16.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Husain, F., Uzuner, O., 2021. A survey of offensive language detection for the arabic language. ACM Trans. Asian Low-Resource Lang. Inf. Process. 20 (1), 1–44.

Jain, P.K., Saravanan, V., Pamula, R., 2021. A hybrid cnn-lstm: A deep learning approach for consumer sentiment analysis using qualitative user-generated contents. ACM Trans. Asian Low-Resource Lang. Inf. Process. 20 (5), 1–15.

Kamal, A., Abulaish, M., 2019. 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' 19), Hyderabad, India, NLPAI. pp. 201–210.

Kamble, S., Joshi, A., 2018. Hate speech detection from code-mixed hindi-english tweets using deep learning models. In: Proceedings of 15th International Conference on Natural Language Processing. ACL, Patiala, India, pp. 155–160.

Khan, S., Kamal, A., Fazil, M., Alshara, M.A., Sejwal, V.K., Alotaibi, R.M., Baig, A., Alqahtani, S., 2022. Hcovbi-caps: Hate speech detection using convolutional and bi-directional gated recurrent unit with capsule network. IEEE Access 10 (1), 7881–7894.

Kwok, I., Wang, Y., 2013. Locate the hate: Detecting tweets against blacks. In: Proceedings of the 27th AAAI Conference on Artificial Intelligence. AAAI, Bellevue, USA, pp. 1621–1622.

Le, Q., Mikolov, T., 2014. Distributed representations of sentences and documents. Proc. Mach. Learn. Res. 32 (2), 1188–1196.

Malmasi, S., Zampieri, M., 2017. Detecting hate speech in social media, in: Proceedings of the Recent Advances in Natural Language Processing, ACL, Varna, Bulgaria. pp. 467–472.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J., 2013. Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26 International Conference on Advances in Neural Information Processing Systems. pp. 1–9.

Mossie, Z., Wang, J.-H., 2020. Vulnerable community identification using hate speech detection on social media. Inf. Process. Manage. 57 (3), 1–16.

Pamungkas, E.W., Basile, V., Patti, V., 2021. A joint learning approach with knowledge injection for zero-shot cross-lingual hate speech detection. Inf. Process. Manage. 58 (4), 1–19.

Park, J.H., Fung, P., 2017. One-step and two-step classification for abusive language detection on twitter. In: Proceedings of the First Workshop on Abusive Language Online, ACL, Vancouver, Canada. pp. 41–45.

Pennington, J., Socher, R., Manning, C.D., 2014. Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543.

Potthast, M., Kopsel, S., Stein, B., Hagen, M., 2016. Clickbait detection. In: Proceedings of the European Conference on Information Retrieval, Springer, Cham, Padua, Italy. pp. 810–817.

Qaisar, A., Ibrahim, M.E., Khan, S., Baig, A.R., 2021. Hypo-driver: A multiview driver fatigue and distraction level detection system. CMC-Comput. Mater. Continua 71 (1), 1999–2017.

Roy, P.K., Tripathy, A.K., Das, T.K., Gao, X.-Z., 2020. A framework for hate speech detection using deep convolutional neural network. IEEE Access 8, 204951–204962.

Vigna, F.D., Cimino, A., Dell'Orletta, F., Petrocchi, M., Tesconi, M., 2017. Hate me, hate me not: Hate speech detection on facebook. In: Proceedings of First Italian Conference on Cybersecurity, CEUR-WS, Venice, Italy. pp. 86–95.

Warner, W., Hirschberg, J., 2012. Detecting hate speech on the world wide web. In: Proceedings of the 2012 Workshop on Language in Social Media, ACL, Montreal, Canada. pp. 19–26.

Waseem, Z., Hovy, D., 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In: Proceedings of the NAACL-HLT, ACL, California, USA. pp. 88–93.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E., 2016. Hierarchical attention networks for document classification. In: Proceedings of International Conference on 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. ACL, California, USA, pp. 1480–1489.

Yin, W., Kann, K., Yu, M., Schutze, H., 2017. Comparative study of cnn and rnn for natural language processing, in: arXiv:1702.01923v1, arXiv. pp. 1–7.

Zhang, Z., Robinson, D., Tepper, J., 2018. Detecting hate speech on twitter using a convolution-gru based deep neural network. In: Proceedings of the European Semantic Web Conference, Springer, Cham, Heraklion, Greece. pp. 745–760.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., Fidler, S., 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In: Proceedings of the IEEE International Conference on computer vision, IEEE Computer Society. pp. 19–27.