ITC 3/53

Information Technology and Control
Vol. 53 / No. 3/ 2024
pp. 659-674
DOI 10.5755/j01.itc.53.3.35112

BiLSTM-Attention-CNN Model Based on ISSA Optimization for Cyberbullying Detection in Chinese Text

Received 2023/09/15 | Accepted after revision 2024/01/23

HOW TO CITE: Fan, W. (2024). BiLSTM-Attention-CNN Model Based on ISSA Optimization for Cyberbullying Detection in Chinese Text. *Information Technology and Control*, 53(3), 659-674. https://doi.org/10.5755/j01.itc.53.3.35112

BiLSTM-Attention-CNN Model Based on ISSA Optimization for Cyberbullying Detection in Chinese Text

Wenting Fan

School of Computer Science and Technology, Taiyuan University of Science and Technology, Taiyuan 030000, China; e-mail: wentingfan96@tom.com

Corresponding author: wentingfan96@tom.com

Cyberbullying has become increasingly common due to the extent and anonymity afforded to users by online social media, and poses a significant risk to the physical and mental health of people. In this study, we propose an ISSA-based model to detect cyberbullying in Chinese text (ISSA-BiLSTM-Attention-CNN) that can determine whether a given comment reflects cyberbullying. The model contains an attention mechanism and the improved sparrow search algorithm (ISSA) for optimization that enables it to focus on important textual information and make full use of the optimal hyperparameters. Before applying the CNN to collect and learn a sufficient number of local features, the model initially uses the bidirectional LSTM (BiLSTM) to concatenate the results of forward and backward processing of the given text. The results of experiments showed that the proposed method can outperform baseline methods, with an accuracy of 90.2% and an f-measure of 89.9%.

KEYWORDS: cyberbullying detection; attention mechanism; improved sparrow search algorithm (ISSA); bidirectional LSTM (BiLSTM); CNN.

1. Introduction

Online social media platforms, such as Weibo and TikTok, have developed rapidly in recent years as the mainstream environment for sharing and interaction. They have made communication remarkably more convenient, but have also spawned a variety of problems, one of which is growing online abuse and cyberbullying. Cyberbullying is defined as insulting or attacking online conduct toward individuals or groups in the form of inappropriate or offensive text comments and images [19], and is considered to be destructive to the physical and mental health of the victims. People who have been cyberbullied have reported experiencing anxiety, low self-esteem, and even suicidal tendencies. For instance, Zheng Linghua was a 24-year-old woman who was cyberbullied because she had dyed her hair pink. The harassment drove her to commit suicide in 2023 [31].

Cyberbullying has recently grown to become a serious problem at a global scale. A variety of approaches has been proposed to automatically detect cyberbullying. Rosa et al. [18] applied fuzzy fingerprinting technology to detect cyberbullying, and their method recorded better performance than logistic regression, naive Bayes, and the SVM. Cheng et al. used a global model to identify cyberbullying in posts by new users of platforms that considers a combination of unique characteristics as well as peer's influence [12] Ehzadi et al. [26] developed method to quickly detect cyberbullying based on transfer learning and fine-tuning the compact BERT model. It achieved impressive results on a dataset of hate speech, with an f-measure of 0.91. Wang et al. developed the FastText and word similarity (FTSW) model to detect cyberbullying [35].

It can identify latent bullying-related words by using the word similarity scheme and assess the morphological characteristics of cyberbullying texts by using schemes of FastText.

A majority of research on identifying cyberbullying has used data in the English language. Models that perform well on training sets in English may not deliver comparable results on Chinese text due to the difference in linguistic structure between English and Chinese. Datasets in English automatically contain separators while Chinese text requires word segmentation and other operations. Characters, rather than words, are processed as the basic unit of learning in the char-CNN model presented by Lu et al. [26], and features at different levels are connected together by shortcuts to detect cyberbullying. This model achieved values of recall, precision, and the f-measure of 79%, 69.8%, and 71.6%, respectively, on the Chinese Weibo dataset, and values of 81%, 70.5%, and 74.2%, respectively, on a dataset of English texts obtained from Twitter.

Using a single neural network model to detect cyberbullying incurs limitations, while a combination of methods can fully exploit the advantages of each model to extract features at various scales and enhance the accuracy of detection. Kumari et al. [21] proposed a multi-modal system for cyberbullying detection that can extract the features of images through the VGG-16 network and obtain textual features through the CNN. The combined image-related and textual features are then significantly improved by using a genetic algorithm. Raj et al. proposed an ensemble CNN-BiLSTM model for cyberbullying detection that is optimized by using stacked two-word embeddings (Glove+FastText) [9].

The hyperparameters that need to be set before creating the neural network significantly influence the performance of the deep learning model for cyberbullying detection. However, most researchers in the area have resorted to manually choosing the hyperparameters based on their expertise and numerous experiments, where this is time and labor intensive [10-15].

In this article, we propose a BiLSTM-Attention-CNN model based on ISSA optimization (ISSA-BiLSTM-Attention-CNN) to detect cyberbullying in Chinese text. The main contributions of this paper are as follows:

1. We collected a dataset of cyberbullying in Chinese from the social media platforms Weibo and TikTok, and used a character-enhanced model of word embeddings to represent each word as a vector [29].
2. We formulate the hybrid BiLSTM-Attention-CNN model to detect cyberbullying. The BiLSTM network can extract features that describe the overall reliability of each sentence based on both the normal and the reverse orders of words in it. Important features are assigned larger weights by the attention mechanism to improve the accuracy of detection. The local dependencies of nearby words are recorded by using the CNN.
3. The hyperparameters of the BiLSTM-Attention-CNN model are optimized by using an improved sparrow search algorithm to locate the most appropriate model configurations and enhance its accuracy of detection.
4. The remainder of this paper is organized as follows: Section 2 reviews past research on the automatic detection of cyberbullying, and Section 3 provides details of the proposed method. Section 4 presents the results of experiments to test it as well as a discussion and an analysis of errors incurred by it. Finally, the conclusions of this study and directions for future work in the area are summarized in Section 5.

2. Related Work

The automatic detection of cyberbullying is frequently framed as a task of classification, one of identifying comments on social media as either exemplars of bullying or not [13]. Many algorithms have been proposed to detect cyberbullying in the context of natural language processing in recent years.

Balakrishnan et al. [6] sed a machine learning algorithm to construct an automatic mechanism to detect cyberbullying based on the psychological traits of Twitter users, including their personalities, sentiments, and emotions. They used four sets of features—emotions, semantics, unigrams, and patterns—in the machine learning algorithm "J48graft," developed by Watanabe et al. [37], to devise a method to identify hate speech on Twitter. Bozyigit et al. [8] investigated the accuracy of several machine learning algorithms (SVM, LR, KNN, NBM, AdaBoost, and RF) in identifying cyberbullying on a dataset consisting of features of social media and texts. They found that the features of social media based on the chi-squared test were crucial for enhancing the accuracy of detection of cyberbullying.

Deep learning has also been frequently used on tasks of cyberbullying detection in recent years. Alhloul et al. [1] used the convolutional layer and the attention layer to extract the pertinent features from a long sequence of tweets, and then used the multi-layer perceptron to identify comments representing cyberbullying. Areej et al. [13] proposed a method to automatically classify hate speech in tweets in Arabic, and compared its results with those of four deep learning models and an SVM-based baseline model. Sadiq et al. [30] proposed a system that yields a high accuracy with the minimal number of layers and incurs a short training time by combining the CNN with the LSTM and BiLSTM networks. The latter networks extract features that describe the long-distance dependency of each sentence, while the CNN is used to identify local relationships between neighboring words. TGBlly, a temporal, graph-based cyberbullying detection system that focuses on user interaction, was proposed by Ge et al. [16]. Wu et al. [38] updated the TF-IDF technique with positional weights and used FastText to build a binary classifier to identify phrases reflecting cyberbullying. Paul et al. [28] fine-tuned the BERT for the task of cyberbullying detection and reported a high accuracy.

Research on identifying cyberbullying has begun incorporating multi-modal data, i.e., text, images, and video clips, in recent years. Kumar et al. proposed an all-in-one deep learning neural network architecture consisting of a CNN (ConvNet) and a capsule network (CapsNet) [20]. CapsNet is based on dynamic routing, and can identify cyberbullying in texts, while ConvNet is used to detect cyberbullying in images. Cheng et al. [11] veloped XBully, a cutting-edge framework for detecting cyberbullying. This model isomorphizes various categories of features to generate a more complete feature representation of bullying behavior by using the correlations and structural dependence of multi-modal data. The system developed by Kumari et al. [22] combines the extraction of textual features by using a three-layer CNN with that of features of images through pre-trained models. They constructed a set of mixed features and then optimized its performance in terms of identifying cyberbullying in multi-modal data by using the firefly optimization algorithm.

Constructing an efficient deep learning neural network relies on many factors, one of which is the choice of appropriate hyperparameters. The selection of hyperparameters usually necessitates professional knowledge and experiments with many configurations of parameters of the network [3]. Many studies have used manual search to find the appropriate values of the hyperparameters, while others have used intelligent swarm optimization algorithms to automatically choose them.

Wang et al. [36] proposed cPSO-CNN, an improved particle swarm optimization algorithm, to optimize the hyperparameters of structure-definite CNNs. Singh et al. [33] developed a multi-level particle swarm optimization (MPSO) approach to simultaneously determine the architecture and values of the hyperparameters of a CNN. Lee et al. examined the capability of the genetic algorithm (GA) to identify the best CNN architecture, from a set of CNNs, consisting of an activation function and an optimization algorithm [24]. The GA has been applied to tune the hyperparameters of the LSTM network and identify the ideal settings of a collection of parameters for the task of next word prediction in NLP [17]. Badawy et al. [12] posed a two-stage CNN model to classify renal illnesses based on transfer learning, and used hyperparameters tuned by the SSA to improve its performance. An SSA-BLS model was developed by Li et al. [25] to forecast interface traffic in the network. The model uses the SSA to optimize two hyperparameters and quickly determine the ideal set of hyperparameters to improve the results of the BLS.

Baghdadi et al. used a CNN, a pre-trained CNN model, and the sparrow search technique to automatically and precisely identify the occurrence of COVID-19 from CT images of the lungs of patients [5]. They used the SSA to obtain the optimal hyperparameters for various CNNs and the transfer learning method to determine the ideal configuration of the network and improve its performance. Tabatabaei et al. tuned the artificial neural network by using the SSA and compared its performance with that of the PSO method [34], with an emphasis on the weights and optimization of the hyperparameters. The SSA recorded faster convergence and higher accuracy in all aspects of optimization, where this demonstrated its promise for used in optimizing neural networks.

The above review shows that many studies have been devoted to the automatic detection of cyberbullying based on deep learning algorithms, where the performance of the system can be improved by using the appropriate hyperparameters. In this paper, we propose a deep neural network model for identifying cyberbullying in Chinese text based on ISSA optimization.

3. Proposed Method

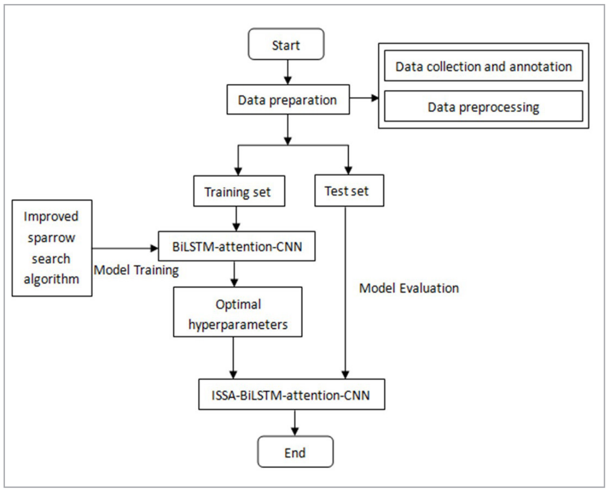
3.1. Overview

We describe our proposed BiLSTM-Attention-CNN model for identifying cyberbullying in Chinese text based on ISSA optimization (ISSA-BiLSTM-Attention-CNN model) in this section. The overall workflow of the model is shown in Figure 1.

3.2. Data Preparation

No publicly available dataset on cyberbullying in Chinese is available at present. We manually collected and processed the data in two steps: data collection and annotation, and pre-processing.

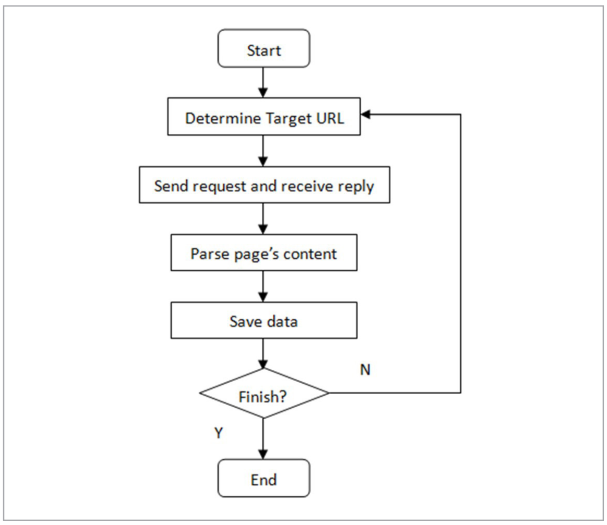
Figure 1
Overall workflow of the ISSA-BiLSTM-Attention-CNN model



3.2.1. Data Collection and Annotation

The data on cyberbullying in Chinese were gathered from the social media networks Weibo and TikTok. The basic procedure for data collection is shown in Figure 2. The primary steps included finding the target URL, sending requests and receiving replies, parsing the contents of pages, and saving the data.

Figure 2
Basic procedure used for data collection



An analysis of the collected data revealed that a majority of comments were not reflective of cyberbullying. To increase the number of comments evincing cyberbullying, we chose trending events on the two social media platforms, and targeted users with a particular reputation whose accounts also regularly received significant engagement. We targeted the relevant URLs, and accessed the network interface by using Python's requests package to construct network request headers, request IPs, user agents, and user cookies. Finally, we created a JSON string from the requested content to parse and save it.

Data annotation is the process of labeling each record in a dataset, and enables the dataset to be used for the training and learning of a deep neural network [32]. The following six categories of instances in the data were classified as those of cyberbullying according to the principle of data annotation provided in [26]:

1. comments containing plainly obscene language or cursing,
2. comments evincing discrimination based on gender, ethnicity, or geography in their content,
3. false information and rumors that had been made up on purpose,
4. threatening or derogatory remarks,
5. insults aimed at one's appearance, body, health, family members, etc., and
6. unflattering nicknames that had negative implications.

The comments in the dataset were assigned the labels "1" if they reflected cyberbullying and "0" if they did not. Three independent annotators labeled the texts, and the final labeling was determined by majority voting: When two or more experts had determined that the same comment was representative of cyberbullying, it was deemed to reflect this behavior. Table 1 shows several samples from the dataset, including the original comments, their labels, and translations in English.

3.2.2. Data Pre-processing

As the data were extracted from social media platforms, they contained Internet-related words, special symbols, and links that needed to be suitably pre-processed. We used the following steps to this end:

1. Data cleaning: We retained all legitimate characters under UTF-8 encoding, and removed useless punctuation, HTML tags, spaces, and other special characters by using regular expressions. The format of the regular expressions was as follows: ([\u4e00-\u9fa5\u0030-\u0039\u0041-\u005a\u0061-\u007a]).
2. Chinese word segmentation: We used the popular Chinese word segmentation toolkit Jieba, and used its results as input for the next phase.

3.3. ISSA-BiLSTM-Attention-CNN Model

The overall architecture of our ISSA-BiLSTM-Attention-CNN model is shown in Figure 3. It includes an embedding layer, a bidirectional LSTM layer, an attention layer, a convolution layer, a max-pooling layer and a fully connected layer. The embedding layer transformed the given text into a word vector, and transferred its results to the bidirectional LSTM layer and the CNN layer for long-distance feature extraction and local feature extraction, respectively. The attention mechanism was used to allocate weights to

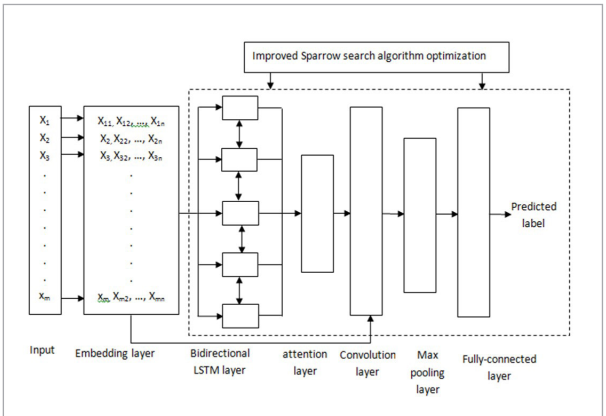
Table 1

Examples of annotated samples in the dataset

comment	label	English translation
你这是道德绑架，好吗？	1	This is a moral kidnapping, okay?
你咬牙切齿的样子真的很难看	1	The way you gnashing your teeth is really ugly
孩子是父母的命，心疼这位妈妈	0	Children are the parents' lives. I feel sorry for this mother
阿姨加油，不要在意恶言恶语我们会继续支持你	0	Come on, Auntie; don't worry about the abusive language. We'll continue to support you
糖水爷爷挺过了疫情，却没有挺过人心	0	Grandpa Tangshui has survived the COVID-19, but has not withstood the public opinion damage
真是倒霉，刷到你这种人	1	it's really incredibly unlucky to browse someone like you

the features after the BiLSTM layer in order to identify the important ones. The outputs of the embedding layer and the BiLSTM-Attention layer were concatenated, and were transmitted to the CNN to glean more information from them. The dimensionality of its output was in turn reduced by the max-pooling layer to reduce computational complexity. The fully connected layer integrated the long-distance features with local information to provide a global result. The latter was used to determine the probability of the given text falling into a certain category, and to yield the final classification as the output.

Figure 3
Overall architecture of the ISSA-BiLSTM-Attention-CNN model



3.3.1. Embedding Layer: Character and Word Embedding (CWE)

Chinese is a typical language, in the sense that any given word in it is usually composed of several characters. The meanings of individual characters that constitute a word are also connected to the meaning of the latter. Character embedding is useful for the vector representation of words as it includes semantic information in word representations [15]. We used character and word embedding (CWE) to represent all words in the vector space and develop the model of the classifier. The representation of the CWE vector of each word x_j in the given context is as follows:

$$x_j = \frac{1}{2}\left(w_j + \frac{1}{N_j}\sum_{K=1}^{N_j} c_k\right), \quad (1)$$

where w_j represents the word embedding of x_j, N_j represents the number of characters in x_j, and c_k represents the character embedding of the k-th character in x_j. Then the multiplier of 1/2 is used to assign the equal weight to word and character embedding to obtain vectorized representations of all words in dataset.

3.3.2. Bidirectional LSTM Layer

The bidirectional LSTM layer receives word vectors as input and extracts long-distance features from them [14, 27]. Its main principle is that every processing sequence consists of two LSTM units that are coupled in the forward and backward directions to gather information from opposing directions, as illustrated in Figure 4.

Figure 4
Structure of the bidirectional LSTM

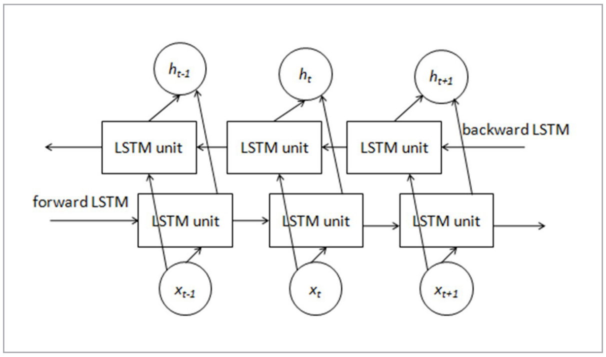


Figure 5
Structure of the LSTM unit

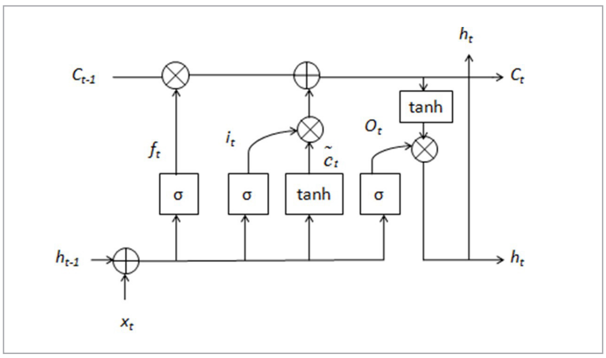


Figure 5 shows the structure of each LSTM unit, which consists of an input gate, a forget gate, an output gate, and a cell state. These gates enable models of the LSTM network to "remember" important information over a range of time steps in a selective manner. Given the word vector x_t, the previous hidden state h_{t-1}, and the previous cell state c_{t-1} in each time step t, the current state can be calculated as follows:

$$f_t = \sigma\left(w_{fx}x_t + w_{fh}h_{t-1} + b_f\right) \tag{2}$$

$$i_t = \sigma\left(w_{ix}x_t + w_{ih}h_{t-1} + b_i\right) \tag{3}$$

$$\tilde{c}_t = \tanh\left(w_{cx}x_t + w_{ch}h_{t-1} + b_c\right) \tag{4}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \tilde{c}_t \tag{5}$$

$$o_t = \sigma\left(w_{ox}x_t + w_{oh}h_{t-1} + b_o\right) \tag{6}$$

$$h_t = o_t \circ \tanh\left(c_t\right), \tag{7}$$

where σ is the sigmoid activation function and $\tanh$ represents the hyperbolic tangent function. w_{*x} represents the weight matrices of the input vector x_t, w_{*h} denotes those of the vector of the previous hidden state h_{t-1}, and b_* is the bias vector. The input, forget, and output gates, designated by i, f, o, respectively, determined whether the input data should be retained to update the state of the cell, whether the prior memory should be retained or forgotten, and whether to output the contents of the memory.

The output of the forward LSTM is denoted by $\overrightarrow{h_t}$. The reverse sequence $S = \left[x_m, x_{m-1}, \ldots, x_1\right]$ is used as the input to the backward LSTM network and $\overleftarrow{h_t}$ is its output. The final BiLSTM model is given by $H = \left[h_1, h_2, \ldots, h_m\right]$, where $h_t = \left[\overrightarrow{h_t}, \overleftarrow{h_t}\right]$.

3.3.3. Attention Mechanism

The attention mechanism of the neural network [40-41] signs different weights to various features. Larger weights are assigned to information that has a stronger impact on the performance of the model in terms of identifying cyberbullying. The accuracy of identification can be significantly increased in this way. We integrate attention mechanism into the outputs of BiLSTM. For each h_t, attention weight α_t can be calculated as follows:

$$score(h_t) = \tanh\left(w_t h_t + b_t\right) \tag{8}$$

$$\alpha_t = \frac{\exp\left(score\right)}{\sum_t \exp\left(score\right)} \tag{9}$$

$$T = \sum_t \alpha_t h_t, \tag{10}$$

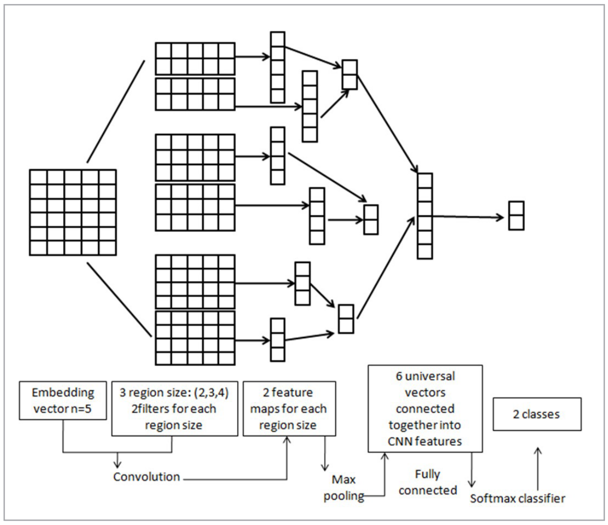
where w_t and b_t denote the weight matrix and bias vector, respectively. $score$ denotes the attention value, which will be turned into a probability distribution α_t by using Softmax function, as shown in Equation (9). Finally, each h_t has been assigned the weight of α_t, and added together to obtain the final text feature vector T.

3.3.4. Convolutional Layer

The combination of the outputs of the embedding layer and the BiLSTM-Attention layer is fed into the CNN to obtain local information. Figure 6 shows the structure of the CNN model.

Figure 6

Structure of the CNN model



Convolution operations are performed on vectors of texts by using multiple filters of various sizes, and the most important combinations of features are extracted by using max pooling. The fully connected layer receives these features as the input, and sends its output to the softmax classifier to determine the final outcome of classification as shown in Equation (11):

$$y = soft\max(w_{fc}F + b_{fc}), \tag{11}$$

where F represents the connected feature, w_{fc} is the weight of the fully connected layer, b_{fc} is bias, and y is the anticipated outcome of the classification. By normalizing the input data, the softmax function can convert them into values of probability between zero and one. The forecasted outcome is then the category with the highest probability.

3.3.5. Hyperparameter Optimization Based on ISSA

The hyperparameters that need to be chosen before constructing a neural network—namely, the size of the convolution kernel, batch size, learning rate, and dropout rate—have a significant impact on the performance of the BiLSTM-Attention-CNN algorithm. For example, convolution kernels of different sizes represent receptive fields of varying sizes, and can extract information on the features to varying degrees. Determining the hyperparameters in general requires expert knowledge and numerous experiments over a wide range of potential combinations of parameters of the network, where this is time consuming and yields unstable outcomes across domains. We use the improved sparrow search algorithm (ISSA) to automatically optimize these hyperparameters while lowering the overhead incurred by them and ensuring steady performance.

3.3.5.1. Standard Sparrow Search Algorithm

The standard sparrow search algorithm (SSA) [39] simulates the foraging behavior of sparrows to obtain the ideal possible solution from a set of potential solutions. Individual sparrows with a high fitness value act as discoverers while the others act as followers. A particular fraction of the population is chosen for scouting and early warning, called vigilantes. The discoverers locate regions with enough food for the flock over the entire search area, and offer areas or directions for foraging to the population. The followers constantly observe the discoverer's actions and update their location to increase their probability of obtaining food. The vigilantes keep an eye on the foraging site and implement anti-predation measures in case danger is detected.

The steps of the SSA are as follows:

1 Initialize the position of the sparrow population

Assume that there are m sparrows in the population, and d represents the number of dimensions of the variable to be optimized. The following matrices (12) (13) represent the initial position and the fitness value of the population, respectively.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} \end{bmatrix}, \tag{12}$$

$$f_x = \begin{bmatrix} f\left(\left[x_{11} \quad x_{12} \quad \cdots \quad x_{1d}\right]\right) \\ f\left(\left[x_{21} \quad x_{22} \quad \cdots \quad x_{2d}\right]\right) \\ \vdots \\ f\left(\left[x_{m1} \quad x_{m2} \quad \cdots \quad x_{md}\right]\right) \end{bmatrix}, \tag{13}$$

where X represents the matrix of its initial position, f is the fitness function, and fx denotes its fitness value.

2 Search for the optimal location

The optimal position of the population is obtained by iterating through the positions of the discoverers and followers. The discoverers' locations are updated as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t} * \exp\left(\dfrac{-i}{\alpha * T}\right), & R < ST \\ X_{i,j}^{t} + Q * L, & R \geq ST \end{cases}, \tag{14}$$

where i, j, and t are the i-th sparrow, j-th dimension, and t-th iteration, respectively, T indicates the maximum number of iterations, α is an arbitrary value within the interval (0, [1], and $R\left(R \in [0,1]\right)$ $ST\left(ST \in [0.5,1]\right)$ represent the warning values and the safety thresholds, respectively. Q is a random value with a standard normal distribution, and L signifies a $1*d$ matrix in that each element is set to 1.

The location of the followers is updated as Equation (15):

$$X_{i,j}^{t+1} = \begin{cases} Q * \exp\left(\dfrac{X_{worst}^{t} - X_{i,j}^{t}}{i^2}\right), & i > \dfrac{n}{2} \\ X_p^{t+1} + \dfrac{1}{D}\sum_{d=1}^{D}\left(rand[-1,1] * \left|X_{i,j}^{t} - X_p^{t+1}\right|\right), & i \leq \dfrac{n}{2} \end{cases}, \tag{15}$$

where X_{worst}^{t} denotes the globally worst position of an individual in the t-th iteration, and x_p^{t+1} indicates the best fitness-related location of an individual in the $(t+1)$-th iteration. When $i > \frac{n}{2}$, this means that the i-th follower must fly to another area to look for food. When $i \le \frac{n}{2}$, this suggests that the i-th follower is close to their best current position.

The position of the vigilant is updated as follows:

$$X_{i,j}^{t+1}=\begin{cases} X_{best}^{t}+\beta*\left|X_{i,j}^{t}-X_{best}^{t}\right|, & f_i \neq f_g \\ X_{i,j}^{t}+K*\dfrac{\left|X_{i,j}^{t}-X_{worst}^{t}\right|}{\left(f_i-f_w\right)+\varepsilon}, & f_i=f_g \end{cases}. \tag{16}$$

X_{best}^{t} represents the optimal global location in the t-th iteration. As the step length of the control parameter, β is a random value that obeys the standard normal distribution. $K\in[-1,1]$, and f_i, f_g, and f_w represent the fitness value of the current individual, and the globally best and the globally worst fitness values, respectively. ε is a constant used to avoid a zero denominator. The i-th sparrow is at the periphery of the population, and is therefore more vulnerable to an attack by the predator if $f_i \neq f_g$. When $f_i = f_g$, this implies that the i-th sparrow is at the center of the population, and continually approaches other sparrows to lessen the risk of being caught by the predator.

3.3.5.2. Improvements to the Standard SSA

1. Equation (14) shows that the algorithm is more likely to enter a local optimum when $R < ST$ causes individual discoverers to flock near the origin. Information on the upper and lower boundaries is introduced to address this issue and enhance the capability of global search of the algorithm. The candidate update of the location is as follows:

$$X_{i_2,j}^{t}=\begin{cases} rand*X_{i,j}^{t}+rand*\left(ub_j-X_{i,j}^{t}\right), & X_{i,j}^{t}\geq\left(ub_j+lb_j\right)/2 \\ rand*X_{i,j}^{t}+rand*\left(lb_j-X_{i,j}^{t}\right), & X_{i,j}^{t}<\left(ub_j+lb_j\right)/2 \end{cases}, \tag{17}$$

$rand$ is an arbitrary value within (0,1], and ub_j lb_j represent the upper and lower boundaries in j-dimensional space, respectively. The probability that a discoverer moves toward the current or candidate position is represented by the probability of movement p:

$$p=\frac{f\left(X_i\right)}{f\left(X_i\right)+f\left(X_{i_2}\right)}, \tag{18}$$

where $f\left(X_i\right)$ and $f\left(X_{i_2}\right)$ represent the fitness values of X_i and X_{i_2}, respectively.

Finally, improved updates of the locations of the discoverers are given in Equation (19):

$$X_{i,j}^{t+1}=\begin{cases} X_{i_2,j}^{t}, & rand\leq p \text{ and } R<ST \\ X_{i,j}^{t}, & rand>p \text{ and } R<ST \\ X_{i,j}^{t}+Q*L, & R\geq ST \end{cases}. \tag{19}$$

2. Equation (15) shows that if the current optimal position of an individual falls into a local optimum, additional sparrows congregate at this position. We use double-sample learning to improve the capacity of the population for globally optimizing its position. This strategy guides the followers to simultaneously search for the current optimal discoverer and any additional locations of discoverers. The improved updates of the locations of the followers are then given in Equation (20):

$$X_{i,j}^{t+1}=\begin{cases} Q*\exp\left(\dfrac{X_{worst}^{t}-X_{i,j}^{t}}{i^2}\right), & i>\dfrac{n}{2} \\ X_{i,j}^{t}+\dfrac{1}{D}\displaystyle\sum_{d=1}^{D}\left[rand\{-1,1\}*\left(r_1*\left|X_{i,j}^{t}-X_{p}^{t}\right|+r_2*\left|X_{i,j}^{t}-X_{R}^{t}\right|\right)\right], & i\leq\dfrac{n}{2} \end{cases}, \tag{20}$$

where X_R^t is the position of other randomly selected discoverer, r_1 is a random value within the interval [0, 1], and $r_2 = 1 - r_1$. When r_1 is small, this is advantageous for global search, while when r_1 is large, this is advantageous for local search.

The pseudo-code of the improved SSA is as follows:

```
Algorithm 1: The framework of ISSA
  Input:
  G:the maximum iterations
  PD:the number of discovers
  SD: the number of sparrows who perceive the danger
  R:the alarm value
  n:the number of sparrows
  Initialize a population of n sparrows and define its relevant parameters
  Output: Xbest,fg
while t<G do
    Rank the fitness values and find the current best individual and the
     current worst individual
    R=rand(1)
    for i=1:PD do
        Using equation(19) update the sparrow's location
    end
    for i=(PD+1):n do
        Using equation(20) update the sparrow's location
    end
    for k=1:SD do
        Using equation(16) update the sparrow's location
    end
    Get the current new location
    if the new location is better than before then
        update it
    end
    t=t+1
end
return Xbest,fg
```

4. Experiments and Results

We conducted experiments to comprehensively assess the effectiveness of the proposed model of cyberbullying detection.

4.1. Dataset

The manually annotated dataset of Chinese text described in Section 3.2 was used to test the proposed method of cyberbullying detection. A total of 21,782 comments were obtained through a statistical analysis of the dataset, including 4,756 samples that were representative of cyberbullying and 17,026 samples were not. The distribution of the dataset was thus considerably unbalanced because only 21.8% of it constituted comments reflective of cyberbullying.

The above imbalance can reduce the capability of generalization of the model. We used oversampling, i.e., added more samples to the class of comments reflecting cyberbullying, to address this issue. We duplicated cyberbullying-related comments in the dataset three times and randomly shuffled their order. We also added one to three randomly chosen non-bullying words and phrases before and after each comment. This led to 31,294 comments in total, of which 13,986 contained cyberbullying content and 17,308 did not after data augmentation.

Table 2 shows the distribution of the data following the division of the dataset into training and testing sets with a ratio of 7: 3.

Table 2

Distribution of the samples after the division of the dataset

	cyberbullying	Non-cyberbullying
Training set	9790	12116
Testing set	4196	5192
total	13986	17308

4.2. Evaluation Metrics

We assessed the performance of the proposed model in terms of accuracy, precision, recall, and the f-measure:

$$accuracy = \frac{Correctly \text{ predicted number of samples}}{Overall \text{ number of samples}} \tag{21}$$

$$precision = \frac{Correctly \text{ predicted number of cyberbullying samples}}{Overall \text{ predicted number of cyberbullying samples}} \tag{22}$$

$$recall = \frac{Correctly \text{ predicted number of cyberbullying samples}}{Overall \text{ actual number of cyberbullying samples}} \tag{23}$$

$$f-measure = 2*\frac{precision*recall}{precision+recall} \tag{24}$$

4.3. Hyperparameter Settings

Past work has shown that a small numbers of hyperparameters are primarily responsible for variations in the performance of algorithms [23]. The computational cost increases significantly when a large number of hyperparameters are optimized. We chose six hyperparameters of the BiLSTM-Attention-CNN model as the objective of optimization to strike a balance between its performance and computational complexity. Table 3 lists the range of values of each hyperparameter to be optimized, and Table 4 provides the constant configuration of hyperparameters of the model throughout the experiments.

Table 3

Range of hyperparameters to be optimized

Hyperparameters	Range	
	Minimum value	Maximum value
convolution kernel Number	1	128
convolution kernel Size	2	8
Stride size of convolution kernel	1	4
Batch size	8	256
Learning rate	0.0001	0.1
Neuron number in hidden layer of BiLSTM	10	300

Table 4

Constant configuration of hyperparameters of the BiLSTM-Attention-CNN model

Hyperparameters	Value
Input vector size	50
Embedding dimension	300
Number of hidden layers	2
Number of convolution layers	1
Dropout rate	0.5
epoch	100

4.4. Results and Discussions

We conducted the experiments detailed below to evaluate the performance of our system.

1 **Comparison of methods of cyberbullying detection**

Table 5 presents comparisons of several models of cyberbullying detection. We used the CNN, LSTM, and BiLSTM to identify cyberbullying, and used the results of models with an attention mechanism for comparison. A total of seven models were tested, and all of them used the representation of the CWE vector and manually configured hyperparameters.

1 CNN: Standard convolutional neural network.
2 LSTM: Standard long short-term memory neural network.
3 BiLSTM: Bidirectional LSTM.
4 BiLSTM-Attention: The addition of the attention mechanism to the BiLSTM.
5 CNN-BiLSTM: A combination of the CNN and BiLSTM that uses the results of the former as the input to the latter.
6 BiLSTM-CNN: A combination of the CNN and BiLSTM that uses the results of the latter as the input to the former.
7 LSTM-Attention-CNN: The addition of an attention mechanism to the LSTM-CNN.
8 BiLSTM-Attention-CNN: The addition of an attention mechanism to the BiLSTM-CNN model.

Table 5 shows that the BiLSTM model (accuracy: 0.82, f-measure: 0.8) outperformed the CNN model (accuracy: 0.795, f-measure: 0.775) and the LSTM model (accuracy: 0.81, f-measure: 0.789), indicating that it could better track the contextual information in the text. The CNN model could not capture features across time steps, which is necessary for cyberbullying detection. The LSTM model could save information only on its previous status, and could not preserve information on its subsequent status to enhance the understanding of the context of the sentence.

Similarly, the LSTM-Attention-CNN model was inferior to the BiLSTM-Attention-CNN model by 0.6% in accuracy and 0.9% in the f-measure. Compared with LSTM-Attention-CNN model, the BiLSTM-Attention-CNN model can extract features more sufficiently from both forward and backward directions of each sentence.

The addition of the attention mechanism (BiLSTM-Attention model) enhanced the performance of the BiLSTM model by 1% in terms of accuracy and the f-measure. Similarly, the BiLSTM-CNN model without the attention mechanism was inferior to the BiLSTM-Attention-CNN model by 0.9% in terms of accuracy and the f-measure. We can conclude that the attention mechanism enabled the model to better collect information hidden in the text by concentrating important words and improve performance.

The hybrid BiLSTM-CNN model improved the accuracy of the BiLSTM model from 0.82 to 0.856, and its f-measure from 0.8 to 0.84. The combination of the BiLSTM network and the CNN enabled the model to simultaneously capture both long-distance and local features to improve its accuracy of detection. The CNN-BiLSTM model performed poorly compared with the BiLSTM-CNN model, with values

Table 5
Results of models of cyberbullying detection

models	accuracy	precision	recall	f-measure
CNN	0.795	0.762	0.788	0.775
LSTM	0.81	0.785	0.793	0.789
BiLSTM	0.82	0.798	0.801	0.8
BiLSTM-attention	0.83	0.809	0.812	0.81
CNN-BiLSTM	0.85	0.824	0.845	0.834
BiLSTM-CNN	0.856	0.834	0.847	0.84
LSTM-attention-CNN	0.859	0.838	0.845	0.84
BiLSTM-attention-CNN	0.865	0.848	0.849	0.849

of accuracy and the f-measure that were lower by 0.6%. It performed poorly because the CNN lost some long-distance features of the inputs received from the embedding layer before feeding its result to the BiLSTM model. This loss of information prevented the subsequent BiLSTM layer from capturing global features and contextual information.

The above results show that the BiLSTM-Attention-CNN model delivered the best performance in terms of cyberbullying detection, with an accuracy of 0.865 and an f-measure of 0.849.

2 Comparison of methods of hyperparameter optimization

We assessed the contribution of the ISSA to the performance of the proposed model of cyberbullying detection. We compared four models by optimizing their hyperparameters through different means, and the results are shown in Table 6.

7 BiLSTM-Attention-CNN: The BiLSTM-Attention-CNN model without optimization, shown in (1).

8 SSA-BiLSTM-Attention-CNN: It applies SSA-based optimization to the BiLSTM-Attention-CNN model.

9 PSO-BiLSTM-Attention-CNN: It applies Particle Swarm Optimization (PSO)-based optimization to the BiLSTM-Attention-CNN model.

10 ISSA-BiLSTM-Attention-CNN: It applies ISSA-based optimization to the BiLSTM-Attention-CNN model (proposed model).

Table 6 shows that the SSA-BiLSTM-Attention-CNN model and the BiLSTM-Attention-CNN model delivered comparable results (difference in accuracy and the f-measure of only 0.1%). The PSO-BiLSTM-Attention-CNN model improved the accuracy of the BiLSTM-Attention model from 0.865 to 0.866, and its f-measure from 0.849 to 0.851.This shows that SSA-based optimization and PSO-based optimization did not significantly improve the accuracy of cyberbullying detection by the model. These results were obtained because the standard SSA and PSO have a limited capability for global search, and readily fall into the local optimum.

We optimized the standard SSA algorithm to improve the performance of the ISSA-BiLSTM-Attention-CNN model compared with that of the SSA-BiLSTM-Attention-CNN model on the task of cyberbullying detection, with an improvement 4.4% in accuracy and 4.9% in the f-measure. Similarly, the ISSA-BiLSTM-Attention-CNN model (accuracy: 0.91, f-measure: 0.899) outperformed the PSO-BiLSTM-Attention-CNN model (accuracy: 0.866, f-measure: 0.851).The performance of the model benefited from the use of the ISSA, which has a powerful ability to escape from the local optimum based on the upper and lower boundaries as well as the mechanism of dual-sample learning. It could thus globally search for the optimal hyperparameters. The process of learning of the ISSA over numerous iterations enabled the model to automatically establish the structure of the network and the appropriate configuration of the hyperparameters, thus avoiding overfitting while enhancing its adaptability and capability of identifying cyberbullying.

3 Comparison of different methods of word representation

We tested the effectiveness of the vector representation of CWE on the models listed below. Table 7 lists the results of the comparison.

9 ISSA-CWE-BiLSTM-Attention-CNN: It applies CWE representation, and is shown in (2) (proposed model).

Table 6

Results obtained by using different methods of hyperparameter optimization

methods	accuracy	precision	recall	f-measure
BiLSTM-attention-CNN	0.865	0.848	0.849	0.849
SSA-BiLSTM-attention-CNN	0.866	0.851	0.849	0.85
PSO-BiLSTM-attention-CNN	0.866	0.852	0.851	0.851
ISSA-BiLSTM-attention-CNN	0.91	0.896	0.903	0.899

Table 7
Results obtained by using different methods to represent text vectors

Methods	accuracy	precision	recall	f-measure
ISSA-CWE-BiLSTM-attention-CNN	0.91	0.896	0.903	0.899
ISSA-SG-BiLSTM-attention-CNN	0.872	0.854	0.861	0.857
ISSA-CBOW-BiLSTM-attention-CNN	0.857	0.836	0.847	0.841
ISSA- Chars- BiLSTM-attention-CNN	0.806	0.786	0.779	0.782

10 ISSA-SG-BiLSTM-Attention-CNN: It uses skip-gram word embedding instead of CWE.

11 ISSA-CBOW-BiLSTM-Attention-CNN: It uses CBOW word embedding instead of CWE.

12 ISSA- Chars- BiLSTM-Attention-CNN: It uses character embedding instead of CWE.

Table 7 shows that the ISSA-CBOW-BiLSTM-Attention-CNN model (accuracy: 0.857, f-measure: 0.841) was slightly inferior to the ISSA-SG-BiLSTM-Attention-CNN model (accuracy: 0.872, f-measure: 0.857). This is because the frequency of words representing cyberbullying behavior in the dataset was lower than that of non-cyberbullying words. In this case, skip gram was able to learn the data in greater detail, and thus was able to better describe the semantics of the words.

The ISSA-Chars-BiLSTM-Attention-CNN model performed poorly compared with the ISSA-CBOW-BiLSTM-Attention-CNN model, with values of accuracy and the f-measure that were lower by 5.1% and 5.9%, respectively. This shows that semantic relations can be learned more effectively from word embeddings by treating each word as a unit, rather than using independent characters as the basic unit.

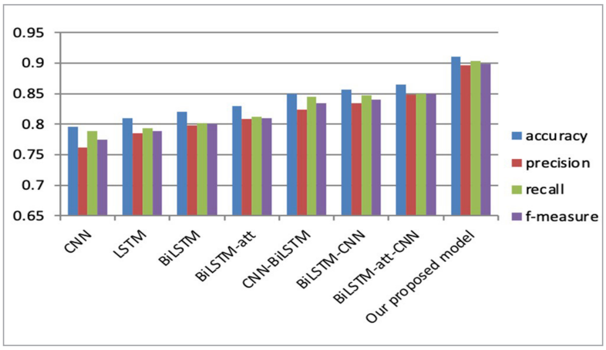
Our proposed ISSA-CWE-BiLSTM-Attention-CNN model delivered the best performance because CWE incorporated information from both words and characters, which were distinct but complementary. The word embedding focused on capturing the contextual semantic relations between words, while the character embedding extracted subtle and local features within the characters that comprised the words.

4 Comparison with state-of-the-art methods

Figure 7 shows the improvement in performance obtained by our proposed model in comparison with the baseline method presented in Equation (1).

We can conclude that the proposed ISSA-BiLSTM-Attention-CNN model achieved the best performance in terms of cyberbullying detection, with an accuracy of 0.91 and an f-measure of 0.899, and outperformed state-of-the-art methods.

Figure 7
Comparison of different models



The above experiments reveal that the improved performance of the ISSA-BiLSTM-Attention-CNN model was obtained due to four components: CWE, the attention mechanism, the hybrid neural network, and the ISSA. All of them improved its performance on the task of cyberbullying detection. A comprehensive semantic representation of the textual content was obtained by CWE. The attention mechanism was able to deduce keywords from the context vectors and allocate larger weights to them. The hybrid neural network compensated for the shortcomings of a single deep learning model. The ISSA was able to escape the local optimum and set the optimal hyperparameters for the deep learning model in the global space to enhance its performance.

4.5. Error Analysis

We now analyze two types of errors incurred by the ISSA-BiLSTM-Attention-CNN model: FN and FP.

The false negatives were obtained for the following reasons:

1 Errors caused by incorrect segmentation of Chinese words

The sentence 百度都搜不到你, 搜狗可以 (“Search for dogs can find you even though Baidu can’t”) actually represents cyberbullying. Our model segmented it as “百度/都/搜不到/你, 搜狗/可以 (“Even Sogou can find you if Baidu can’t”) and identified it as a benign sentence.

2 Errors caused by homophonic

The sentence 大熊猫点外卖—笋到家了 (“The take-out bamboo shoots ordered by the panda have arrived home”) is an example of cyberbullying, where 笋到家了 (“The bamboo shoots have arrived home”) is a homophonic representation of 损到家了 (“bitterly sarcastic extremely”). Our model identified this sentence as benign.

3 Errors caused by ambiguity

The sentence 你厨艺很好啊, 我看你挺会添油加醋的 (“You are an excellent cook. I think you’re quite good at adding oil and vinegar”) was identified by our model as a benign sentence. It actually accuses the interlocutor of being exceptionally skilled at creating, portraying, and inflating falsehoods, and is an example of cyberbullying.

The false positives were obtained for the following reasons:

4 Errors caused by words indicative of bullying

The sentence 网络喷子太多了, 心态扭曲, 看不得人家好, 落井下石 (“There are too many Internet trolls with a distorted mentality, and they can’t see the good intentions of others”) uses the word “trolls” to refer to cyberbullies. Our ISSA-BiLSTM-Attention-CNN model identified it as reflective of cyberbullying, whereas this sentence is a non-evaluative statement that is benign in the given context.

5 Errors caused by network buzzwords

The sentence 板猪,送你个大拇指 (“Pig，I will thumbs up offer you”) was recognized by our model as reflective of cyberbullying owing to the suggestion of an insult. In fact, 板猪 (“Pig”) is a network buzzword with the same meaning as 版主 (“moderator”). Therefore, this sentence should have been identified as benign.

5. Conclusions

In this study, the authors proposed the ISSA-BiLSTM-Attention-CNN model to identify cyberbullying on social media. The primary conclusions are as follows:

1 The hybrid BiLSTM-CNN deep neural network architecture consists of the BiLSTM network for capturing the contextual characteristics of the text, and the CNN for obtaining the salient features of text reflecting cyberbullying.

2 The attention mechanism assigns large computational weights to words that reflect cyberbullying.

3 The modified sparrow search algorithm was used to automatically optimize the configuration of the hyperparameters.

4 The ISSA-BiLSTM-Attention-CNN model outperformed baseline models on a manually annotated corpus of Chinese text containing comments indicative of cyberbullying.

The proposed model still has certain limitations, which we expect to address in future work in the area in the following ways:

1 We plan to improve the vector representation of the model by optimizing CWE. This is accomplished by estimating the semantic contributions of Chinese characters to the corresponding words, rather than simply assigning the same weight to the word and character vectors.

2 We intend to improve the accuracy of detection of the model by incorporating multi-modal information into it, including images, video, and audio, as well as the features of social media and sentiments analysis.

3 We will investigate the use of more powerful deep learning models to detect cyberbullying in Chinese text.

References

1. Alhloul A., Alam A. Bullying Tweets Detection Using CNN-Attention. International Journal on Cybernetics & Information (IJCI), 2023, 12(1), 65-78. https://doi.org/10.5121/ijci.2023.120106

2. Areej, A. H., Hmood, A. D. Detection of Hate Speech in Arabic Tweets Using Deep Learning. Multimedia Systems, 2022, (28), 1963-1974. https://doi.org/10.1007/s00530-020-00742-w

3. Aszemi N. M, Dominic P. D. Hyperparameter Optimization in Convolutional Neural Network Using Genetic Algorithms. International Journal of Advanced Computer Science and Applications, 2019, 10(6), 268-278. https://doi.org/10.14569/IJACSA.2019.0100638

4. Badawy M., Almars A. M., Balaha H. M., Shehata M., Qaraad M., Elhosseini M. A Two-Stage Renal Disease Classification Based on Transfer Learning with Hyperparameters Optimization. Frontiers in Medicine, 2023, (10). https://doi.org/10.3389/fmed.2023.1106717

5. Baghdadi N. A., Malki A., Abdelaliem S. F., Balaha H. M., Badawy M., Elhosseini M. An Automated Diagnosis and Classification of COVID-19 from Chest CT Images Using a transfer Learning-Based Convolutional Neural Network. Computers in Biology and Medicine, 2022, 144. https://doi.org/10.1016/j.compbiomed.2022.105383

6. Balakrishnan V., Khan S., Arabnia H. R. Improving Cyberbullying Detection Using Twitter Users' Psychological Features and Machine Learning. Computers and Security, 2020, 90, 101710. https://doi.org/10.1016/j.cose.2019.101710

7. Behzadi, M., Harris, I. G., Derakhshan, A. Rapid Cyber-Bullying Detection Method Using Compact BERT Models. In ICSC, 2021, 199-202. https://doi.org/10.1109/ICSC50631.2021.00042

8. Bozygit A., Utku S., Nasibov E. Cyberbullying Detection: Utilizing Social Media Features. Expert System with Applications, 2021, (179), 115001. https://doi.org/10.1016/j.eswa.2021.115001

9. Chen X. X., Xu L., Liu Z. Y., Sun M. S., Luan H. B. Joint Learning of Character and Word Embedding. Proceedings of the 24th International Joint Conference on Artificial Intelligence, 2015, 1236-1242.

10. Cheng L., Guo R.C., Silva Y., Hall D., Liu H., Hierarchical Attention Network for Cyberbullying Detection on the Instagram Social Network. Proceedings of the 2019 SIAM International Conference on Data Mining (SDM). https://doi.org/10.1137/1.9781611975673.27

11. Cheng L., Li J. D., Silva Y. N., Hall D. L. Xbully: Cyberbullying Detection Within a Multi-Model Context. WSDM'19: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019, 339-347. https://doi.org/10.1145/3289600.3291037

12. Cheng, L., Li, J., Silva, Y., Hall, D., Liu, H. Pi-Bully: Personalized Cyberbullying Detection with Peer Influence. In IJCAI-19, 5829-5835.

13. China Youth Daily. The Young Pink-Haired Girl Who Suffered From Cyberbullying Has Passed Away [EB/OL]. http://zqb.cyol.com/html/2023-02/22/nw.D110000zgqnb_20230222_2-05.htm

14. Deng, J. F., Cheng, L. L., Wang, Z. W. Attention-Based BiLSTM Fused CNN with Gating Mechanism Model for Chinese Long Text Classification. Computer Speech & Language, 2021, (68). https://doi.org/10.1016/j.csl.2020.101182

15. Fan, Y. Y., Zhang, Y., Guo, B. S., Luo, X. Y., Peng, Q. J., Jin, Z. L. A Hybrid Sparrow Search Algorithm of the Hyperparameter Optimization in Deep Learning. Mathematics, 2022, 10, 3019. https://doi.org/10.3390/math10163019

16. Ge, S. Y., Lu, C., Liu, H. Improving Cyberbullying Detection with User Interaction. WWW'21: Proceedings of the Web Conference, 2021, 496-506. https://doi.org/10.1145/3442381.3449828

17. Gorgolis, N., Hatzilygeroudis, I., Istenes, Z., Gyenne, L. Z. Hyperparameter Optimization of LSTM Network Models Through Genetic Algorithm. 2019 10th International Conference on Information Intelligence, Systems and Applications (IISA). https://doi.org/10.1109/IISA.2019.8900675

18. Hugo, R., Joao, P. C., Pável, C., Bruno, M., Ricardo, R., Luisa, C. Using Fuzzy Fingerprints for Cyberbullying Detection in Social Networks. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2018), 2018, 1-7.

19. Hugo, R., Nádia, P., Ricardo, R., Paula, Costa, F., Joao Paulo, C., Sofia, O., Luísa, C., Paula, P., AM Veiga, S., Isabel, T. Automatic Cyberbullying Detection: A Systematic Review. Computers in Human Behavior, 2019, 93, 333-345. https://doi.org/10.1016/j.chb.2018.12.021

20. Kumar, A., Sachdeva, N. Multimodal Cyberbullying Detection Using Capsule Network with Dynamic Routing and Deep Convolutional Neural Network. Multimedia Systems, 2022, 28, 2043-2052. https://doi.org/10.1007/s00530-020-00747-5

21. Kumari, K., Singh, J. P. Identification of Cyberbullying on Multi-Modal Social Media Posts Using Genetic Algorithm. Transactions on Emerging Telecommunications Technologies, 2021 32(2), e3907. https://doi.org/10.1002/ett.3907

22. Kumari, K., Singh, J. P. Multi-Modal Cyber-Aggression Detection with Feature Optimization by Firefly Algorithm. Multimedia Systems, 2022, 28, 2043-2052. https://doi.org/10.1007/s00530-021-00785-7

23. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P. Gradient-Based Learning Applied to Document Recognition. Proceeding IEEE, 1998, 86, 2278-2324. https://doi.org/10.1109/5.726791

24. Lee, S., Kim, J., Kang, H., Kang, D. Y., Park, J. Genetic Algorithm Based Deep Learning Neural Network Structure and Hyperparameter Optimization. Applied Sciences, 2021, 11(2). https://doi.org/10.3390/app11020744

25. Li, X. Y., Li, S. B., Zhou, P., Chen, G. L. Forecasting Network Interface Flow Using a Broad Learning System Based on the Sparrow Search Algorithm. Entropy, 2022, 24(4), 478. https://doi.org/10.3390/e24040478

26. Lu, N. J., Wu, G. H., Zhang, Z., Zheng, Y. T., Ren, Y. Z., Choo, K. R. Cyberbullying Detection in Social Media Text Based on Character-Level Convolutional Neural Network With Shortcuts. Concurrency and Computation: Practice and Experience, Special Issue on Applications and Techniques in Cyber Intelligence 2019 (ATCI2019). Special Issue on Emerging intelligent big data analytics for cloud and edge computing (CBD2019), 2020, 32(23). https://doi.org/10.1002/cpe.5627

27. Ma, R. X., Teragawa, S., Fu, Z. J. Text Sentiment Classification Based on Improved BiLSTM-CNN. Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), 2020. https://doi.org/10.1109/IPEC49694.2020.9115147

28. Paul, S., Saha, S., CyberBERT: BERT for Cyberbullying Identification. Multimedia Systems, 2022, 28, 1897-1904. https://doi.org/10.1007/s00530-020-00710-4

29. Raj, M., Singh, S., Solanki, K., Selvanambi, R. An Application to Detect Cyberbullying Using Machine Learning and Deep Learning Techniques. SN Computer Science, 2022, 3, 401. https://doi.org/10.1007/s42979-022-01308-5

30. Sadiq, S., Mehmood, A., Ullah, S., Ahmad, M., Choi, G. S. Aggression Detection Through Deep Neural Model on Twitter. Future Generation Computer Systems, 2021, 114, 120-129. https://doi.org/10.1016/j.future.2020.07.050

31. Salawu, S. H Y., Lumsde, J. Approaches to Automated Detection of Cyberbullying: A Survey. IEEE Transactions on Affective Computing, 2017, 11(1), 3-24. DOI: 10.1109/TAFFC.2017.2761757 https://doi.org/10.1109/TAFFC.2017.2761757

32. Sangwan, S. R., Bhatia, M. P. S. D-BullyRumbler: A Safety Rumble Strip to Resolve Online Denigration Bullying Using a Hybrid Filter-Wrapper Approach. Multimedia System, 2022, 28, 1987-2003. https://doi.org/10.1007/s00530-020-00661-w

33. Singh, P., Chaudhury, S., Panigrahi, B. J. Hybrid MPSO-CNN: Multi-Level Particle Swarm Optimized Hyperparameters of Convolutional Neural Network. Swarm and Evolutionary Computation, 2021, 63, 100863. https://doi.org/10.1016/j.swevo.2021.100863

34. Tabatabaei, S. M., Attari, N., Panahi A., Pakfar, M. A., Sedaee, B. EOR Screening Using Neural Network by Sparrow Search Algorithm. Engineering, 2023, 229. https://doi.org/10.1016/j.geoen.2023.212023

35. Wang, K., Cui, Y. P., Hu, J. W., Zhang, Y., Zhao, W., Feng, L. M. Cyberbullying Detection: Based on the FastText and Word Similarity Schemes. ACM Transactions on Asian and Low-Resource Language Information Processing, 2020, 20(1), 1-15. https://doi.org/10.1145/3398191

36. Wang, Y. L., Zhang, H. X., Zhang, G. W. CPSO-CNN: An Efficient PSO-Based Algorithm for Fine-Tuning Hyper-Parameters of Convolutional Neural Network. Swarm and Evolutionary Computation, 2019, 49, 114-123. https://doi.org/10.1016/j.swevo.2019.06.002

37. Watanabe, H., Bouazizi, M., Ohtsuki, T. Hate Speech on Twitter: A Pragmatic Approach to Collect Hateful and Offensive Expressions and Perform Hate Speech Detection. IEEE Access, 2018, 6, 13825-13825. https://doi.org/10.1109/ACCESS.2018.2806394

38. Wu, J. L., Wen, M., Lu, R. X., Li, B. B., Li, J. G. Toward Efficient and Effective Bullying Detection in Online Social Network. Peer to Peer Networking and Applications, 2020, 13, 1567-1576. https://doi.org/10.1007/s12083-019-00832-1

39. Xue, J., Shen, B. A Novel Swarm Intelligence Optimization Approach: Sparrow Search Algorithm. System Science and Control Engineering, 2020, 8(1), 22-34. https://doi.org/10.1080/21642583.2019.1708830

40. Yang, L., Shami, A. On Hyperparameters Optimization of Machine Learning Algorithms: Theory and Practice. Neurocomputing, 2020, 415, 295-316. https://doi.org/10.1016/j.neucom.2020.07.061

41. Zhang, A. M., Li, B. H., Wan, S., Wang K. Cyberbullying Detection with BiRNN and Attention Mechanism. International Conference on Machine Learning and Intelligent Communications (MLICOM), 2019, 623-635. https://doi.org/10.1007/978-3-030-32388-2_52