applied sciences

MDPI

Article

RB_BG_MHA: A RoBERTa-Based Model with Bi-GRU and Multi-Head Attention for Chinese Offensive Language Detection in Social Media

Meijia Xu and Shuxian Liu *

College of Information Science and Engineering, Xinjiang University, Urumqi 830046, China; mjxu_xju@163.com
* Correspondence: liushuxian@xju.edu.cn

Abstract: Offensive language in social media affects the social experience of individuals and groups and hurts social harmony and moral values. Therefore, in recent years, the problem of offensive language detection has attracted the attention of many researchers. However, the primary research currently focuses on detecting English offensive language, while few studies on the Chinese language exist. In this paper, we propose an innovative approach to detect Chinese offensive language. First, unlike previous approaches, we utilized both RoBERTa's sentence-level and word-level embedding, combining the sentence embedding and word embedding of RoBERTa's model, bidirectional GRU, and multi-head self-attention mechanism. This feature fusion allows the model to consider sentence-level and word-level semantic information at the same time so as to capture the semantic information of Chinese text more comprehensively. Second, by concatenating the output results of multi-head attention with RoBERTa's sentence embedding, we achieved an efficient fusion of local and global information and improved the representation ability of the model. The experiments showed that the proposed model achieved 82.931% accuracy and 82.842% F1-score in Chinese offensive language detection tasks, delivering high performance and broad application potential.

Keywords: NLP; offensive language detection; RoBERTa; Bi-GRU

Citation: Xu, M.; Liu, S. RB_BG_MHA: A RoBERTa-Based Model with Bi-GRU and Multi-Head Attention for Chinese Offensive Language Detection in Social Media. *Appl. Sci.* **2023**, *13*, 11000. https://doi.org/10.3390/app131911000

Academic Editor: Alexandre Carvalho

Received: 27 August 2023
Revised: 27 September 2023
Accepted: 4 October 2023
Published: 6 October 2023

1. Introduction

In recent years, with the widespread popularity of social media platforms such as Twitter, Facebook, and microblogging, the scope and scale of the online discussions people engage in has expanded. Social networks provide people around the world with the opportunity to express and share their thoughts instantly and widely. Unfortunately, a few users abuse the anonymity provided by online social media as an advantage and engage in social behavior that would not be acceptable in the real world, resulting in the constant emergence of offensive language. The reasons for the proliferation of offensive language may be: (1) due to the accessibility and anonymity of the platform [1]. Social media provides a relatively anonymous and virtual platform allowing people to express their opinions without revealing their identities. This anonymity may make some users more prone to adopting offensive language because they do not have to bear the social consequences they might face when communicating face-to-face. (2) Social media has emphasized freedom of speech, allowing people to express their views more freely, even extreme or offensive ones, which raises questions about the balance between free speech and offensive speech. (3) There may be some commercial motives. These reasons provide a fertile environment for the spread of offensive and harmful content. Due to the lack of a well-developed legal system regarding offensive language, the difficulty of setting limits for the evolving cyberspace, the increased need for individuals to express their views and opponents to fight back, and the delay in manual checks by Internet operators, the spread of offensive language online has gained new momentum. It continues to challenge policymakers and the research community [2].

Offensive language often involves racism, sexism, and negative stereotypes about countries, religious entities, individuals, or minorities [1]. This language can easily cause adverse effects, such as spreading hate and harassment. In addition, offensive language encourages inappropriate online behavior, such as personal attacks, cyberbullying [3], and hate speech, which seriously undermines the harmony of the online environment [4]. In order to solve the above problems, many social media platforms check users' comments for offensive language and other violations through manual review mechanisms. Still, compared to the scale of people's online discussions, the manual review can only cover a small part of online interactions and is particularly costly in human resources [5]. Automatic identification of offensive language allows the platform to detect offensive language and remove it faster and more efficiently than manual filtering, which is very time-consuming. Therefore, the task of automatic offensive language detection [6,7] has been widely discussed by researchers.

Currently, there are several methods for detecting offensive language and inappropriate content. These methods include rule-based, machine learning, deep learning, integrated, and pre-training methods in deep learning. The rule-based approach uses predefined rules and patterns to detect offensive language. For example, you can create rules to detect common offensive words, threatening phrases, or abusive speech. While these methods are relatively simple, they often require a lot of manual rule design and updating. Machine learning methods that can grasp diverse connections between text fragments and predict specific outputs for specific inputs using pre-labeled data as training material is widely favored in the scholarly exploration of detecting offensive language. Common machine learning algorithms include support vector machine (SVM), decision tree, random forest, etc. With the continuous improvement of machine learning models and the dramatic improvement in the performance of modern models trained for language tasks, AI-based offensive language detection has become a reality. Deep learning methods have made remarkable progress in the detection of offensive language. These methods use deep neural networks such as recurrent neural networks (RNN), short and long-term memory networks (LSTM), and Transformer to capture complex patterns in text. These models often require large amounts of labeled data to train, but they can achieve high performance in many cases. An integrated approach combines several different models or detectors to improve detection performance. Integrated approaches can include voting methods, stacking, or meta-learning techniques. Pre-trained deep learning models, such as BERT, GPT, and RoBERTa, have recently been used for offensive language detection. These models are pre-trained on large-scale text data and can be fine-tuned to suit specific detection tasks. They often perform excellently because they can understand the context.

Over the past few years, task performance in natural language processing has exploded. On the one hand, thanks to improvements in natural language algorithms, such as Text3D [8] (a 3D convolutional neural network for text classification), text classification research on low-resource language Tigrinya [9], optimization of LSTM model by adding an attention layer based on dropout layer and bidirectional recursion layer [10], and only using questions-answers archive automatically creates chatbots [11]. On the other hand, this is mainly due to the development of the Transformer architecture. BERT, a Transformer-based model developed by Google, is the first deeply bidirectional, unsupervised language representation model by co-adjusting the left and right context in all layers [12]. It has shown overwhelming performance and has been used as a central part of many studies in offensive language detection. RoBERTa is an improved model based on BERT, which further optimizes the structure of BERT by adjusting training hyperparameters and expanding training data sets, thereby improving its performance for natural language processing tasks [13]. Compared to BERT, RoBERTa uses a longer training time and a larger corpus, allowing RoBERTa to understand semantics and context in natural language better. Techniques such as Bidirectional Gated Recurrent Unit and Multi-head attention have also been gradually applied in offensive language detection. Bidirectional GRU is based on recurrent neural network technology. It enables the extraction of long-term dependencies

in the text by considering both forward and reverse information of the input sequence [14], thus capturing text fragments containing offensive category features. Multi-head attention can automatically calculate the weighted sum of different position vectors to realize the attention and memory of different positions in the text sequence and to capture rich information in different aspects of the text. Applying these techniques can supplement and enhance the feature representation and performance of the model from different angles. Integrating RoBERTa with bidirectional GRU and Multi-head attention may have great application prospects for enhancing the text perception ability of the model and the accuracy of offensive language detection.

Unfortunately, most of the work in offensive language detection has been carried out on English datasets, and most existing models focus on detecting offensive language in English. Although other languages such as Arabic, Turkish, Greek, Danish, and Urdu [15] are also increasing in proportion in this field [2], it is worth noting that Chinese, as a language with a large number of global speakers, has little research in the field of offensive language detection. Due to the lack of labeled data sets and reliable detectors, the problem of Chinese offensive language detection has yet to be well studied [16].

Therefore, the RB_BG_MHA model is proposed in this paper to detect offensive Chinese language in social media. First, we use the RoBERTa model to extract semantic information at the sentence and word levels and use the pre-trained language model to learn. Secondly, We leverage the bidirectional GRU model to further process word embedding for a richer context-dependent representation. We then integrate the information of the input sequence through the multi-head self-attention mechanism to extract global relationships and important features. Finally, we concatenate the output of multi-head attention with RoBERTa's sentence embedding to enable the model to take advantage of both local and global information.

The contributions of this work are as follows:

(1) Addressing research gap: This study fills the Chinese language research gap in the field of offensive language detection. Although there has been a lot of research on the detection of English offensive language, the research on Chinese is relatively insufficient, especially considering the complexity of Chinese text;
(2) Multi-level information fusion: Our proposed approach uses a multi-level information fusion strategy, combining RoBERTa's sentence-level and word-level embedding, bidirectional GRU model, and multi-head self-attention mechanism. This strategy enables the model to consider different levels of semantic information more comprehensively and helps to understand better and deal with the complexity of Chinese text;
(3) Fine-grained detection of offensive language: Our research further expands the detection scope of offensive language, paying particular attention to the specific classification of offensive language, including race, gender, and region. This fine-grained exploration provides deeper insights to help us better understand and address the complexity of Chinese offensive language;
(4) We conducted experiments on real data sets, and in various established indicators, the proposed model outperforms the existing baseline model and can effectively detect Chinese offensive language.

In the next section, we review related work on offensive language detection. Section 3 provides an overview of the relevant theory required and details our proposed approach. In Section 4, we present and discuss the experimental results in depth. Finally, Section 5 summarizes the main conclusions of this study and provides prospects for future work.

2. Related Work

2.1. Research Methods of Offensive Language Detection

Offensive language refers to hurtful language, including hate speech, derogatory language, and profanity [2]. Offensive language detection plays a vital role in maintaining the harmony of social platforms and promoting civilized communication. In recent years, many researchers have made efforts and proposed many methods for automatically detecting

offensive language work. These techniques can be classified into two categories: conventional machine learning methodologies and deep learning approaches. Traditional machine learning methods build feature engineering for models, manually extracting meaningful features that can be used to train machine learning models such as naive Bayes, logistic regression, and support vector machines. Deep learning approaches employ multi-layer neural networks to automatically extract valuable features from input raw data.

2.1.1. Machine Learning Methods

Chen et al. [17] proposed the lexical-syntactic feature (LSF) structure to detect offensive content and potentially aggressive social media users. Based on the lexical and syntactic features of sentences, the offending value of sentences is obtained, and the traditional learning method is improved by using style, structure, and context features to predict potential aggressive users in social media. Nevertheless, the LSF framework is exclusively designed for English and disregards all other prominent languages, such as Chinese. Shylaja et al. [18] used Doc2Vec to generate document embedding as a feature of several supervised machine-learning methods to detect aggressive comments. Experimental results found that the method combined with Doc2Vec embedding and SVM classifier achieved the best detection results in a series of models. Bohra et al. [19] first attempted to detect hate speech in social media text with mixed Hindi-English code, proposing a supervised classification method that uses various character levels, word levels, and dictionary-based features to identify hate speech in mixed code text. Akhter et al. [20] developed an offensive language dataset of Urdu to detect the offensive language of Urdu, a resource-poor language. The n-grams technique extracted the features, and the offensive languages in Urdu and Roman Urdu data sets were detected and compared using multiple machine learning classifiers.

2.1.2. Deep Learning Methods

In recent years, many studies have used deep learning to solve the problem of offensive language detection with excellent performance, and some studies have confirmed the superiority of deep learning models over machine learning models. Roy et al. [21] created an automated system for hate speech detection on Twitter, employing deep convolutional neural networks (DCNN) to address the issue of identifying hate speech. The authors use GloVe embedding to represent the vector of the tweet text, capturing the semantics of the tweet with the help of convolution operations. Lu et al. [22] introduced a model for detecting cyberbullying involving a Character-Level Convolutional Neural Network with Shortcuts (Char-CNNS). Characters represent the most basic learning elements, enabling the model to tackle the issue of text misspelling on social media platforms. Learn mixed bullying signals with Shortcuts to stitch together different levels of features. Zhou et al. [12] used popular deep learning methods such as BERT, ELMo, and CNN to detect hate speech. They amalgamated the outcomes from diverse classifiers to enhance the classification performance. The classification results of CNN, ELMo, and BERT were combined to validate the practicality of the fusion technique in identifying hate speech. Djandji et al. [23] combined a pre-trained Arabic model (AraBERT) with multi-task learning to accurately enhance the AraBERT model to detect offensive language on Arabic social platforms. The experimental results show that the proposed multi-task AraBERT model is superior to single-task and multi-label AraBERT.

2.1.3. Other Methods

In addition to offensive language detection methods based on machine learning and deep learning, there are other approaches. For example, Gémes et al. [24] proposed a Bert-based offensive language detection method that performed well regarding F1 scores. In addition, they used the custom framework to build a high-precision rule-based offensive language detection method that can be used either as a standalone high-precision classifier or as a supplement to improve recall rates for Bert-based methods. Pradeep et al. [25]

proposed a Dravidian hate speech and offensive language detection method based on a deep integration framework, which proposed some integration models mainly composed of DNN, BERT, and xlm-RoBERTa models. The distilBERT, DNN, and xlm-RoBERTa integrated models and BERT, DNN, and MuRIL integrated models performed best on two real datasets, respectively. Segun et al. [26] evaluated the intermediate pre-training of the offensive language recognition task. They used the pre-trained language model for the Twitter field to solve the offensive language detection task in Spanish and Mexican Spanish. They also found that further training in multilingual sentiment analysis benefited this task.

2.2. Correlation Data Sets

Due to the lack of a reliable Chinese data set for offensive language detection, Deng et al. [16] proposed a benchmark—COLD, which includes a Chinese data set for offensive language and a baseline detector trained on the data set. To the extent of the authors' awareness, this represents the initial openly accessible Chinese dataset for offensive language, comprising 37,480 comments annotated with binary labels covering diverse topics such as region, race, and gender. Chung et al. [1] proposed a Chinese abusive language dataset, TOCAB, which contained 121,344 comments from social media sites. The authors used several baseline systems of machine learning and deep learning to test this benchmark, and experimental results indicated that the deep learning model exhibited superior performance compared to the machine learning model. Lu et al. [27] proposed TOXICN, a fine-grained dataset containing indirect toxicity samples, which promotes fine-grained Chinese toxic language detection. It also constructs insult words containing implied profanity. It proposes the Toxic Knowledge Enhancement (TKE) benchmark to introduce lexical features to detect toxic language and verifies the effectiveness of TKE through experiments. On the other hand, Zhou et al. [28] used the offensive language detection data of Korean and English, two different cultural backgrounds, to explore the influence of transfer learning on identifying offensive language in the Chinese context. Their findings demonstrate the promise of non-English offensive language detection in resource-limited Settings, emphasizing the significance of cross-cultural transfer learning in enhancing the performance of offensive language detection.

2.3. Literature Review: Summary and Evaluation

Most traditional machine learning methods rely on manually designed features, which can have limitations when dealing with data diversity and a changing network context. However, deep learning can overcome some limitations of traditional methods in offensive language detection, avoid cumbersome manual feature engineering, and improve the adaptability and generalization ability of the model. Other methods, such as rule-based approaches, are suitable for simple tasks and specific contexts but perform poorly in complex and diverse offensive language detection. An integration-based approach can improve performance but requires more computing resources and data. Pretraining-based methods perform well in modern offensive language detection because of their strong representation and adaptability, but they also need more computational resources and data support.

There is a significant research gap in the field of offensive language detection, that is, the focus tends to be on English, while the research on offensive language detection of other languages is relatively insufficient, which leads to an urgent need to focus on other languages. In terms of Chinese offensive language detection, the complexity of Chinese text increases the difficulty of this task. Chinese has a rich grammatical structure, which may require more complex natural language processing techniques to cope with, but also may present researchers with more challenges. Therefore, we chose to focus our research on Chinese, which, despite its large number of speakers, has relatively few studies on offensive detection.

Our proposed approach fills this research gap by employing multi-level information fusion strategies, including RoBERTa's sentence-level and word-level embedding, bidirec-

tional GRU model, and multi-head self-attention mechanism, which enable the model to consider different levels of semantic information at the same time and contribute to a better understanding of the complexity of Chinese text. In addition, by concatenating the output of multi-head attention with RoBERTa's sentence embedding, the model makes use of local and global information simultaneously, improving the model's representation ability and helping to better deal with the complex context in Chinese text.

Another research gap is that most studies still need to address the issue of fine-grained offensive language detection. In our study, we further expanded the scope of detection of offensive language, paying special attention to the specific classification of offensive language, including race, gender, and region. This fine-grained exploration provides deeper insights to help us better understand and address the complexity of Chinese offensive language.

3. Theory and Method

3.1. RoBERTa Layer

RoBERTa aims to learn a common language representation by pre-training large-scale unlabeled text data for fine-tuning and application on various downstream natural language processing tasks. Figure 1 is an architecture diagram of RoBERTa's model.

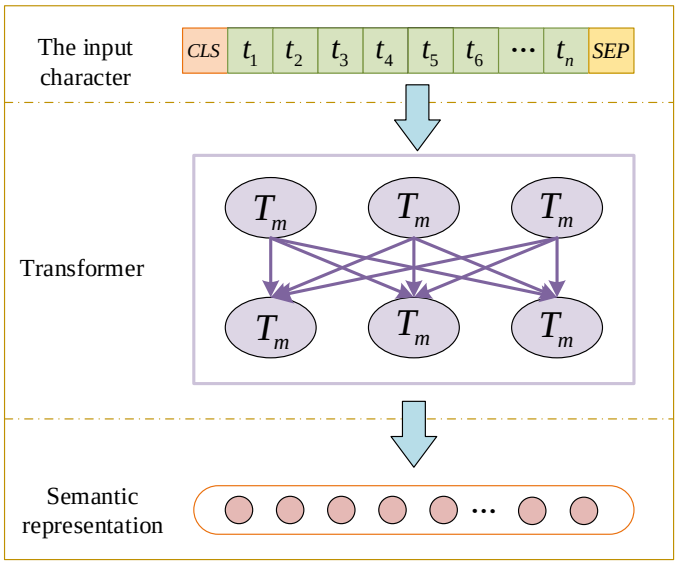

Figure 1. Schematic diagram of the structure of the RoBERTa model.

It improves BERT in the following ways:

(1) Pre-training dataset: RoBERTa conducts pre-training with large amounts of unlabeled text data, using more data than BERT and not limiting the maximum length of each sentence;

(2) Pre-training tasks: RoBERTa employs the same two pre-training tasks as BERT, namely Masked Language Modeling (MLM) and Next Sentence Prediction (NSP).

In the MLM task, the model needs to predict the obscured words based on the context.

For a given input text $T = \{W_1, W_2, \ldots, W_n\}$, RoBERTa first masks some random words and replaces them with the special MASK token "[MASK]" to obtain the mask text $M = \{m_1, m_2, \ldots, m_n\}$.

Then, RoBERTa encodes through the Transformer model to obtain the output $H = \{h_1, h_2, \ldots, h_n\}$ of the encoder layer. For a mask position m_i, RoBERTa maps d_{model} to a dimensional vector $z = W_{out}h_i$ by linear transformation and obtains the final predicted distribution $P(z|T)$ by softmax operation.

In the NSP task, the model needs to determine whether two sentences are continuous.

For a pair of input sentences $S = (s_1, s_2)$, RoBERTa represents them as a special sequence [CLS]s_1[SEP]s_2[SEP]. RoBERTa then passes this composite sequence into the Transformer model for encoding, yielding the output $H = \{h_1, h_2, \ldots, h_n\}$. Here, h_1 represents the output of the composite sequence's first token "[CLS]". RoBERTa then takes the vector h_1 representing the entire sequence as input and runs it through a binary classifier to determine whether the two sentences occur in succession.

(3) Training parameters: RoBERTa has improved some training hyperparameters. It uses larger batch sizes, longer training sequences, and more training steps to enhance the model's performance;
(4) Dynamic mask: RoBERTa introduced the concept of dynamic mask. Each training instance randomly generates a mask during each training step. This can improve the model's generalization ability and reduce the dependence on location information;
(5) Randomization during training: RoBERTa used more randomization methods in the pre-training process, including random removal and dynamic masking, which helped the model better adapt to various natural language processing tasks.

3.2. Bidirectional GRU Layer

A Bidirectional Gated Recurrent Unit is a recurrent neural network (RNN) variant that considers past and future contextual information in time series data. GRU is a gated loop unit that models long-term dependencies in sequence data. In a traditional one-way RNN, information flows in only one direction, from the past to the future. The bidirectional GRU incorporates an additional reverse layer, enabling information propagation in both the forward and backward directions, thus enhancing the capture of contextual information within sequence data.

3.2.1. Forward GRU

The internal cell structure of a forward GRU is shown in Figure 2.

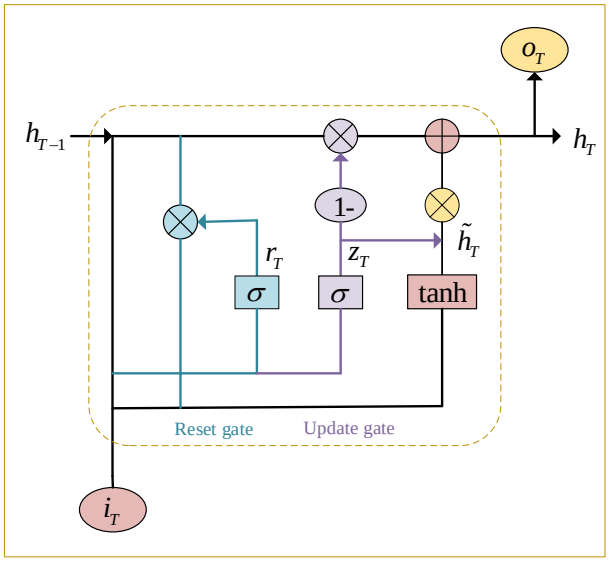

Figure 2. Schematic diagram of the forward GRU model.

The formula for the forward GRU is

$$z_T^f = \sigma\left(W_z^f i_T + U_z^f h_{T-1}^f + b_z^f\right), \tag{1}$$

$$r_T^f = \sigma\left(W_r^f i_T + U_r^f h_{T-1}^f + b_r^f\right), \tag{2}$$

$$\tilde{h}_T^f = tanh\left(W_h^f i_T + U_h^f\left(r_T^f \bigodot h_{T-1}^f\right) + b_h^f\right), \tag{3}$$

$$h_T^f = \left(1 - z_T^f\right) \bigodot h_{T-1}^f + z_T^f \bigodot \tilde{h}_T^f, \tag{4}$$

where z_T^f represents the update gate, r_T^f represents the reset gate, and $\odot$ represents the multiplication of element levels, h_{T-1}^f is the hidden state of the previous time step, and h_T^f is the hidden state of the current time step. W_z^f, W_r^f, W_h^f are the weight matrices of the inputs to z_T^f, r_T^f, $\tilde{h}_T^f$, respectively. U_z^f is the weight matrix from h_{T-1}^f to z_T^f, U_r^f is the weight matrix from h_{T-1}^f to r_T^f, and U_h^f is the weight matrix dealing with r_T^f and h_{T-1}^f.

3.2.2. Backward GRU

The formula for the backward GRU is similar to that for the forward GRU, except that the forward input sequence is arranged in reverse order in the time dimension to obtain the reverse input sequence, and then the operation is performed on it. The internal cell structure of a backward GRU is shown in Figure 3.

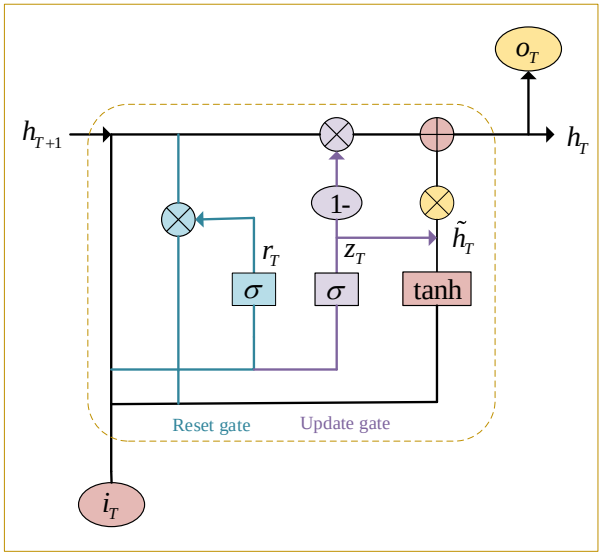

Figure 3. Schematic diagram of backward GRU model.

The formula for the backward GRU is

$$z_T^b = \sigma\left(W_z^b i_T + U_z^b h_{T+1}^b + b_z^b\right), \tag{5}$$

$$r_T^b = \sigma\left(W_r^b i_T + U_r^b h_{T+1}^b + b_r^b\right), \tag{6}$$

$$\tilde{h}_T^b = tanh\left(W_h^b i_T + U_h^b\left(r_T^b \bigodot h_{T+1}^b\right) + b_h^b\right), \tag{7}$$

$$h_T^b = \left(1 - z_T^b\right) \bigodot h_{T+1}^b + z_T^b \bigodot \tilde{h}_T^b, \tag{8}$$

where z_T^b represents the backward update gate, r_T^b represents the backward reset gate, and $\odot$ represents the multiplication of element levels, h_T^b is the hidden state of the current time step, h_{T+1}^b is the hidden state of the latter time step. W_z^b, W_r^b, W_h^b are the weight matrices

of the inputs to z_T^b, r_T^b, $\tilde{h}{}_T^b$, respectively. U_z^b is the weight matrix from h_{T+1}^b to z_T^b, U_r^b is the weight matrix from h_{T+1}^b to r_T^b, and U_h^b is the weight matrix dealing with r_T^b and h_{T+1}^b.

3.2.3. Bidirectional GRU Output

After going through the forward and backward GRUs, we spliced their outputs to obtain the output of the bidirectional GRU:

$$h_T = Concat\left(h_T^f, h_T^b\right), \tag{9}$$

where *Concat* represents the concatenation of two vectors according to dimension.

3.3. Multi-Head Attention Layer

Multi-head attention is an extended attention mechanism based on the self-attention mechanism to deal with information interaction and representation learning in sequence data. It is a crucial component of the Transformer model and is widely used in natural language processing tasks. Here are the critical steps for Multi-head attention:

3.3.1. Linear Mapping

Given an input sequence $X = \mathbb{R}^{n \times d_{model}}$, we transform it into a sequence of queries, keys, and values via linear mapping:

$$Q_i = W_i^Q X \in \mathbb{R}^{n \times d_k}, \tag{10}$$

$$K_i = W_i^K X \in \mathbb{R}^{n \times d_k}, \tag{11}$$

$$V_i = W_i^V X \in \mathbb{R}^{n \times d_v}, \tag{12}$$

where $i \in [1, h]$, d_k, d_v are dimensions that can be specified by hyperparameters, and W_i^Q, W_i^K, W_i^V are learnable weight matrices with shapes $\mathbb{R}^{d_{model} \times d_k}$, $\mathbb{R}^{d_{model} \times d_k}$, and $\mathbb{R}^{d_{model} \times d_v}$.

3.3.2. Self-Attention Matrix

For the query, key, and value of group i, we calculated the similarity through the dot product and then performed softmax to obtain the corresponding self-attention matrix $A_i \in \mathbb{R}^{n \times n}$:

$$A_i = softmax\left(\frac{Q_i (K_i)^T}{\sqrt{d_k}}\right), \tag{13}$$

The dot product attention mechanism defines the similarity calculation of query and key as

$$Attention = (Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{14}$$

The result of matrix multiplication QK^T is a matrix of size (n,n), where each element (i,j) represents a similar fraction of the ith element in the query sequence and the jth element in the key sequence.

3.3.3. Final Output

By concatenating the self-attention matrix A_i and the corresponding value V_i, we can obtain

$$\widetilde{V} = \mathbb{R}^{n \times hd_v}, \tag{15}$$

$$\widetilde{V} = Concat(A_1 V_1, A_2 V_2, \ldots, A_h V_h), \tag{16}$$

where *Concat* indicates splicing by dimension.

Finally, the concatenated $\widetilde{V}$ is converted to the corresponding output by linear mapping $W_o \in \mathbb{R}^{hd_v \times d_{model}}$:

$$Out = \widetilde{V} W_o, \tag{17}$$

3.4. Proposed Model

The proposed model gradually extracts the semantic information of the input sequence through representation learning at multiple levels, as shown in Figure 4. First, the RoBERTa model encoded each sentence to generate sentence embedding, while also generating word embedding for each word. These embedding vectors contain the semantic information of the text. We then passed the word embedding extracted by the RoBERTa model to the bidirectional GRU model, which processes the word embedding for each word, considering the contextual information of the word in the sequence. We then used a multi-head self-attention mechanism to integrate the output of the bidirectional GRU, which helps identify essential features in the text. Finally, the output results of the multi-head attention mechanism were concatenated with the sentence embedding extracted by the RoBERTa model, which combines the global relational and context-dependent representation with the semantic information at the original sentence level to obtain a richer semantic representation.

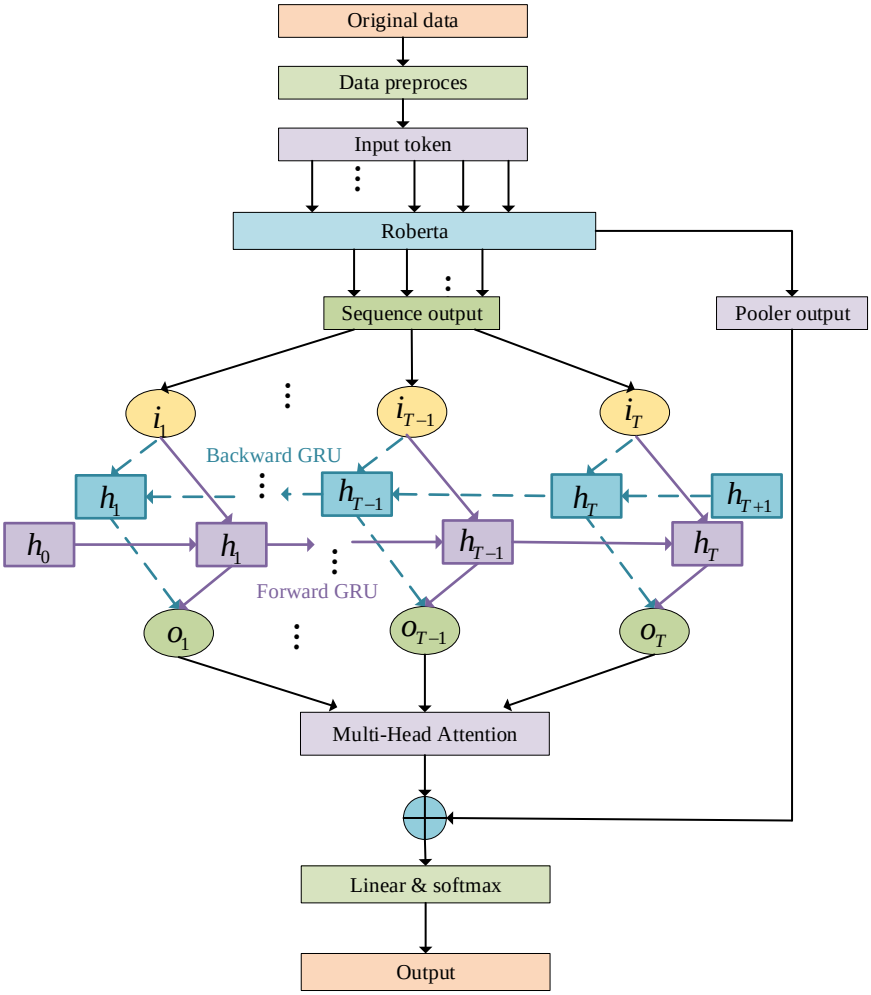

Figure 4. Schematic diagram of the structure of the proposed RB_BG_MHA model. (The solid arrow represents the forward GRU and the dashed arrow represents the backward GRU.)

The model can take advantage of local and global information by concatenating the output of multi-head attention with RoBERTa's sentence embedding. This kind of model integration can provide more abundant feature representation, thus enhancing the expressiveness of the model.

The process is as follows:

Given an input sequence $X \in \mathbb{R}^{n \times d_x}$, where d_x represents the dimensional size of the input sequence.

First, we used the RoBERTa model to extract the semantic representation of the input sequence to obtain the sentence embedding $H_R \in \mathbb{R}^{d_r}$, where d_r Represents the sentence embedding dimension of the RoBERTa model. At the same time, the word embedding matrix $H_W \in \mathbb{R}^{n \times d_w}$ is obtained through the RoBERTa model, where each line represents the embedding vector of a word, representing the semantic information of each word in the input sequence, and d_w is the dimension of word embedding.

Second, we use the bidirectional GRU model to further process the word embedding, yielding the output $H_G \in \mathbb{R}^{n \times d_g}$ of the GRU model, where d_g represents the output dimension of the GRU model.

Then, we used the multi-head self-attention mechanism to further integrate the information from the input sequence and obtain the final representation $H_A \in \mathbb{R}^{d_a}$, where d_a represents the output dimension of the attention mechanism.

Specifically, we first took the output H_G of the GRU model as the input query, key, and value in the self-attention mechanism, respectively, and then used multiple attention heads to compute in parallel, concatenate the output of each head to obtain the feature representation of the multi-head attention computation, and, after that, we applied a linear layer to transform the output of the multi-head attention to obtain the output result $H_A \in \mathbb{R}^{d_a}$. Finally, H_A and H_R were concatenated, and then the concatenated result was fed into the fully connected layer for classification to obtain the final result.

To sum up, the formula of the entire model is expressed as follows:

$$H_R = RoBERTa(X) \in \mathbb{R}^{d_r}, \tag{18}$$

$$H_W = RoBERTa(X) \in \mathbb{R}^{n \times d_w}, \tag{19}$$

$$H_G = BiGRU(H_W) \in \mathbb{R}^{n \times d_g}, \tag{20}$$

$$H_A = MultiHeadAttention(H_G) \in \mathbb{R}^{n \times d_a}, \tag{21}$$

$$H_{concat} = [H_R; H_A], \tag{22}$$

$$O = Dense(H_{concat}), \tag{23}$$

where $RoBERTa(\cdot)$ represents the RoBERTa model, $BiGRU(\cdot)$ represents the bidirectional GRU model, $MultiHeadAttention(\cdot)$ represents the multi-head self-attention mechanism, d_r, d_g, and d_a represents the output dimensions of the RoBERTa model, bidirectional GRU model, and multi-head self-attention mechanism, respectively.

4. Experimental Research

4.1. Data Set Preparation

The experiment in this paper was conducted on the COLD Chinese offensive language dataset collected by Deng et al. [16], which was created using authentic data published on Chinese social media platforms (Zhihu and Weibo) and contained 37,480 comments with binary offensive labels covering various topics of race, gender, and region. The details are shown in Table 1. The partitioning of the data set is shown in Table 2.

Table 1. Statistics of COLD dataset under each topic. List the number of offensive language samples and non-offensive language samples corresponding to the race, region, and gender topics.

Topic	Race	Region	Gender	Total
Offen	7683	5550	4808	18,041
Non-Offen	7370	7090	4979	19,439
Total	15,053	12,640	9787	37,480

Table 2. Distribution statistics of samples from different classes in the COLD dataset.

	Offensive	Non-Offensive	Total
Train/Dev	15,934	16,223	32,157
Test	2107	3216	5323
Total	18,041	19,439	37,480

4.2. Experimental Environment and Parameter Setting

This experiment was run on a 3090-24G GPU, PyTorch 1.13.1, Cuda 11.7.0, and the Python3.8 environment. Using AdamW as the optimizer, the learning rate was set to0.00001, the Weight Decay coefficient was set to0.00001, the Hidden Size was set to 768, the number of heads of the attention mechanism was set to 4, and the batch size was set to 64. The training process introduces the dropout technique, and the dropout rate was set to 0.1. This technique can randomly discard nodes in a particular proportion, thus effectively preventing overfitting.

4.3. Evaluation Index

The model effect was evaluated by accuracy, precision, recall, and F1-score.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{24}$$

$$Precision = \frac{TP}{TP + FP}, \tag{25}$$

$$Recall = \frac{TP}{TP + FN}, \tag{26}$$

$$F1\text{–}Score = 2 * \frac{Precision \times Recall}{Precision + Recall}, \tag{27}$$

where *TP*: true positive (positive category predicts correct outcome), *TN*: true negative (negative categories predict correct negative categories predict correct results), *FP*: false positive (positive categories predict incorrect outcome), *FN*: false negative (negative categories predict incorrect outcome).

4.4. Baseline Method

Baidu Text Censor: It is a text content review service provided by Baidu, designed to help enterprises and developers achieve automatic review and filtering of text content. The API can identify and filter harmful content, such as abuse, pornography, violence, terrorism, etc., to protect users from wrong information.

COLDET [16]: COLDET(COLDETECTOR) uses an architecture based on the transformer and pre-trained model BERT. The backbone of the model is the bert-base-chines model with 12 layers and 12 attention heads.

RoBERTa + TKE [27]: The model applies the TKE (Toxic Knowledge Enhancement) method to the RoBERTa model by means of an insult lexicon containing explicit profanity and implicit profanity, combined with lexical features to detect toxic language.

XLM-Rlarge [29]: A cross-language pre-trained series of models based on Transformer architecture developed by the Facebook AI Research team XLM-R (Cross-lingual Language Model for Robust Pre-Training). XLM-Rlarge is the larger version of the XLM-R family, with a deeper model structure and more parameters than XLM-Rbase.

4.5. Experimental Results and Analysis

4.5.1. Contrast Experiment

To verify the RB_BG_MHA model, a series of experiments were conducted and compared with other models. Simultaneously, the experimental results were analyzed in detail to showcase the efficacy of our model in identifying offensive language in the Chinese context.

According to the comparative experimental results in Table 3, the accuracy of the RB_BG_MHA model on the test set was 82.931%, the precision was 82.257%, the recall was 83.436%, and the F1-score was 82.842%. Compared with other models, our proposed model shows better results in accuracy, precision, recall, and F1-score. The RoBERTa + TKE model also showed good performance, while the XLM-Rlarge model achieved high results in accuracy but a relatively low F1-score. However, the Baidu Text Censor model performs poorly on the whole, with low accuracy, precision, and recall, the latter displaying a rate of only 27.005%. Compared with Baidu Text Censor, the method based on deep learning has achieved better performance. A plausible rationale is that the online API's filtering mechanism predominantly depends on keyword dictionaries. As a result, it cannot effectively detect sentences that contain indirect offensive language. In summary, compared with other models, our proposed model can provide more abundant feature representation and can understand and express the semantics of sentences more accurately, thus achieving better results.

Table 3. Comparison of offensive language detection performance across different models on our test set.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Baidu Text Censor [1]	64.738	62.665	27.005	37.745
COLDET [16]	81	80	82	81
RoBERTa [13]	81.058	80.563	81.886	81.219
XLM-Rlarge [27]	81.87	/	/	79.09
RoBERTa + TKE [29]	81.908	82.233	81.908	81.740
RB_BG_MHA (ours)	82.931	82.257	83.436	82.842

[1] https://ai.baidu.com/tech/textcensoring (accessed on 26 August 2023).

Although the current performance is only slightly optimized compared to the RoBERTa + TKE method, we will consider carrying out the following the future to improve the performance of our method:

Introducing lexical features: For offensive language detection tasks, it is relatively easy to detect direct offensive samples, but it is challenging to detect indirect offensive instances (such as stereotypes and sarcasm). Therefore, our future work will construct a dictionary containing implied insult and offensive words to introduce indirect offensive word features and obtain a weighted enhanced representation of each word so that the obtained sentence embedding will include implied offensive word features to improve the performance of offensive language detection;

Data enhancement and diversity: We plan to collect more data on the Chinese offensive language and ensure the variety of the data set, which will help the model to better generalize to situations of various offensive languages.

From Table 4 we can notice that the RB_BG_MHA model has a higher number of parameters and a longer running time with respect to the RoBERTa and RoBERTa + TKE models. This reflects the higher complexity of our model, which may lead to a waste of computational resources in some cases. However, we recognize this drawback and will further

optimize and reduce our model complexity to reduce the running time without harming performance to meet the dual requirements of performance and computational efficiency.

Table 4. Experimental comparison of the number of parameters and running time in RoBERTa-based models.

Model	Params	Runtime(s)
RoBERTa	102,269,186	590.35
RoBERTa + TKE	102,864,386	615.98
RB_BG_MHA	131,995,747	863.23

4.5.2. Fine-Grained Experiment

Previous studies failed to detect specific types of offensive content in detail, so we conducted further fine-grained detection of Chinese offensive language, focusing on specific categories of offensive language, including race, gender, region, etc. We also performed offensive language detection for each of the three types of social text. Through this research, we aim to more accurately identify and understand offensive language and provide more effective content filtering and monitoring mechanisms for social media.

Figure 5 shows the results of the specific classification of offensive language by race, gender, and region. Among these performance indicators, the highest was recall (81.3%), and the lowest is precision (80.3%), while accuracy and F1-score were around 80.8%. The experimental results of offensive language detection for different text types showed that the gender category performed best in offensive language detection, despite the small number of data. This phenomenon may be attributed to the inclusion of specific words (such as "male" and "female") in the offensive language of the gender category, which may be more easily captured by the model, thus improving the accuracy of the detection. In contrast, the offensive language in the race category presents a more complex text structure, which makes it difficult for the model to accurately capture hidden features, resulting in lower detection performance.

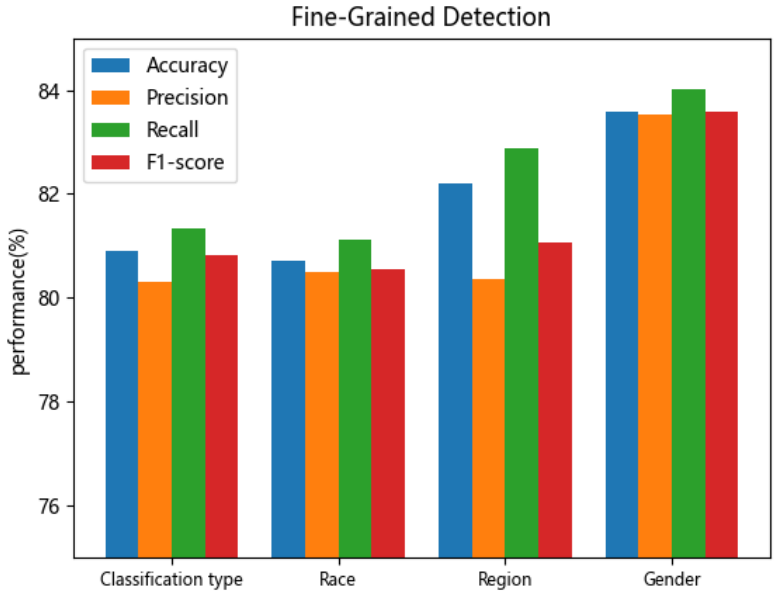

Figure 5. Offensive language detection: category prediction and analysis of race, gender, and region topics.

This phenomenon can also be explained by the fact that in the gender category, the explicit gender-related words lead to clearer feature performance, which is conducive to effective classification. In the race category, the offending language may be more subtle and varied, with greater changes in semantics and context, which increases the complexity of

model identification. Therefore, differences in data characteristics and language complexity may be one of the key factors leading to performance differences.

4.6. Ablation Analysis

In order to verify the role of each component of the model and the integrity of the model, this paper strips each component from the model one by one and observes the degree of its impact on the model performance. If the model performance degrades after removing a component, it indicates that the component is indispensable and of independent importance. Otherwise, it is dispensable. If the model performance suffers when each component is removed, then the model is unified, and each component is indispensable.

We conducted the following ablation experiments to verify the integrity of the RB_BG_MHA model. We removed each component one by one and observed changes in model performance. If the model's performance decreased after removing a component, it indicated that the component plays an essential role in the model's performance.

Table 5 shows the ablation experiment results, which are visually represented as shown in Figure 6. The model was decomposed into three modules: sentence embedding, word embedding + BiGRU, and multi-head self-attention, and one module was removed one by one to evaluate its impact on the model performance.

Table 5. The results of the ablation study (RB: RoBERTa, MLP: Multi-Layer Perceptron, BG: Bidirectional Gated Recurrent Unit, MHA: Multi-Head Attention).

Model	Embedded Mode	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
RB	Sentence embedding	81.058	80.563	81.886	81.219
RB_MLP	Sentence embedding	81.670	81.183	82.530	81.851
RB_BG	Word embedding	81.561	81.080	82.428	81.748
	Word embedding + Sentence embedding	81.037	80.600	81.961	81.275
RB_BG_MHA	Word embedding	81.863	81.237	82.481	81.854
	Word embedding + Sentence embedding	82.931	82.257	83.436	82.842

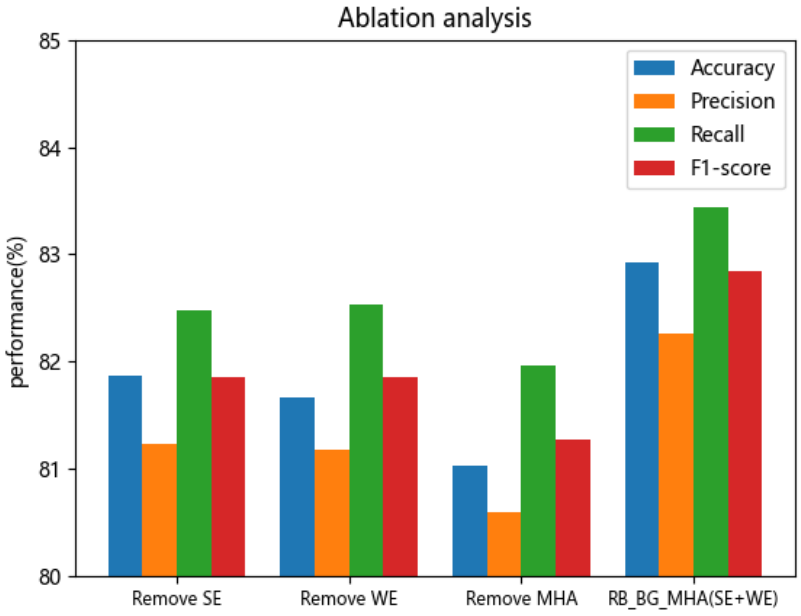

Figure 6. Visual representation of ablation analysis (SE: Sentence Embedding, WE: Word Embedding, MHA: Multi-Head Attention).

When the sentence embedding module was removed, the RB_BG_MHA model using only word embedding was formed, and the model's accuracy decreased by about 1.1%. The precision, recall, and F1-score all decreased by about 1.0%. When the multi-head self-attention module was removed, the RB_BG model using both word embedding and sentence embedding was formed, and the accuracy of the model was reduced by about 1.9%, the precision by about 1.7%, and the recall by about 1.5%. The F1-score fell by about 1.6%. When the word embedding module was removed, the RB_MLP model using only sentence embedding was formed, and the accuracy of the model decreased by about 1.3%, the precision decreased by about 1.1%, the recall decreased by about 0.9%, and the F1-score decreased by about 1.0%.

Through the above experimental results, this paper can fully reveal the role of each component of the model and the integrity of the model, and it can be clearly observed that the performance of the RB-BG-MHA model depends on the joint action of each module, which is indispensable.

5. Conclusions

Although English offensive language detection has been extensively studied, Chinese offensive language detection needs to be more studied, especially considering the complexity of Chinese text. Our study fills the Chinese research gap in offensive language detection. Our approach adopts multi-level information fusion strategies, including RoBERTa's sentence-level and word-level embedding, a bidirectional GRU model, and a multi-head self-attention mechanism, which enables the model to consider different levels of semantic information simultaneously. It helps to understand the complexity of Chinese text better. In addition, by concatenating the output results of multi-head attention with RoBERTa's sentence embedding, the efficient fusion of local and global information was realized, and the representation ability of the model was improved, which helps to better deal with the complex context in Chinese text. Our study further expands the scope of detection of offensive language, focusing on specific categories of offensive language, including race, gender, and region. This fine-grained exploration provides deeper insights to help us better understand and address the complexity of Chinese offensive language.

The experimental results show that our proposed method effectively detects Chinese offensive language. In addition, the ablation experiment also confirmed that the performance of our model depends on the synergistic effect of each module, and each module is indispensable. We also developed a series of strategies to continue to improve. First, we plan to build a dictionary of implicit insult and offensive words to introduce implicit offensive word features, thereby improving the performance of the model. Secondly, we will actively collect more Chinese offensive language data to ensure better generalization performance of the model. In addition, our future work will focus on promoting research in the field of mixed-language offensive language detection to adapt to diverse language environments and application scenarios. Finally, we plan to delve deeper into the joint analysis of multimodal information to further improve the performance and applicability of our method.

Author Contributions: Methodology, M.X.; data curation, M.X.; writing—original draft preparation, M.X.; writing—review and editing, M.X. and S.L.; supervision, S.L.; project administration, S.L.; funding acquisition, S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Grant No.: 61762085) and the Natural Science Foundation of Xinjiang Uygur Autonomous Region from Xinjiang, China (Grant No.: 2019D01C081).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Dataset derived from https://github.com/thu-coai/COLDataset (accessed on 26 August 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chung, I.; Lin, C.J. TOCAB: A Dataset for Chinese Abusive Language Processing. In Proceedings of the 2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI), IEEE, Las Vegas, NV, USA, 10–12 August 2021; pp. 445–452.
2. Jahan, M.S.; Oussalah, M. A systematic review of Hate Speech automatic detection using Natural Language Processing. *Neurocomputing* **2023**, *9*, 126232. [CrossRef]
3. López-Vizcaíno, M.; Nóvoa, F.J.; Artieres, T.; Cacheda, F. Site Agnostic Approach to Early Detection of Cyberbullying on Social Media Networks. *Sensors* **2023**, *23*, 4788. [CrossRef] [PubMed]
4. Wulczyn, E.; Thain, N.; Dixon, L. Ex machina: Personal attacks seen at scale. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 1391–1399.
5. Zhao, Y.; Tao, X. ZYJ123@ DravidianLangTech-EACL2021: Offensive Language Identification Based on XLM-RoBERTa with DPCNN. In Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages, Kiev, Ukraine, 19–23 April 2021; pp. 216–221.
6. Kar, P.; Debbarma, S. Multilingual hate speech detection sentimental analysis on social media platforms using optimal feature extraction and hybrid diagonal gated recurrent neural network. *J. Supercomput.* **2023**, *79*, 19515–19546. [CrossRef]
7. Pereira-Kohatsu, J.C.; Quijano-Sánchez, L.; Liberatore, F.; Camacho-Collados, M. Detecting and monitoring hate speech in Twitter. *Sensors* **2019**, *19*, 4654. [CrossRef] [PubMed]
8. Wang, J.; Li, J.; Zhang, Y. Text3D: 3D Convolutional Neural Networks for Text Classification. *Electronics* **2023**, *12*, 3087. [CrossRef]
9. Fesseha, A.; Xiong, S.; Emiru, E.D.; Diallo, M.; Dahou, A. Text classification based on convolutional neural networks and word embedding for low-resource languages: Tigrinya. *Information* **2021**, *12*, 52. [CrossRef]
10. Wang, Z.; Kim, S.; Joe, I. An Improved LSTM-Based Failure Classification Model for Financial Companies Using Natural Language Processing. *Appl. Sci.* **2023**, *13*, 7884. [CrossRef]
11. Massaro, A.; Maritati, V.; Galiano, A. Automated self-learning chatbot initially build as a FAQs database information retrieval system: Multi-level and intelligent universal virtual front-office implementing neural network. *Informatica* **2018**, *42*, 2173. [CrossRef]
12. Zhou, Y.; Yang, Y.; Liu, H.; Liu, X.; Savage, N. Deep learning based fusion approach for hate speech detection. *IEEE Access* **2020**, *8*, 128923–128929. [CrossRef]
13. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
14. Bahdanau, D.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
15. Bilal, M.; Khan, A.; Jan, S.; Musa, S.; Ali, S. Roman Urdu hate speech detection using transformer-based model for cyber security applications. *Sensors* **2023**, *23*, 3909. [CrossRef] [PubMed]
16. Deng, J.; Zhou, J.; Sun, H.; Zheng, C.; Mi, F.; Meng, H.; Huang, M. Cold: A benchmark for chinese offensive language detection. *arXiv* **2022**, arXiv:2201.06025.
17. Chen, Y.; Zhou, Y.; Zhu, S.; Xu, H. Detecting offensive language in social media to protect adolescent online safety. In Proceedings of the 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing, IEEE, Amsterdam, The Netherlands, 3–5 September 2012; pp. 71–80.
18. Shylaja, S.S.; Narayanan, A.; Venugopal, A.; Prasad, A. Document embedding generation for cyber-aggressive comment detection using supervised machine learning approach. In Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017), Bangalore, India, 13–15 September 2017; pp. 348–355.
19. Bohra, A.; Vijay, D.; Singh, V.; Akhtar, S.S.; Shrivastava, M. A dataset of Hindi-English code-mixed social media text for hate speech detection. In Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media, New Orleans, LA, USA, 6 June 2018; pp. 36–41.
20. Akhter, M.P.; Jiangbin, Z.; Naqvi, I.R.; Abdelmajeed, M.; Sadiq, M.T. Automatic detection of offensive language for urdu and roman urdu. *IEEE Access* **2020**, *8*, 91213–91226. [CrossRef]
21. Lu, N.; Wu, G.; Zhang, Z.; Zheng, Y.; Ren, Y.; Choo, K.R. A framework for hate speech detection using deep convolutional neural network. *IEEE Access* **2020**, *8*, 204951–204962.
22. Lu, N.; Wu, G.; Zhang, Z.; Zheng, Y.; Ren, Y.; Choo, K.R. Cyberbullying detection in social media text based on character-level convolutional neural network with shortcuts. *Concurr. Comput. Pract. Exp.* **2020**, *32*, e5627. [CrossRef]
23. Djandji, M.; Baly, F.; Antoun, W.; Hajj, H. Multi-task learning using AraBert for offensive language detection. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, Marseille, France, 11–16 May 2020; pp. 97–101.

24. Gémes, K.; Kovács, Á.; Reichel, M.; Recski, G. Offensive text detection on English Twitter with deep learning models and rule-based systems. In Proceedings of the Forum for Information Retrieval Evaluation (Working Notes), (FIRE), Bangladesh, India, 13–17 December 2021.
25. Roy, P.K.; Bhawal, S.; Subalalitha, C.N. Hate speech and offensive language detection in Dravidian languages using deep ensemble framework. *Comput. Speech Lang.* **2022**, *75*, 101386. [CrossRef]
26. Aroyehun, S.T.; Gelbukh, A.F. Evaluation of Intermediate Pre-training for the Detection of Offensive Language. In Proceedings of the IberLEF@ SEPLN, Malaga, Spain, 3–6 September 2021; pp. 313–320.
27. Lu, J.; Xu, B.; Zhang, X.; Min, C.; Yang, L.; Lin, H. Facilitating Fine-grained Detection of Chinese Toxic Language: Hierarchical Taxonomy, Resources, and Benchmarks. *arXiv* **2023**, arXiv:2305.04446.
28. Zhou, L.; Cabello, L.; Cao, Y.; Hershcovich, D. Cross-Cultural Transfer Learning for Chinese Offensive Language Detection. *arXiv* **2023**, arXiv:2303.17927.
29. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised cross-lingual representation learning at scale. *arXiv* **2019**, arXiv:1911.02116.