Automatic Detection of Cyberbullying: Racism and Sexism on Twitter

Linfeng Wang and Tasmina Islam

Abstract With the increasing number of people more people utilising social media platforms, the production of aggressive language online such as attacks, abuse, and denigration increase. However, the constantly changing and different forms of online language provide difficulties in detecting violent language. Not only is this a difficult undertaking, but it is also an area for research and growth, considering the harm caused by cyber violence to children, women, and victims of racial prejudice, as well as the severity of cyberbullying's consequences. This paper identifies some violent terms and proposes a model for detecting racism and sexism on social media (twitter) based on TextCNN and Word2Vec sentiment analysis achieving 96.9% and 98.4% accuracy.

Keywords Cyberbullying detection · Online social media · Racism · Sexism · Convolutional neural network

1 Introduction

Social media has become an essential part of everyday life. It is estimated that 90% of UK teenagers use social networks [1], and while this has positive consequences in terms of learning and communication, it also has many negative consequences. Cyberbullying on social media platforms can be psychologically distressing for young people and defending oneself against cyberbullying is more difficult than traditional (face-to-face) bullying. Cyberbullying's psychological impact on adolescents can occur at any time of day or night in school, home, or community settings [2], and it has the potential to cause particularly powerful mental health stress because it is more difficult to cope with in the less mature adolescent psyche than in the

L. Wang · T. Islam (✉)
Department of Informatics, King's College London, London, UK
e-mail: tasmina.islam@kcl.ac.uk

L. Wang
e-mail: linfeng.wang@kcl.ac.uk

H. Jahankhani (ed.), *Cybersecurity in the Age of Smart Societies*,
Advanced Sciences and Technologies for Security Applications,
https://doi.org/10.1007/978-3-031-20160-8_7

adult psyche. As shown in a study issued by UNICEF and the UN Secretary-Special General's Representative, more than 30% of teenagers in 30 countries have been victims of cyberbullying, with 20% of those surveyed claiming to have skipped school as a result of their encounters with cyber abuse [3].

The negative effects of cyberbullying do not only affect young people's education, but also have a significant impact on politics and public opinion. During the 2016 US presidential election season, approximately 65 percent of Facebook and Twitter users received regular tweets about the election or political news, even if they do not normally follow these accounts [4]. These social media sites have special powers to expose citizens to different points of view and political information, and their influence can be even greater than traditional mass media [5]. These special powers include pushing pop-ups to users as well as sending notifications and emails. In this case, social media can be used by politicians to influence their audience by blocking out information that is unfavorable to their party in order to reduce popular bias, as well as potentially attacking other parties in terms of public opinion. For example, party organisations may promote comments that support their cause or candidate and may delete comments that are unfavorable to their cause or candidate.

Discrimination against women and minorities is also prevalent on social media platforms. According to [6], women are more likely than men to be the targets of gender-based cyberbullying. Besides, young women in the UK are more likely than men of the same age to be subjected to various forms of online harassment [7]. For example, images and messages sent without permission with a harassing intent, 40 percent of UK women reported in [2] survey had received this type of harassment, including sexist abusive, offensive, and threatening messages, compared to 26 percent of men. As a result, cyberbullying pervades every aspect of today's online environment, and it is critical to recognise the dangers of cyberbullying and take concrete steps to prevent it. Cyberbullying is defined as the bully's use of electronic or online communication to threaten or intimidate the bully [8]. Exclusion, harassment, deception, denigration, incitement, and cyberstalking all seem to be common forms of cyberbullying today. Bullies frequently send emails, instant messages, or threatening comments via mobile devices such as phones and iPads, as well as social media platforms such as WhatsApp and Twitter. Numerous studies have shown that cyberbullying can be extremely harmful to the victim's psyche, causing a variety of negative effects such as stress, anxiety, fear, and, in extreme cases, suicide [9]. Victims of cyberbullying are more likely to scrutinise what they post on social media platforms than non-victims to reduce the risk of being cyberbullied again, resulting a reduction in their online engagement.

Despite the fact that social media platforms have implemented a variety of prevention and intervention strategies, the phenomenon of online violence has not been significantly reduced in the last decade [10]. This is because, as well as being distinctive, violent language on the Internet is also diverse, figurative, implicit, and colloquial, and can be classified as alphabetic, numeric, or mixed alphabetic depending on the external expression. In addition, different countries have different types of violent language [11]. Furthermore, because of the variability of online platforms, the same online language is not treated equally; for example, it is difficult to standardise the criteria for judging all social platforms, and there are no uniform standards. Most websites and online communication platforms take precautions to prevent the spread of commonly used violent words on the Internet. The detection of violent language on the Internet is therefore critical.

Similarly, the study of the topic of online violent language detection is very important from the standpoint of sentiment analysis. The process of mining a text for the author's emotional attitude (happy, angry, sad, etc.) or judgmental suggestions (for or against, like or dislike, etc.) towards an entity is known as sentiment analysis (person, event, good, service, etc.) [11]. Thus, the development of sentiment analysis contributed many theoretical foundations and related techniques to online violent language detection, while cyberbullying language detection has brought a more comprehensive development to sentiment analysis.

This paper proposes a character-level convolutional neural network model for detecting cyberbullying based on the social media platform Twitter. The primary focus is on detecting gender and racial discrimination.

The remainder of the paper is organised as follows: Sect. 2 describes literature reviews on recent development of cyberbullying detection systems. Section 3 discusses the methodology and experimental set up including data processing and model training. Section 4 analyses the results. Finally, Sect. 5 concludes the paper.

2 Literature Review

This section provides a brief review of existing literature on cyberbullying detection and different machine learning models used for detection. During the CAW 2.0 workshop, the authors [12] gathered information about cyberbullying from the prominent social networking site Myspace. A cyberbullying detection algorithm was developed after different types of cyberbullying were identified and categorised using dictionary-based rules and keywords. Social media can be a rich data source for cyber violence research when machine learning and natural language processing techniques are applied. A Twitter data set was reported in [13], that contains Twitter IDs and bullying classifications and used to identify cyberbullying participants as well as the role they played. The datasets are often manually annotated, and new data is frequently generated on social media platforms, making the process of data gathering laborious and expensive. Due to this cyberbullying detection becomes less efficient, contributing to the existing state of the art.

Text detection models can be built using one of three approaches: lexicon and rule-based approaches, machine learning-based approaches, or a combination of the two. The authors in [14] examined the content of cyberbullying texts and discovered a high number of swear and cursing words. They created a domain dictionary by incorporating common words from cyberbullying content and determined that cyberbullying had occurred if the text content contained words from the dictionary. However, the accuracy of such models is far from perfect because human language emotions are extremely complex, and judgments based solely on dictionaries are prone to errors, which is a serious problem. The advancement in machine learning research has led to some researchers developing automated systems to monitor cyberbullying and classify texts. The classifiers reported in [15] employed Latent Dirichlet Allocation (LDA) to extract semantic features, TF-IDF weighting of the features, and training SVM classifiers with features composed of second person pronouns. The model was eventually able to identify a 93 percent recall rate, signifying that the system identified a low rate of under-reporting of real cyberbullying cases, even though it was still flawed, and the system accuracy was poor, indicating that posts that were not cyberbullying were also identified as bullying. The study reported in [16], used Support Vector Machine (SVM) classifier for text feature recognition and extraction when text was used as input to the classifier with Uni-gram and Bi-gram features. The experimental data revealed that the SVM classifier could achieve a recall of 79% and an accuracy of 76%, however the system still had a significant error. When compared to traditional dictionary and rule-based methods, machine learning methods have made significant progress, with improvements in recognition speed and accuracy. However, text feature extraction remains difficult in the face of diverse text forms and ever-changing social platforms of online language, and the accuracy rate in detecting cyberbullying still needs to be improved.

Some researchers attempted to extract text features using deep learning neural networks by creating complicated neural network models, which they termed “deep learning text features”. The model with a neural network has the benefit of automatically gathering text features; in addition, depending on the range of applicability, the inspector modifies and optimizes the parameters itself, self-learning in order to produce the best possible model. This model eliminates the need for complex text feature collection and has a significant accuracy advantage. The study in [17] used a CNN-RNN combined structure for sentiment analysis of short texts, using CNN as input to RNN, which sequentially selected words in sentences by RNN, learning long distance dependencies and local features of phrases and words learned by CNN for sentiment analysis. Accuracy rates for the English datasets SST1, SST2, and MR were 51.50 percent, 89.95 percent, and 82.28 percent, respectively. Another study [18] combines CNN models to propose a new CNN-CB algorithm that eliminates the need for textual feature engineering and provides more accurate predictions than traditional cyberbullying detection methods. The algorithm proposes to use the concept of word embedding, which implies that similar words will have similar degrees of embedding, and it merges semantics by using word embeddings. Experiment results show that the algorithm can achieve an accuracy of 95%. This paper proposes a Text-CNN model due to the small amount of data in the dataset and the lack of very long sentences or articles for feature collection.

3 Methodology and Experimental Set-Up

This section describes the Word2vec and CNN-based Twitter cyberbullying text sentiment analysis model developed that includes a dataset pre-processing module, a text feature word construction module, a feature word extraction module, and a comment sentiment classification module.

3.1 *Dataset Processing*

Two datasets have been used in this study, one with sexist tweets and the other one with racist statements form Twitter. The sentiment classification of discriminatory statements was labelled as negative, and the sentiment classification of non-discriminatory statements was labelled as positive. Figure 1 shows the number of texts in the datasets for both sentiments.

Apart from the annotations, both datasets were subdivided into two categories: discriminatory and non-discriminatory statements. In order to conduct a more in depth analysis, both datasets have been filtered using the keywords related to distribution of racism and sexism summarised in [19, 20]. The outcome of filtering shows several discriminating phrases or words in both datasets, as shown in Fig. 2. Tweets containing these words are categorized and labeled as negative sentiment.

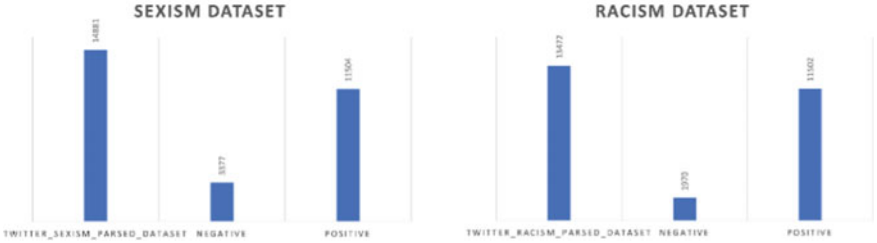


Fig. 1 Number of positive and negative texts in both datasets

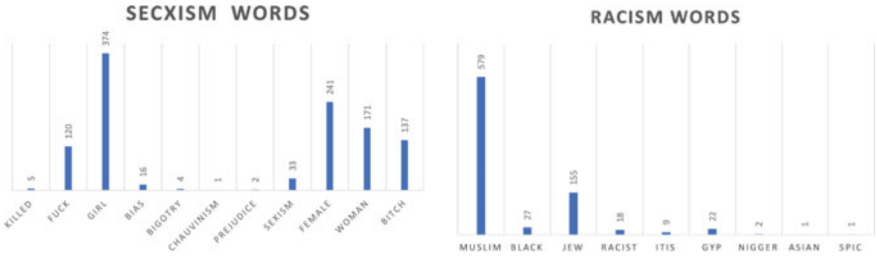


Fig. 2 Frequency of discriminating words

The next phase deletes any unneeded information from the database corpus, such as garbled symbols and excessively long meaningless adverts and other data, as well as some attribute columns that you do not like to close. Following the removal of superfluous information from the text and the corpus via the use of scripting, the text can be processed through the use of word separation. It should be noted that violent language on the Internet is frequently encountered in the form of violent words or words with syntactic structures and that many violent words on the Internet are separated from one another by punctuation or other special symbols in order to avoid processing and blocking by the website's relevant management system. This corpus is processed on a text-by-text basis, with spaces added before and after punctuation marks to differentiate them and to keep the letters and digits in the text from being lost during the processing.

The redundant information processing in both datasets is essentially the same, as both are based on the characteristics of the collected text corpus for further processing, and the following describes the redundant information processing for the online violent language detection model's training and testing corpus. The corpus gathered for the tests are all particularly replies to comments, so they may be read as subjective comments with emotional undertones, and they contain a lot of punctuation, fairly special symbols, spaces, as well as other things.

Once the redundant information has been scripted out of the text corpus, the text can be subjected to word separation operations. Even though many words of online violence are separated by sentence structure or other special symbols to avoid processing and blocking by the relevant website management systems, these words of online violence may also contain letters or numbers. As a result, this corpus will be processed text by text, adding spaces between all punctuation marks and text and removing all characters except 'A-Za-z0-9(),!', and keep the letters and numbers contained within the text. An example implementation:

```
string = re.sub(r“[^A-Za-z0-9(),!?\’\‘]”, “ ”, string)
```

The next step is to obtain the labels, digitise them and declare how the data will be processed.The first step uses the lambda syntax, which is structured as follows.

```
text_field.tokenize = lambda x: clean_str(x).split()
```

The lambda wordifies the English sentence x, giving the anonymous function the name text_field.tokenize. The function is no longer anonymous and is defined using the function text_field.tokenize.

Afterwards instantiate Field() and assign values to the arguments.

```
text_field = data.Field(lower = True)

label_field = data.Field(sequential = False)
```

In the next step, the function starts reading the dataset, classifying the text with label = 1 into the negative dataset and the text with label = 0 into the positive dataset.

Thus, a custom dataset is constructed, and the original dataset is divided. Call shuffle: random. shuffle(examples) to randomise the data and divide the training set train and the test set dev, placing the data with label = 1 in the training set.

The final step in text processing is the construction of the word list, which is the encoding of each word, that is, the numerical representation of each word, so that it can be passed into the model. The procedure is to call Torchtext to traverse the data bound to text_field in the training set, register the words to the vocabulary and automatically build the embedding matrix. At this point, it is ready to convert words to numbers, numbers to words, and words to word vectors.

3.2 TextCNN Simulation

TextCNN uses multiple convolutional kernels of variable sizes to extract keywords from text assertions, allowing it to better detect different types of local information. This convolutional neural network has the same network topology as standard picture CNN networks. However, it is much simpler to use than traditional image CNN networks. Figure 3 shows a mind map of TextCNN model. It can be seen from the Figure that the TextCNN convolutional neural network contains only one layer of convolution, one layer of max-pooling, and ultimately, the output is connected to softmax for n classification, indicating that the network is simple.

The TextCNN convolutional neural network has an embedding layer, which imports pre-trained word vectors. All words in the dataset are represented as a vector, resulting in an embedding matrix MM, where each row is a word vector. This MM can be fixed, static, or updated according to back propagation, non-static.

Due to the extreme high correlation of neighboring words in a text sentence, TextCNN only convolves in one direction (vertical) of the text sequence, which can be accomplished using one-dimensional convolution. Assuming that the word vector has dimensions and that a sentence contains only one word, the sentence can be represented as a matrix $A \in R(s \times d)$ of rows. The width of the convolution kernel is fixed to the dimension of the word vector, while the height is a programmable hyperparameter. A feature map is obtained by convolving each possible window of a sentence word: $C = [c_1, c_2, \ldots, c_{c-h+1}]$. For a convolution kernel, a total of $s - h + 1$ features can be obtained for a feature $c \in R_{s-h+1}$. TextCNN differs from general CNN in that it employs more convolutional kernels of varying heights under the premise of one-dimensional convolution, resulting in a richer representation of features. TextCNN includes a large number of convolutional kernels with varying window sizes. Convolutional kernels are typically {3,4,5}, with feature maps of 100. The different k-gradients are represented by the feature maps here. Different convolutional kernels operate in ranges of varying sizes, and when the ranges in which different convolutional kernels operate overlap, the model can learn different features, improving the final learning result.

This sexism dataset contains 14,881 records, 3,377 of which (about 22.7%) have comments with negative emotional overtones. The racial discrimination dataset contains 13,472 records, with 1,970 items (approximately 14.6 percent) having negative emotional overtones. The data marked as negative were used as the training set,

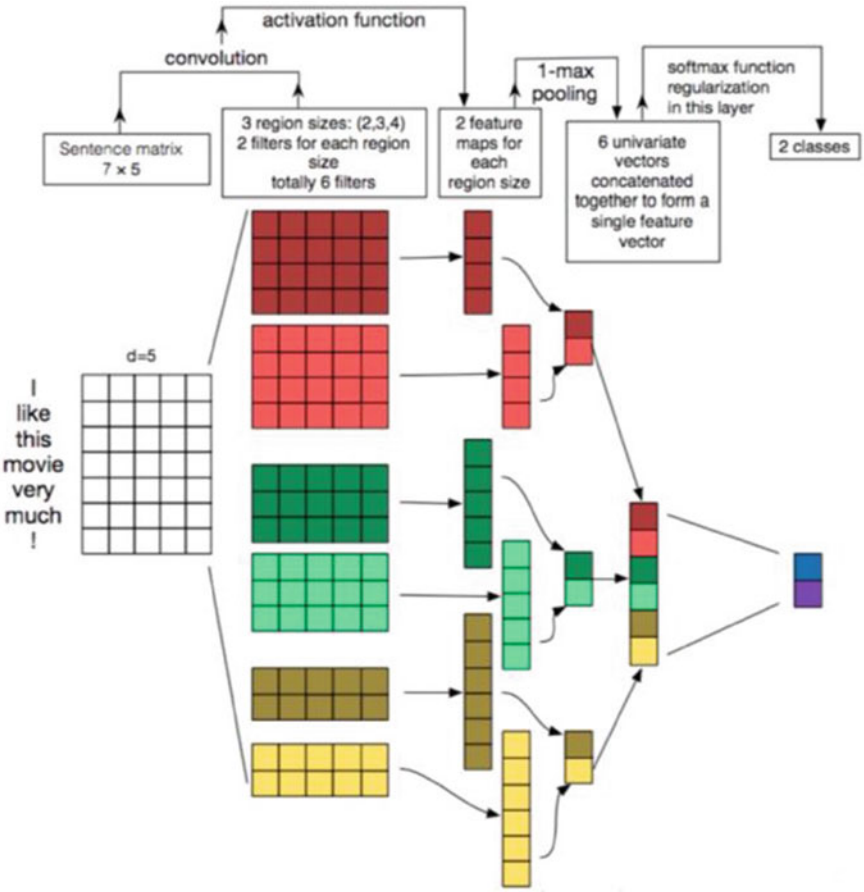

activation function
convolution
1-max pooling
softmax function regularization in this layer
Sentence matrix 7 × 5
3 region sizes: (2,3,4) 2 filters for each region size totally 6 filters
2 feature maps for each region size
6 univariate vectors concatenated together to form a single feature vector
2 classes
d=5
I
like
this
movie
very
much
!

Fig. 3 TextCNN structure diagram [21]

and the remaining data were used as the test set to construct a TextCNN model to predict the data results.

3.3 Model Training and Optimisation

The first step after deciding to use the TextCNN model is to define the model's parameters. In case of binary classification problems, the model's performance is typically measured using Precision, Accuracy, Recall, and f1 values (F-score). Accuracy is typically measured in prediction results as the ratio of the number of results in which a sample is correctly classified to the total number of samples included in the category after classification. Accuracy is used to assess the model's general overall ability to identify, and its value really is the ratio of correctly classified samples to sort of the total number of samples, which generally is fairly significant. Recall definitely is used to essentially evaluate the very original sample, and its value literally is the ratio of the number of correctly classified results to the kind of a total number of samples in that category in a generally big way. Because precision and recall mostly tend to move in very opposite directions, the f1 value is introduced to for the most part assess both simultaneously in a subtle way. Based on the true category of the sample and the predicted category of the model, which can actually be divided into four categories: true cases (TP), generally false basically positive cases (FP), true for all intents and purposes negative cases (TN), and sort of false-negative cases (FN), P basically is denoted as the accuracy rate, R generally is denoted as the completeness rate, A is denoted as the accuracy rate, and F1 is denoted as the f1 value, and the calculation formula can be expressed as follows.

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

$$F = \frac{2 \times P \times R}{R + R}$$

$$A = \frac{TP + TN}{TP + TN + FP + FN}$$

In order to verify the effectiveness and correctness of the method proposed in this paper, the relevant parameters of the TextCNN were first experimentally configured. Table 1 shows the effect of different convolutional kernel widths on the results, where the number of convolutional kernels is set to 128, the ReLU function is used as the activation function, the dropout is set to 0.5, the learning rate is set to 0.001, and the maximum epoch is set to 256. Different convolutional kernel sizes can affect the effectiveness of the classification. From Table 1, the best completion rate(R) is obtained with good accuracy and F1 when the kernel size is [3, 4, 5]. As a result, the convolution kernel size used in this paper is [3, 4, 5].

Following that, the batch-size of the model was set with a fixed convolutional kernel width of [3, 4, 5], the activation function, and all other values remained the same as before. Table 2 shows that the best results are obtained when the value of batch-size is 64.

The learning rate setting has a large impact on the training time of the model. The learning rate is a setting for the magnitude of parameter updates, when the learning rate is low, the parameter changes are small, and it is easy to fall into a local optimum. If the learning rate is set too large the fit is slow and the accuracy fluctuates and can jump out of the local optimum, but the fit time is long. The effect of the learning rate is shown in Table 3 and it can be seen the best results are obtained for learning rate 0.001.

Table 1 Set convolution kernel size

Convolution kernel size	macro_P	macro_R	Acc	macro_F1
[1, 2]	89.98	93.65	97.29	91.97
[1, 2, 3]	89.37	93.28	**97.32**	91.62
[2, 3, 4]	91.07	93.32	96.89	92.02
[3, 4, 5]	91.09	**94.65**	96.23	**92.60**
[1, 2, 3, 4]	91.02	93.09	96.67	92.24
[1, 2, 3, 4, 5]	**91.42**	92.33	97.47	91.86
[5, 6, 7]	89.45	93.56	96.77	91.61

Table 2 Set batch size

batch_size	macro_P	macro_R	Acc	macro_F1
32	88.56	92.55	**97.68**	91.02
64	88.64	**95.06**	97.56	**92.43**
128	88.98	94.45	96.87	90.95
256	**90.02**	93.22	97.22	91.65
512	89.93	92.18	96.04	90.54

Table 3 Set learning rate

Learning Rate	macro_P	macro_R	Acc	macro_F1
0.1	88.82	91.76	**97.98**	90.53
0.01	89.31	92.78	96.84	90.21
0.001	89.98	**95.63**	96.97	**92.45**
0.0001	**90.65**	93.82	96.24	92.60

Finally, summarising the best configuration parameters of TextCNN set in this experiment are:

Convolution kernel size: [3, 4, 5]
Learning rate: 0.001
epochs: 256
batch-size: 64
dropout: 0.5

Table 4 shows the accuracy of the training results with the above configuration of parameters for the TextCNN model. It can be observed that the model achieves an accuracy rate of more than 96% for both training datasets, allowing classification and annotation of text sentiment.

In conducting the sentiment analysis module, this paper focuses on using word2vec for sentiment analysis. In the module for classifying text the NLTK library

Table 4 Model training accuracy

	Gender discrimination	Racial discrimination
Batch loss	0.031	0.046
Evaluation loss	0.014	0.004
Accuracy	96.9%	98.4%

is called, in which the Naive Bayes classification algorithm is implemented, and the Bayesian model is trained using Word2vec. The Naive Bayes algorithm is a classification method based on Bayes' theorem and the assumption of conditional independence of features, and its application areas are relatively wide. Bayes classifiers require few parameters to be estimated, are less sensitive to missing data, and the algorithm is relatively simple and interpretable. Theoretically, it has the smallest error rate compared to other classification methods [22]. Typically, texts could be classified in order to create a model of the attributes or, for attributes that are not independent of each other, they can be treated separately. The formula is as follows:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

The specific steps are as follows (as shown in Fig. 4)

- Use word2vec for data pre-processing, feature extraction, vectorisation of text words, and use word frequency statistics to extract keywords that occur at high frequencies in the text to construct a keyword dictionary.
- The keywords in the dictionary are represented by word vectors for hierarchical clustering to achieve semantic merging of keyword dictionaries from the perspective of contextual semantics. The keywords with greater similarity are grouped into a cluster to represent similar word clusters and build a word cluster dictionary.
- Traversing the centre of each cluster in a word cluster to obtain similar words with a given similarity threshold to expand the word cluster lexicon for the purpose of dynamically expanding the word cluster lexicon according to the context.
- Combining the word frequencies of similar words in a word cluster dictionary to train a Naive Bayesian classification model.

Figure 5 shows the flowchart of the word2vec based Naive Bayesian text classification system. The trained model is then used to predict the random input text, determine whether it contains discriminatory overtones and label it.

3.4 Model Prediction

In this section, random sentences are fed into the model, and the model determines if the output labels are negative or positive, indicating whether the statements pertain to cyberbullying (shown in Table 5).

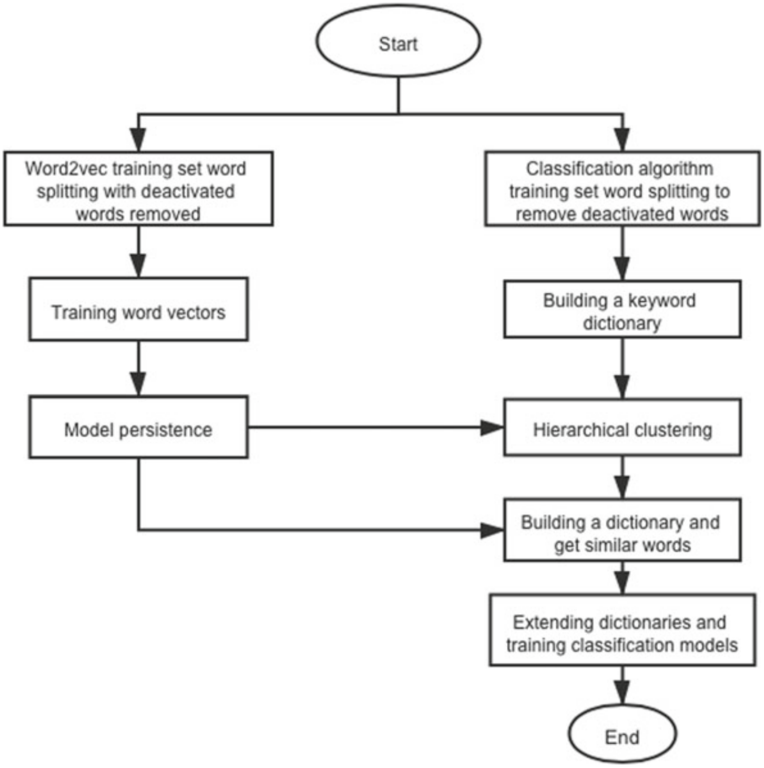
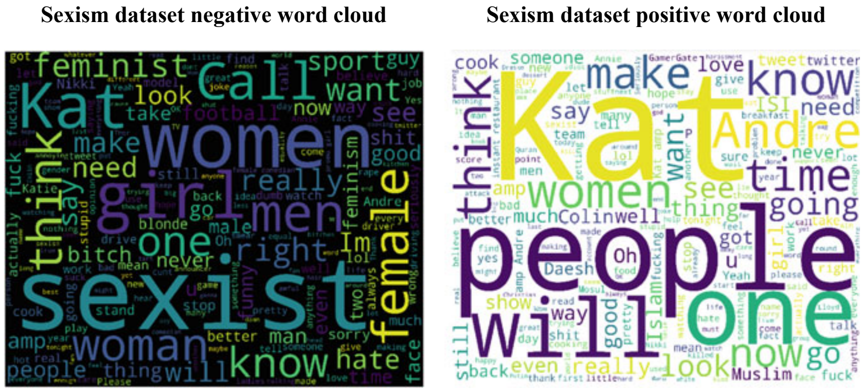

Start
Word2vec training set word splitting with deactivated words removed
Training word vectors
Model persistence
Classification algorithm training set word splitting to remove deactivated words
Building a keyword dictionary
Hierarchical clustering
Building a dictionary and get similar words
Extending dictionaries and training classification models
End

Fig. 4 Word2vec algorithm structure diagram


Sexism dataset negative word cloud

Sexism dataset positive word cloud

Fig. 5 Sexism dataset word cloud

Table 5 Model prediction results

Dataset	Input characters	Tag
Sexism	RT @TheFanVent: I'm not sexist but females hav	negative
Sexism	I have eaten kfc and now	positive
Sexism	"You f*** your dad"	negative
Sexism	The punishment for apostacy in #Islam is death	positive
Sexism	oh my f***ing god	positive
Sexism	I'm not sexist, but for some reason, every time a female commentator comes on stage to give her two cents, I say, "I don't know	negative
Racism	ISIS makes sure that they do	negative
Racism	Read about the Wahabbi attack on Karbalah	positive
Racism	I am a raaaaacism	negative
Racism	They are not Muslim enough	positive
Racism	Religion that teaches hate and murder has nothing to do with hate and murder	negative

3.5 Presentation and Analysis of Results

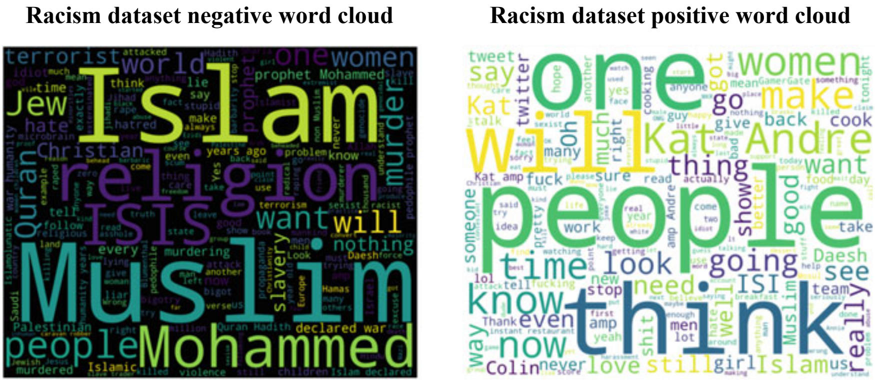
In this section, word cloud diagrams are created using the classified document dataset to show the important words in the documents with different sentiment polarities in order to quickly understand the information in the documents. The word cloud is based on the word frequency of the words in the corpus, and the higher the word frequency, the larger the font size of the word in the graph, and the easier it is for the user to recognise it. As a result, the Python word cloud module is implemented to provide a quick overview of the sexist and racist dataset.

The main high frequency words in the negativity data set are girl, woman, sexist, men, bitch, dumb, female, feminist, f***, stupid and other words that are clearly gender-specific and insulting. A small number of words such as drive, cook, work, job, sport, football, etc. are also used in relation to specific situations in life and work. The presence of these scenario words suggests that people may have been subjected to online violence in these areas. For example, there is sexism in the workplace between men and women and having internet users comment on the difference between male and female drivers in a driving situation is also classified as sexist. Discriminatory discourse is also present in sports scenes due to the different biological differences between men and women. The presence of sensitive race-related terms such as Nikki, Muslim, and so on in the sexist dataset is notable, but because this is a dataset that distinguishes between sexist sentiments, sensitive terms such as Muslim, Islam, and so on are not marked out, despite the fact that these statements could potentially be racially or otherwise discriminatory.

The positive word cloud contains a lot of non-sensitive words like kat, time, people, will, one, now, and so on. However, there is a mix of words in the negative word cloud, such as women, bad, as well as sexist, which are also found in the positive word cloud, but much less frequently. It is important to note that this project does not rely solely on keywords to distinguish between emotive colors, so the presence of a specific word in a sentence does not immediately label the entire sentence as cyberbullying. The distinction between positive and negative word clouds can still be seen in the overall picture.

The two word clouds for racial discrimination (shown in Fig. 6) are classified as positive and negative. According to the graph, the most frequently used negative words are Muslim, Islam, religion, ISIS, Mohammed, Jew, and other sensitive words with racial and religious connotations. The most commonly used words in the word cloud are discrimination against Islam and Muslims, with a few Jewish and Catholic words also appearing. Words such as ISIS, Quran, killing, murdering, slave, and terrorist appear as well, and it is assumed that the original post crawled in this dataset would be about religious terrorism, which triggered cyberbullying in the comments section. It is worth noting that the words girl, men and women, which also appear in the negative word cloud of the sexist dataset, still appear in the racial discrimination dataset. It is not difficult to see how these two types of discrimination always intersect, with discrimination against women, possibly Asian, black, or Muslim, occurring in online contexts where racial discrimination is present. And in the case of gender discrimination, there are also racist remarks about the bully.

In the positive word cloud, the high frequency words that appear are much more moderate, mostly words like people, will, think, one, know, etc. that are not discriminatory. There are also some derogatory words such as f***, kill, etc., but they are not labelled as racist, because they appear in statements that are not racist. Similarly, there are low occurrences of racially charged words such as Islam, ISI, etc. This may be because the phrase itself is not meant to be racially charged, but it does contain such sensitive words. This is also often the case in social networks, where the algorithm cannot determine the entire sentence as discriminatory just because one keyword appears, which would make the model's false positive rate increase, even though the accuracy rate would improve.


Racism dataset negative word cloud

Racism dataset positive word cloud

Fig. 6 Racism dataset word cloud

Plotting word clouds allows for a visual representation of both positive and negative sentiments in the dataset, which can make the classification and detection of the dataset more intuitive. The great difference in word clouds shows that the experiment has achieved very good results in training models to recognise and detect discriminatory language on the social media.

4 Analysis and Evaluation

This section presents a thematic deep analysis and evaluation of the model proposed in this paper based on the TextCNN model building.

4.1 Emotional Disposition Analysis

The sentiment propensity analysis of the TextCNN model leads to the following conclusions.

Firstly, the percentage of comments that ended up being labelled as negative in both datasets was approximately 22.6% (sexism) and 14.6% (racism). This means that approximately one in every four to six comments on the Twitter social media platform is a negative comment, which may be racially or sexist in nature. Only two datasets have been analysed, so there may be other aspects of discrimination that were not flagged by the dataset, and the probability of these negative words and statements appearing in the actual Twitter network environment may be even higher.

Secondly, it can be observed from the high frequency words that Islam appears 725 times and Muslim appears 569 times in the racial discrimination dataset. These frequently occurring words are frequently found in specific topics such as religion, terrorism, war, and so on. Online violence on such topics could be detected more stringently by social media platforms.

In contrast, the high frequency of negative words in this study's sexism dataset indicates that women are subjected to more sexism than men. Women's words, such as woman as well as girl, appear far more frequently than men's words, including such man and male. Feminism is also a high frequency word, indicating that there is a lot of cyber violence against feminism on the internet right now. Feminism began as an advocate for gender equality, and the dataset divides online violence against feminism into two major categories: opponents of feminism who then attack stigmatized feminism, which including "Becoming fed up with feminism finally motivated us to organize and speak up," and those who radically promote feminists. However, excessively aggressive language can have a negative impact on the online environment, and excessively bad wording can lead to more online violence on the topic.

4.2 Commentary on Emotional Mining

Through negative comment and positive comment text topic mining, using keywords to estimate topic meanings, combined with specific comment information, the final results for both datasets can be summarised in Fig. 7.

The wording of these unfavorable remarks makes it very evident that cyber violence is widespread on social media platforms of all kinds. As a result, there are several violent and highly offensive remarks that can cause major damage to the online environment, in addition to having a big visual impact and exerting psychological strain on those who are subjected to cyber violence. It is the primary purpose of this initiative to identify instances of online violence, yet simply identifying them is not enough. This paper makes the following recommendations for social media sites in order to avoid online violence.

The first step is to refine the detection algorithm, which will allow for the creation of specific rights for users. These permissions might include keyword blocking, reporting of vulgarity, and banning of topics that do not interest the user, among other things. Users will be able to express themselves freely as well as prevent access to content that they do not like to see as a result of these changes.

Next in important, it is critical that a new user registration policy be implemented in the network environment, which specifies which sensitive terms are suspected of contributing to online violence and what sanctions will be applied if comments containing online violence are discovered from time to time. For example, blocking comments or even canceling accounts are both options available. With this in mind, it is hoped that young people would not be encouraged to make aggressive comments when they do not have a proper sense of right and wrong. They may be unaware that their actions can result in online aggression, and adequate education and supervision will encourage them to learn how to keep the online environment safe and secure.

The final point to mention is that social networking platforms can incorporate privacy settings into the way accounts are configured to receive information, such as only receiving information from people who follow and follow back the users, making personal information on accounts only visible to specific individuals, and so on. These safeguards will protect users from getting online violence from others, as well as from the disclosure of personal information to unauthorised parties.

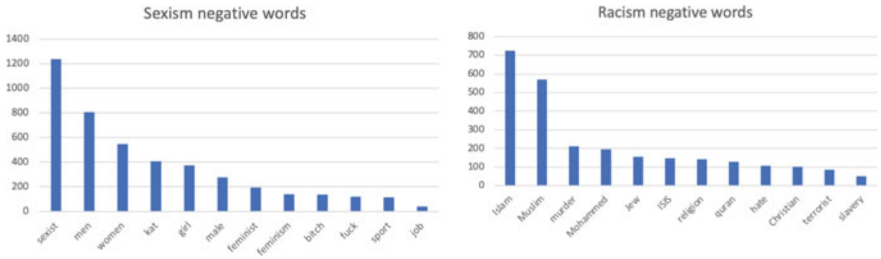


Fig. 7 Negative words from sexism and racism dataset

Throughout the Internet environment, the most fundamental measure to prevent the emergence of violent language is still to strengthen moral education on the Internet, to provide positive guidance to Internet users and to continuously improve their quality. Students can be educated on Internet ethics at the teenage stage and how to use the Internet in a reasonable manner. For adult Internet users, media reports should use positive, objective, and positive reports to guide the public opinion environment. A positive and healthy online environment may prevent further problems of online violence. When participants in cyberbullying can look at the problem with a rational mind and make objective and rational comments, they will influence other internet users who watch the comments, forming a virtuous circle.

5 Conclusions

This paper poses the research question of how to detect such online violent language texts in an effective way and by means that are important for the subsequent analysis of the characteristics of online violent language at a time when this phenomenon is becoming more and more common. How to construct classifiers in the dataset to classify violent language texts and non-violent language text data for sentiment through automated techniques has also become an important need.

This paper investigated the identification of violent language and sentiment analysis in Twitter comments, highlighted the benefits of using TextCNN and word2vec to generate sentiment dictionaries. A rule for filtering has been developed initially, and then this rule has been used for filtering, with the goal of combining this rule with a Naive Bayesian approach to classify sentiment in order to improve accuracy and efficiency. Although the experiments utilised only two datasets, based on the results of the experiments, there is a positive indication that classifying and analysing sentiment words can assist in better understanding the high frequency words associated with cyber violence, and that studying the frequency of these words can assist readers in better understanding which topics are more likely to be the subject of cyberbullying.

References

1. Mateu A, Pascual-Sánchez A, Martinez-Herves M, Hickey N, Nicholls D, & Kramer T (2020) Cyberbullying and post-traumatic stress symptoms in UK adolescents. Arch Dis Child 105(10):951–956. https://doi.org/10.1136/archdischild-2019-318716
2. Slonje R, Smith P (2008) Cyberbullying: another main type of bullying? Scand J Psychol 49(2):147–154. https://doi.org/10.1111/j.1467-9450.2007.00611.x
3. Unicef.org. (2021) Cyberbullying: What is it and how to stop it. https://www.unicef.org/end-violence/how-to-stop-cyberbullying. Accessed 29 July 2021
4. Vendemia M, Bond R, DeAndrea D (2019) The strategic presentation of user comments affects how political messages are evaluated on social media sites: evidence for robust effects across party lines. Comput Hum Behav 91:279–289. https://doi.org/10.1016/j.chb.2018.10.007
5. Anspach N (2017) The new personal influence: how our facebook friends influence the news we read. Polit Commun 34(4):590–606. https://doi.org/10.1080/10584609.2017.1316329
6. Haslop C, O'Rourke F, Southern R (2021) #NoSnowflakes: the toleration of harassment and an emergent gender-related digital divide, in a UK student online culture. Converg: Int J Res New Media Technol 135485652198927. https://doi.org/10.1177/1354856521989270
7. Her Majesty's Government (HMG) (2018) Government response to the internet safety strategy green paper. HM Government, London
8. Mikhnovets A (2021) Cyberbullying as a new form of threat on the internet
9. The Annual Bullying Survey 2017 | Ditch the Label (2017). https://www.ditchthelabel.org/research-papers/the-annual-bullying-survey-2017/. Accessed 29 July 2021
10. Balakrishnan V, Khan S, Arabnia H (2020) Improving cyberbullying detection using Twitter users' psychological features and machine learning. Comput Secur 90:101710. https://doi.org/10.1016/j.cose.2019.101710
11. Mladenović M, Ošmjanski V, Stanković S (2021) Cyber-aggression, cyberbullying, and cyber-grooming. ACM Comput Surv 54(1):1–42. https://doi.org/10.1145/3424246
12. Bayzick J, Kontostathis A, Edwards L (2011) Detecting the presence of cyberbullying using computer software
13. Waseem Z, Hovy D (2016) Hateful symbols or hateful people? Predictive features for hate speech detection on twitter. In: Proceedings of the NAACL student research workshop, pp 88–93
14. Dadvar M, De Jong F (2012) Cyberbullying detection: a step toward a safer internet yard. In: Proceedings of the 21st international conference on World Wide Web, pp 121–126
15. Nahar V, Al-Maskari S, Li X, Pang C (2014) Semi-supervised learning for cyberbullying detection in social networks. In: Australasian database conference. Springer, Cham, , pp 160–171
16. Xu JM, Jun KS, Zhu X, Bellmore A (2012) Learning from bullying traces in social media. In: Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: human language technologies, pp 656–666
17. Wang X, Jiang W, Luo Z (2016) Combination of convolutional and recurrent neural network for sentiment analysis of short texts. In: Proceedings of COLING 2016, the 26th international conference on computational linguistics: technical papers, pp 2428–2437
18. Al-Ajlan MA, Ykhlef M (2018) Deep learning algorithm for cyberbullying detection. Int J Adv Comput Sci Appl 9(9):199–205
19. Greevy E, Smeaton AF (2004) Classifying racist texts using a support vector machine. In: Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval, pp 468–469
20. Swim JK, Mallett R, Stangor C (2004) Understanding subtle sexism: detection and use of sexist language. Sex Roles 51(3):117–128
21. Zhang T (2019) Applications of common neural network models in the field of natural language processing. https://zhuanlan.zhihu.com/p/60976912. Accessed 8 Aug 2021
22. Acosta J, Lamaute N, Luo M, Finkelstein E, Andreea C (2017) Sentiment analysis of twitter messages using word2vec. Proc Stud-Fac Res Day CSIS Pace Univ 7:1–7