Peer-to-Peer Networking and Applications (2020) 13:1567–1576
https://doi.org/10.1007/s12083-019-00832-1

Toward efficient and effective bullying detection in online social network

Jiale Wu[1] · Mi Wen[1] · Rongxing Lu[2] · Beibei Li[3] · Jinguo Li[1]

Received: 15 August 2019 / Accepted: 28 September 2019 / Published online: 28 April 2020

Abstract

With the advances of Information Communication Technology (ICT) and the popularity of intelligent terminals, Online Social Network, which is characterized by powerful functions of information publishing, dissemination, acquisition and sharing, has attracted a huge number of users and become one of the most popular internet application services currently. However, the growth of Online Social Network has also led to the emergence of cyberbullying issues. Information spreads extremely fast via Online Social Network, making the harm caused by cyberbullying grow exponentially with time. As a result, it becomes critical to detect the cyberbullying in a quick and efficient way. In this paper, in order to solve this challenge, we propose an improved TF-IDF based fastText (ITFT) model for effective cyberbullying detection. Specifically, in our proposed scheme, we improve the TF-IDF algorithm by adding the position weight, keywords are extracted by the improved algorithm and used as input to achieve the purpose of filtering noise data to improve the accuracy. We use the fastText to construct a binary classifier to categorize the input data. Extensive experiments are conducted, and the results demonstrate that our proposed scheme can achieve better efficiency and accuracy in cyberbullying detection as compared with baselines.

Keywords Cyberbullying detection · Online social network · Text classification · Natural language processing

1 Introduction

With the advances of internet and information communication tools, a new form of bullying has emerged in Online Social Network, named cyberbullying [1]. Essentially, cyberbullying refers to the use of information and communication technologies such as email, instant text messages, personal websites or online personal voting sites to intentionally and repeatedly commit malicious acts aimed at harming others. The harm caused by the phenomenon of cyberbullying has gradually attracted the attention of the society, and an increasing number of researchers have put their efforts in the research of cyberbullying [2].

Compared with traditional bullying, cyberbullying may be more harmful. This is due to cyberbullying uses the Online Social Network as the medium and spreads faster in a larger scale, as shown in Fig. 1. Once the bully publishes the victim's personal information, images, videos or rumors to the internet, in a very short period of time, the information will be viewed by netizens all over the country and even around the world. It can lead to high psychological stress towards the victims and even threaten the personal and property safety of the victims. On the

This work is supported by the National Natural Science Foundation of China under Grant No.61872230, No.61702321 and No.61572311

✉ Mi Wen
miwen@shiep.edu.cn

Jiale Wu
jiale.w@qq.com

Rongxing Lu
RLU1@unb.ca

Beibei Li
libeibei@scu.edu.cn

Jinguo Li
lijg@shiep.edu.cn

1 College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, 200090, China

2 Faculty of Computer Science, University of New Brunswick, Fredericton, E3B 5A3, Canada

3 College of Cybersecurity, Sichuan University, Chengdu, 610065, China

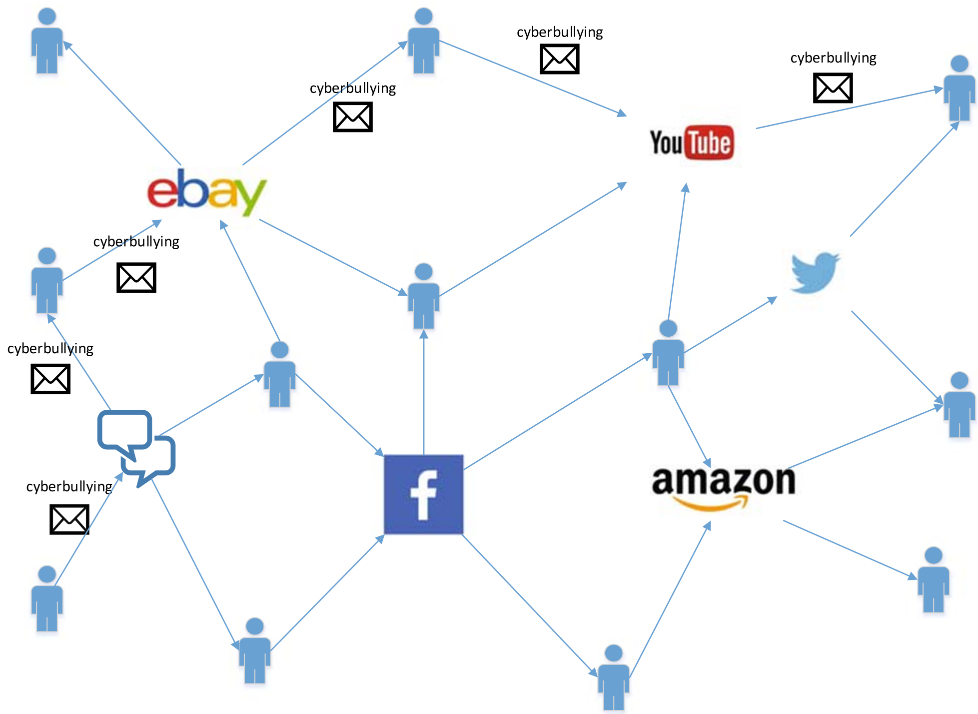


Fig. 1 Cyberbullying spreads faster and more widely through online social network

other hand, in real life, although some bullying is for fun, bullies are not unscrupulous. While in cyberbullying, because bullies are hidden in the Internet, bullying can escape punishment and responsibility, bullies can openly commit bullying without fear of punishment, which makes bullies more unscrupulous, and even easily bully victims in multiple capacities [3].

Because of the concealment of cyberbullying, the monitoring of cyberbullying becomes challenging. Currently, the main monitoring method of cyberbullying is relies on users' marking of cyberbullying information. However, not all bullying information can be marked. In addition, due to the rapid dissemination of network information, the sooner the cyberbullying information is detected, the less harmful the cyberbullying information will cause [4]. Therefore, it is very pressing to detect the cyberbullying in a quick and efficient way.

Recently, many researchers have worked with different Natural Language Processing techniques [5], and text classification algorithm has been commonly used to resolve cyberbullying detection problem, as it is a special binary classification problem and can be defined as a task of classifying texts as cyberbullying or normal message. As for text classification, many mature models have been proposed, including the Logistic Regression (LR) classification model, Support Vector Machine (SVM) model, and neural network classification model. Convolutional Neural Network (CNN) is one of the popular models, which can effectively capture the local correlation such as n-gram [6]. However, it is unable to model longer sequence information. To address this issue, Tang et al. [7] proposed a method that represent context information by Recurrent Neural Network (RNN). However, due to the complexity of deep learning models, the training time increases sharply with the growth of the amount of training data. In order to solve this challenge, Mikolov et al. [8] proposed a text classification model, named fastText. In terms of the computational efficiency, fastText outperforms the most advanced depth neural network model by several orders of magnitude, and the accuracy of classification is almost the same. Although fastText significantly improves the speed of text classification, its classification accuracy is slightly lower than CNN.

Motivated by the above-mentioned, in this paper, we attempt to jointly address the above challenges by proposing an improved TF-IDF based fastText (ITFT) for cyberbullying detection. Firstly, we employ skip-gram to embed word. Secondly, the weight of each word is calculated based on the improved TF-IDF algorithm, and then the extraction of keywords is on the basis of the weight of each word. After training set and test set are processed, only keywords are retained, so as to achieve the purpose of filtering noise data. Finally, the filtered data are used for training and text classification. We evaluate our model over a real-world set, and the experimental results show that our model achieves both efficiency and accuracy improvements in text classification as compared with baselines. Concretely, the contributions of this paper can be summarized as follows:

- The traditional TF-IDF algorithm does not distinguish the position of feature words in documents, so we improve the TF-IDF algorithm by adding the position weight, and the experiments show that it can really improve the accuracy of cyberbullying detection.
- We propose an improved TF-IDF based fastText (ITFT) model for cyberbullying detection. The improved TF-IDF algorithm is adopted for keywords extraction to filter the noise data, by comparing with the baseline – the fastText model, our improved model can improve the efficiency and accuracy of cyberbullying detection.

The remainder of this paper is organized as follows. In Section 2, some related works are introduced. Then, in Section 3, we describe the methodologies used in our proposed scheme, including word embedding, keywords extraction, and text classification, in Section 3. After that, we evaluate our proposed scheme with extensive experiments in Section 4. Finally, we draw our conclusion and identify our future work in Section 5.

2 Related work

With the spread of cyberbullying on social media and its negative impact on young people gradually expanding, the research on the detection of cyberbullying becomes pressing in recent years. The current research methods mainly use machine learning and natural language processing technology to identify the characteristics of cyberbullying. In 2017, Salawu et al. [9] divided the existing research methods into four categories: supervised learning, lexicon based, rule based and mixed-initiative approaches.

Supervised learning based methods usually use classifiers to develop predictive models for cyberbullying detection. Nandhini and Sheeba [10] proposed a cyberbullying detection system using Levenshtein algorithm and Naive Bayes classifier to identify and classify cyberbullying activities such as fire, harassment, racism and terrorism in Online Social Network. Based on the characteristics of individuals, social network and their content, Squicciarini et al. [11] used C4.5 decision tree classifier to detect and classify cyberbullying. Through feature selection of information (including skip-gram), Chavan and Shylaja [12] improved the accuracy by 4% compared with the results of SVM and LR classifier without feature selection.

Lexics-based systems detect cyberbullying by identifying whether or not it contains specific bullying information. Fahrnberger et al. [13] proposed a cyberbullying detection algorithm based on 4-CBAF and SecureString 2.0, named SafeChat, where SecureString 2.0 is responsible for filtering the display terms in encrypted messages, and 4-CBAF is responsible for verifying the identity of the information source and authorizing the sender. Perez et al. [14] also proposed a security model based on analyzing instant messages to detect cyberbullying.

Rule-based approach matches the text with predefined rules for identifying bullying. Serra and Venter [15] introduced the concept of dynamic problem solving based on neural network system to dynamically identify threats based on the risk characteristics of individual users in order to solve the conflict situation of identifying risks. Ying et al. [16] proposed the Lexical Syntactical Feature (LSF) to identify offensive contents in social media by combining the Features including conversation history and writing style, and experiments showed that LSF performs better than SVM and Naive Bayes. Bretschneider et al. [17] proposed a pattern-based approach to detect cyberbullying by first standardizing the text, and then finding the relevant personnel through the identification module. Compared with the baseline, the performance of the pattern-based system had been greatly improved. Agrawal and Awekar [18] showed that DNN models can be used for cyberbullying detection on various topics. But DNN has a lot of parameters to adjust, manual attempts can be extremely hard and slow therefore, metaheuristic optimization algorithm is incorporated to find the optimal or near optimal values [19].

The hybrid proactive approach combines one or more of the above-mentioned methods. By training Naive Bayes, C4.5 decision tree, and SVM classifier, Dadvar et al. [20] combined the training results with Multi-Criteria Evaluation System (MCES) into a mixed system, and they found that the performance of the mixed system was better than that of any independent system. Later, Dadvar et al. [21] also adopted a hybrid active method to detect cyberbullying, which weights user information, including age, gender, registration time, etc., and then inputs these weighted information into MCES. Their experiments showed that this preprocessing procedure can improve the classification performance of the model. Silva et al. [22] detects cyberbullying by Combination of Textual, Visual and Cognitive.

3 Our proposed scheme

In this section, we present our proposed scheme, an improved TF-IDF based fastText (ITFT) model for effective cyberbullying detection. We note that, although fastText has the advantages in correct classification and fast speed, noise data will inevitably be introduced, as the input contains all words in the text. Therefore, in order to solve the problem, we propose an improved TF-IDF based fastText model for text classification. As illustrated in Fig. 2, the details of joint training process are described as follows: we first transform

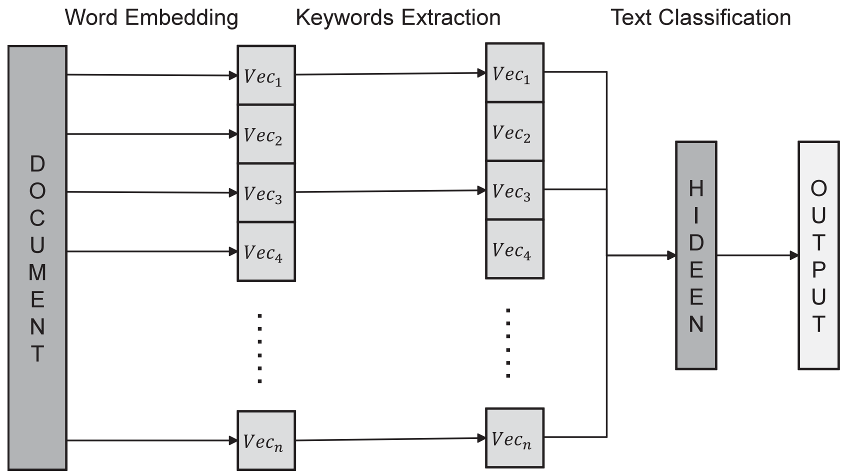


Fig. 2 Illustration of ITFT model, including: word embedding, keyword extraction and text classification

words into vectors via word embedding, then, the weight of each word is calculated based on the improved TF-IDF algorithm, and the keywords are extracted according to the weight of each word, and only the keywords are retained to achieve the purpose of filtering noise data. Finally, we use the filtered data for text classification via fastText.

3.1 Word embedding

Since our raw data are words, computers are hard to directly understand them. In order to enable computers to process natural languages, word embedding needs to be employed first. Word embedding aims to transform words into distributed representations which capture syntactic and semantic meanings of the words. Recent research has shown that they can accurately capture semantic and grammatical information about words. Using word embedding has become common practice for enhancing many other Natural Language Processing tasks.

In this work, we will use the skip-gram model to train word embedding [23]. The skip-gram model is based on the MNLM model [24] and the C&W model [25] to retain its core parts to obtain word vectors in a more efficient way. The goal of skip-gram is to predict the context probability based on the current words. The weight obtained after the iteration is the word vector we need. Our raw data are the sentence s consisting of m words $s = w_1, w_2, \cdots, w_m$. Firstly, we transform words into one-hot vectors $w_i = v_1, v_2, \cdots, v_n$, the one-hot vector is made up of a number of '0's and one '1'. With the position of the 1 representing the corresponding word, and the other positions filled with zeros, e.g., $v_i = [1, 0, 0 \cdots 0]$. As we can see in Fig. 3, the skip-gram model is a single neural network. The input of skip-gram is the one-hot vectors of w_i, the output is the one-hot vectors of the N words before w_i and the ont-hot vectors of the N words after w_i. After training, we can get the weight of the network, and the vectors v_i of w_i is the $weight_i$. The objective function of the skip-gram is:

$$J = \sum_{(w,c)\in D} \sum_{w_j \in c} \log P\left(\frac{w}{w_j}\right), \tag{1}$$

where w is the target word, c means the contexts and D is the documents. If two different words have very similar contexts, it means window words are similar, such as "Kitty climbed the tree" and "Cat climbed the tree". Through the skip-gram model training, the embedding vector of 'Kitty' and 'Cat' will be very similar.

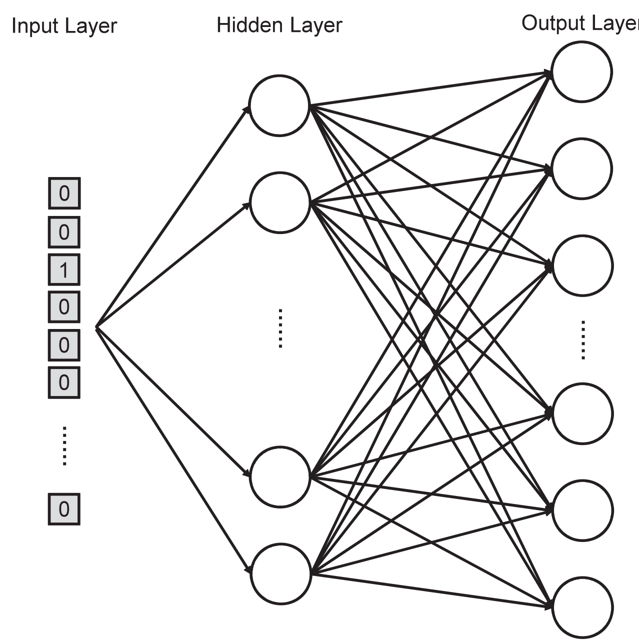


Fig. 3 The architecture of skip-gram used for word embedding

3.2 Keywords extraction

Keywords are a group of words that represent the important content of an article, which plays an important role in text clustering, classification, automatic summary etc. In addition, it also enables people to easily browse and access information. Common keywords extraction algorithms include TF-IDF algorithm [26], TextRank algorithm [27], LDA algorithm [28] and PLSA algorithm [29].

TF-IDF algorithm is a statistical-based computing algorithm, which is often used to evaluate the importance of a word to a document in a document set. The more important a word is to a document, the more likely it is to be a keyword. TF-IDF algorithm consists of two parts: TF algorithm and IDF algorithm. TF algorithm is to count the number of times a word appears in a document, the basic idea is that the more words appear in a document, the more times a word appears in a document. Then its ability to express the document will also be stronger. The calculation method of the tf value is as follows:

$$tf_{ij} = \frac{n_{ij}}{\sum_k n_{kj}}, \tag{2}$$

where n_{ij} is the frequency of the word i appears in document j.

IDF algorithm is to count how many documents a word appears in a document set. The basic idea is that if a word appears in fewer documents, its ability to distinguish documents will be stronger. The calculation method of the idf value is as follows:

$$idf_i = \log\left[\frac{|D|}{1+|D_i|}\right], \tag{3}$$

where$|D|$ is the total number of documents in the document set, and $|D_i|$ is the number of documents in which the word i appears in the document set.

Then, TF-IDF algorithm is a combination of TF algorithm and IDF algorithm. The calculation method of the tf-idf value is as follows:

$$tfidf_i = tf_{ij} \times idf_i = \frac{n_{ij}}{\sum_k n_{kj}} \times \log\left[\frac{|D|}{1+|D_i|}\right]. \tag{4}$$

As we can see, the tf-idf value is proportional to the number of times a word appears in the document, inversely proportional to the number of times the word appears in the set. The higher the tf-idf value, the more important the word.

The traditional TF-IDF algorithm does not distinguish the position of feature words in documents. In fact, the positions of feature words in the document are various, and the contribution to text category information is also different. According to the difference of category information expression ability of feature word position in text, we do different weighting processing of feature word position in text. Considering the frequency and position of words, the calculation function of candidate word weight is proposed as follows:

$$wight_i = tfidf_i + \alpha * pos_i. \tag{5}$$

In order to obtain the position information of each word, it is necessary to determine the way in which the position information is recorded and the contribution of each position in reflecting the topic of the article [30]. pos_i is 2.0 when word i appears at the beginning or end of a sentence, and 1.0 when word i appears elsewhere. If a word appears repeatedly in each position, its highest position value is selected.

After determining the formula of two characteristic items affecting the weight of words, it is necessary to consider how to determine the adjustment factors α, so that they can reflect the contribution of each factor to the weight more reasonably. The training samples are used to automatically adjust the adjustment factors. In our scheme, the adjustment factor of the training formula is studied by the Least-Mean-Square training rule. The concrete operation method is described as follows: Firstly, the value of adjustment factor is given randomly, and then the weight of each word in each text is calculated in turn, and the word set in each text is sorted from high to low according to its weight value. At the same time, a series of manual tags are needed. The training samples of text keywords, each of which describes the set of keywords in the text and the important sequences of these keywords in the text. We compare the calculated order of each word with the order of manually marked keywords, and set the sort difference:

$$diff = \sum_{j=1}^{n}(sort_{(i,j)} - sort_{(k,i,j)}). \tag{6}$$

Then, adjust the value of α by the following formula:

$$\alpha = \alpha + \eta * diff * pos. \tag{7}$$

The words are sorted according to the weight and the appropriate keyword set is extracted based on the given threshold.

3.3 Text classification

FastText is an open source text categorization tool for Facebook. Compared with other text classification models, such as SVM, LR, and neural network, fastText reduces the training time and testing time while maintaining the classification effect. As we can see in Fig. 4, fastText model inputs a sequence of words (a paragraph of text or a sentence) and outputs the probability that the sequence of words belongs to different categories.

FastText is based on the hierarchical softmax, which is an efficient approximate method. Morin and Bengio first introduced this method into the neural network language

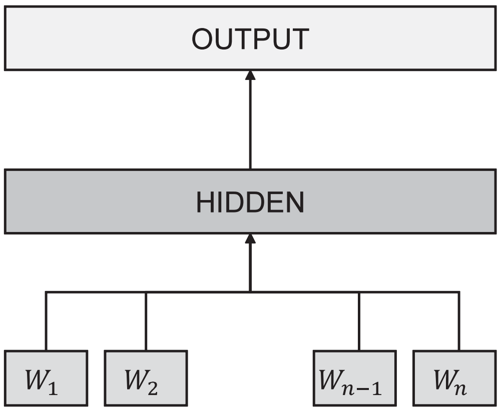


Fig. 4 The architecture of fastText used for text classification

model [31]. In order to obtain the probability distribution, this method does not need to evaluate the W output nodes in the neural network, but only needs to evaluate about $\log_2(W)$ nodes. As we can see in Fig. 5, hierarchical softmax uses a binary tree structure to represent all the words in the dictionary. We only need to follow the nodes through the bold line passes to find the corresponding words, instead of searching for each word.

The word in the input layer form a feature vector, then the feature vector is mapped to the hidden layer by a linear transformation, the hidden layer is used for solving the maximum likelihood function, then the Huffman tree is constructed according to the weight and the model parameters of each class, the Huffman tree is used as an output. The common feature is the bag-of-words model. But the bag-of-words model can not take into account the order of words, so fastText also adds n-gram features. For a set of N texts, and the optimization objective function of fastText is

$$J = -\frac{1}{N}\sum_{n=1}^{N} y_n log\left(f\left(BAx_n\right)\right), \tag{8}$$

where x_n is the standardized feature packages of text. A and B are weight matrices, y_n is the category of the nth text.

4 Experiments

4.1 Dataset

We have crawled the comments of posts from Weibo as our dataset. Weibo refers to a broadcast social media and network platform based on user relationship information sharing, transmission and acquisition, which shares short real-time information, and realizes instant information sharing, transmission and interaction through text, pictures, video and other multimedia forms. we extract 60000 comments from it, and the comments we selected are chosen randomly. The comments were manually labeled. Of the comments in our dataset, 1273 comments were identified as cyberbullying (2.13%).

4.2 Evaluation metrics

We use the hold-out method to evaluate the experiment, the held-out evaluation provides an approximate measure of precision without requiring costly human evaluation. The dataset is divided into two mutually exclusive sets, one as the training set and the other as the test set. The training set accounts for 80% of the dataset and the test set accounts for 20% of the dataset.

We select $Precision$, $Recall$ and $F1$ as a measure of performance, and the calculation formulas are

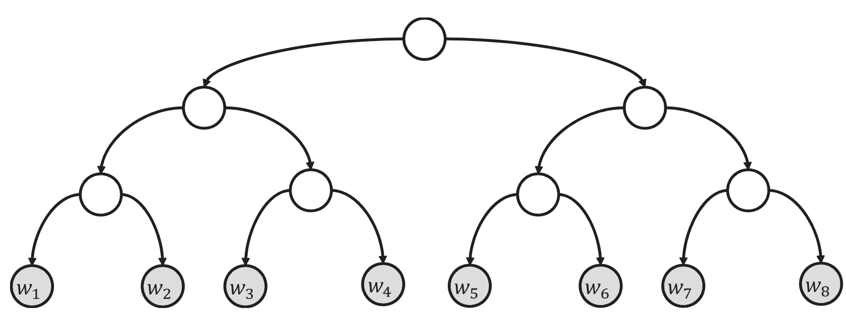
$$Precision = \frac{TP}{TP + FP}, \tag{9}$$

$$Recall = \frac{TP}{TP + FN}, \tag{10}$$

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision}, \tag{11}$$

where TF is true positive, FP is false positive, TN is true negative and FN is false negative.

Fig. 5 Hierarchical softmax uses a binary tree structure to represent all the words in the dictionary



4.3 Baselines

We adopted five state-of-the-art models as our baseline:

- **LR** is a kind of generalized linear regression analysis model. Its advantages are simple implementation, small computation, fast speed, and less storage resources. However, when data features are missing or feature space is large, the performance is not good.
- **SVM** can be applied to both linear classification and nonlinear classification. However, it is inefficient when dealing with large amounts of data. And finding the right kernel function is relatively difficult.
- **CNN** is a kind of feedforward neural network with deep structure and convolution computation. There is no need to select the features manually, so the feature classification is good. Likewise, the disadvantage is that the parameters need to be adjusted and large sample size is required.
- **RNN** is a kind of recursive neural network which takes sequence data as input, recurses in sequence evolution direction and connects all nodes by chain. It can share the statistical strength of different sequence length and different position in time. Whereas due to its chain structure, it cannot be calculated in parallel, so it has a large time cost.
- **fastText** is a text classification model based on single-layer neural network. Due to the simplicity of the model, the time cost has been greatly reduced, on the other hand the simple structure also leads to low classification accuracy.

4.4 Cyberbullying detection

4.4.1 Impact of keywords extraction

Since network structures such as CNN/RNN have the ability to automatically obtain feature expression, the improved TF-IDF algorithm is not needed for feature extraction. To demonstrate the effects of keywords extraction, we compare 3 different models (SVM, LR and fastText) in basic version, adding TF-IDF version and adding improved TF-IDF version on cyberbullying detection.

Figure 6 shows the results of different data preprocessing in LR, SVN and fastText respectively. We can observe that: (1) With the word extraction, model can perform better than their basic version. It indicates that after filtering noise data, keywords can focus on the bullying words, thus the performance can be improved. (2) Compared with the adding TF-IDF version, the adding improved TF-IDF version brings better performance. Because bullying words always appear in the begin or the end of comments,

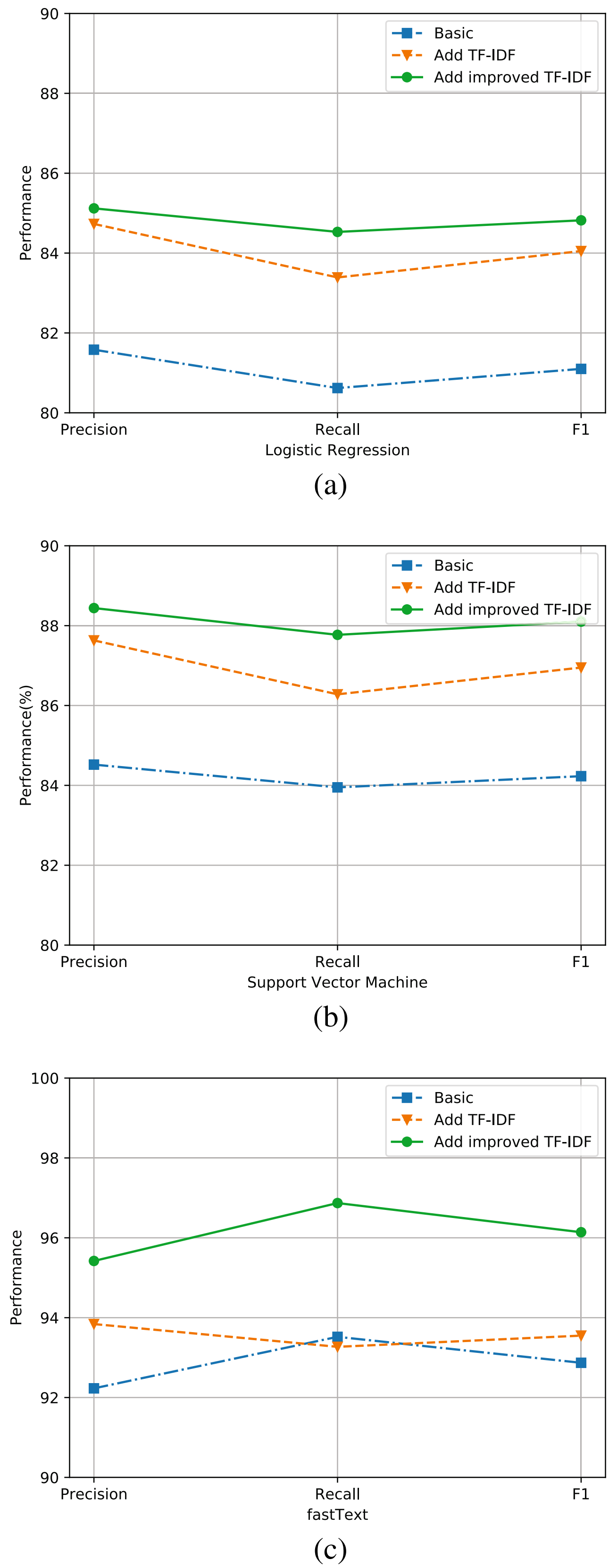


Fig. 6 Results of different data preprocessing in LR, SVN and fastText

Table 1 Result of our proposed model and baselines on cyberbullying detection

	Precision	Recall	F1
CNN	94.15%	93.36%	93.75%
RNN	93.42%	93.47%	93.44%
fastText	92.23%	93.52%	92.87%
Proposed ITFT	95.42%	96.87%	96.14%

adding position weight helps in detecting cyberbullying more efficiently.

4.4.2 Comparison with other models

We choose CNN, RNN and fastText as comparison model, and we compare the performance between the comparison model and the improved model proposed by us.

Table 1 shows the result of our proposed model and baselines on cyberbullying detection. We can see that the fastText is slightly worse than CNN and RNN, but ITFT performs best in all models. This is because many noise words are in the dataset, it may disturb the classifier's judgment, after word extraction, these data has been filtered, so ITFT can bring a better performance.

4.4.3 Time cost analysis

Graphics Processing Uni (GPU) can provide the infrastructure of multi-core parallel computing with a large number of cores, which can support the parallel computing of a large amount of data. It also has higher memory access speed than Central Processing Unit (CPU). Both CNN and RNN are trained on a NVIDIA GeForce GTX 1070 Ti, while our models are trained on a Intel Core i5-8400 CPU using 8 threads.

Table 2 shows the time cost of our proposed model and baselines on cyberbullying detection. From the table we find that: The time cost of CNN is about 30 times than the ITFT model, and RNN's time cost is about 100 times than the ITFT model, even compare with fastText, the ITFT model only need the half time to complete the training and testing, So cyberbullying can be detected efficiently through the ITFT model.

Table 2 Time cost of our proposed model and baselines on cyberbullying detection

	Training time	Testing time	Total time
CNN	62.43s	6.52s	68.95s
RNN	283.84s	26.27s	310.11s
fastText	4.95s	0.61s	5.56s
Proposed ITFT	2.45s	0.34s	2.79s

The reasons are: (1) Fasttext runs in multiple threads and is faster than a single thread. Meanwhile fastText adopts Huffman structure, which makes the running speed decrease exponentially. (2) Even though the improved TF-IDF algorithm takes up a part of the time, but after extracting keywords, the number of word in document is less than that of original document, and so the time cost has been reduced.

5 Conclusion and future works

This paper focuses on detecting cyberbullying text message in Online Social Network by using Natural Language Processing. In particular, we have proposed an improved TF-IDF based fastText (ITFT) for cyberbullying detection, which solves the problem that the basic version fastText model classification effect is reduced due to the input of the noise data. We use position weight to improve the accuracy of TF-IDF algorithm to extract keywords, and use the improved TF-IDF algorithm to extract keywords, reduces the input of noise data. At the same time, it reduces the amount of text data, so the time required for text classification is reduced. The experimental results show that compared with the basic version fastText model, neural network model and traditional machine learning model, the ITFT model achieves the best performance in accuracy and efficiency. However, due to the lack of continuity between words in the extracted keywords, it is hard to use the n-gram feature, and it causes the loss of the sequence between the words. How to keep the sequence between the words in keywords extraction is the direction of our future research.

References

1. Slonje R, Smith PK (2008) Cyberbullying: Another main type of bullying? Scand J Psychol 49(2):147–154
2. Hinduja S, Patchin JW (2010) Bullying, cyberbullying, and suicide. Arch Suicide Res 14(3):206–221
3. Patchin JW (2006) Bullies move beyond the schoolyard: A preliminary look at cyberbullying. Youth Viol Juvenile Just 4(2):148–169
4. Smith PK, Mahdavi J, Carvalho M, Fisher S, Tippett N (2010) Cyberbullying: Its nature and impact in secondary school pupils. J Child Psychol Psych 49(4):376–385
5. Schmidt A, Wiegand M (2017) A survey on hate speech detection using natural language processing. In: Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, SocialNLP@EACL 2017, Valencia, Spain, pp 1–10

6. Kalchbrenner N, Grefenstette E, Blunsom P (2014) A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014. Long Papers Baltimore, Vol 1, pp 655–665
7. Tang D, Qin B, Liu T (2015) Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, pp 1422–1432
8. Joulin A, Grave E, Bojanowski P, Mikolov T (2016) Bag of tricks for efficient text classification. arXiv:1607.01759
9. Salawu S, He Y, Lumsden J (2017) Approaches to automated detection of cyberbullying: A survey. IEEE Transactions on Affective Computing
10. Nandhini B, Sheeba J (2015) Cyberbullying detection and classification using information retrieval algorithm. In: Proceedings of the 2015 International Conference on Advanced Research in Computer Science Engineering and Technology (ICARCSET 2015). ACM, pp 20
11. Squicciarini AC, Rajtmajer SM, Liu Y, Griffin C (2015) Identification and characterization of cyberbullying dynamics in an online social network. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2015, Paris, France, pp 280–285
12. Chavan VS, Shylaja SS (2015) Machine learning approach for detection of cyber-aggressive comments by peers on social media network. In: 2015 International Conference on Advances in Computing, Communications and Informatics, ICACCI 2015, Kochi, India, pp 2354–2358
13. Fahrnberger G, Nayak D, Martha VS, Ramaswamy S (2014) Safechat: a tool to shield children's communication from explicit messages. In: International conference on innovations for community services
14. Pérez PJC, Valdez CJL, Ortiz MdGC, Barrera JPS, Pérez PF (2012) Misaac: Instant messaging tool for ciberbullying detection. In: Proceedings on the International Conference on Artificial Intelligence (ICAI). The Steering Committee of The World Congress in Computer Science, Computer, pp 1
15. Serra S, Venter HS (2011) Mobile cyber-bullying: A proposal for a pre-emptive approach to risk mitigation by employing digital forensic readiness. In: Information Security South Africa Conference 2011, Hyatt Regency Hotel, Rosebank, Johannesburg, South Africa, Proceedings ISSA 2011
16. Chen Y, Zhou Y, Zhu S, Xu H (2012) Detecting offensive language in social media to protect adolescent online safety. In: 2012 International Conference on Privacy, Security, Risk and Trust, PASSAT 2012, and 2012 International Confernece on Social Computing, SocialCom 2012, Amsterdam, Netherlands, pp 71–80
17. Bretschneider U, Wöhner T, Peters R (2014) Detecting online harassment in social networks. In: Proceedings of the International Conference on Information Systems - Building a Better World through Information Systems, ICIS 2014, Auckland, New Zealand
18. Agrawal S, Awekar A (2018) Deep Learning for Detecting Cyberbullying Across Multiple Social Media Platforms. In: European Conference on Information Retrieval. Springer, pp 141–153
19. Al-Ajlan MA, Ykhlef M (2018) Optimized Twitter Cyberbullying Detection based on Deep Learning. In: 2018 21st Saudi Computer Society National Computer Conference (NCC). IEEE, pp 1–5
20. Dadvar M, Trieschnigg RB, de Jong F (2013) Expert knowledge for automatic detection of bullies in social networks. In: 25th Benelux Conference on Artificial Intelligence, BNAIC 2013. TU Delft, pp 57–64
21. Dadvar M, Trieschnigg D, de Jong F (2014) Experts and machines against bullies: A hybrid approach to detect cyberbullies. In: Advances in Artificial Intelligence - 27th Canadian Conference on Artificial Intelligence, Canadian AI 2014. Proceedings, Montréal, pp 275–281
22. Silva C, Santos R, Barbosa R (2018) Detection and Prevention of Bullying on Online Social Networks: The Combination of Textual, Visual and Cognitive. In: International Conference on Intelligent Technologies for Interactive Entertainment. Springer, pp 95–104
23. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. Computer Science
24. Bengio Y, Ducharme R, Vincent P, Janvin C (2003) A neural probabilistic language model. J Mach Learn Res 3:1137–1155
25. Collobert R, Weston J, Bottou L, Karlen M, Kavukcuoglu K, Kuksa P (2011) Natural language processing (almost) from scratch. J Mach Learn Res 12(1):2493–2537
26. Wu HC, Luk RWP, Wong K, Kwok K (2008) Interpreting TF-IDF, term weights as making relevance decisions. ACM Trans. Inf. Syst. 26(3):13:1–13:
27. Mihalcea R, Tarau P (2004) Textrank: Bringing order into text. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, Barcelona, Spain, pp 404–411
28. Tasci S, Gungor T (2009) Lda-based keyword selection in text categorization. In: The 24th International Symposium on Computer and Information Sciences, ISCIS 2009, North Cyprus, pp 230–235
29. Bosch A, Zisserman A, Muñoz X (2006) Scene classification via plsa. In: Computer vision - ECCV 2006, 9th european conference on computer vision. Proceedings, Graz, Part IV, pp 517–530
30. Chen J, Chen C, Liang Y (2016) Optimized tf-idf algorithm with the adaptive weight of position of word. In: 2016 2Nd international conference on artificial intelligence and industrial engineering (AIIE 2016). Atlantis Press
31. Davidian M (2002) Hierarchical linear models: Applications and data analysis methods. Publ Amer Stat Assoc 98(463):767–768

Jiale Wu received the Bachelor's degree in Electric Power Engineering from Shanghai University of Electric Power, China, in 2015, and the master degree in Department of Computer Science and Technology from Shanghai University of Electric Power, China, in 2020. His research interest includes information security, Natural Language Processing and deep learning.

Mi Wen received the M.S. degree in Computer Science from University of Electronic Science and Technology of China in 2005 and the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China in 2008. She is currently an Associate Professor of the College of Computer Science and Technology, Shanghai University of Electric Power. From May 2012 to May 2013, she was a visiting scholar at University of Waterloo, Canada. She serves Associate Editor of Peer-to Peer Networking and Applications (Springer). She keeps acting as the TPC member of some flagship conferences such as IEEE INFOCOM, IEEE ICC, IEEE GLOEBECOM, etc from 2012. Her research interests include privacy preserving in wireless sensor network, smart grid etc.

Rongxing Lu has been an assistant professor at the Faculty of Computer Science (FCS), University of New Brunswick (UNB), Canada, since August 2016. Before that, he worked as an assistant professor at the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU), Singapore from April 2013 to August 2016. Rongxing Lu worked as a Postdoctoral Fellow at the University of Waterloo from May 2012 to April 2013. He was awarded the most prestigious Governor Generals Gold Medal, when he received his PhD degree from the Department of Electrical & Computer Engineering, University of Waterloo, Canada, in 2012; and won the 8th IEEE Communications Society (ComSoc) Asia Pacific (AP) Outstanding Young Researcher Award, in 2013. He is presently a senior member of IEEE Communications Society. His research interests include applied cryptography, privacy enhancing technologies, and IoT-Big Data security and privacy. He has published extensively in his areas of expertise, and was the recipient of 8 best (student) paper awards from some reputable journals and conferences. Currently, Dr. Lu currently serves as the Vice-Chair (Publication) of IEEE ComSoc CIS-TC (Communications and Information Security Technical Committee). Dr. Lu is the Winner of 2016-17 Excellence in Teaching Award, FCS, UNB.

Beibei Li received his B.E. degree (awarded Outstanding Graduate) in communication engineering from Beijing University of Posts and Telecommunications, China, in 2014 and his Ph.D. degree (awarded Full Research Scholarship) from the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, in 2019. He was invited as a visiting researcher at the Faculty of Computer Science, University of New Brunswick, Canada, from March to August, 2018, as well as the research group of NEtworked Sensing and Control (NESC), College of Control Science and Engineering, Zhejiang University, China, from February to April, 2019. Dr. Li is currently an associate professor at the College of Cybersecurity, Sichuan University, China. His research interests span several areas in cyber-physical system security, with a focus on intrusion detection techniques, applied cryptography, and big data privacy in smart grids and industrial control systems. He served as a TPC member for several international conferences, including IEEE GLOBECOM, WCSP, and ICNC, etc. His research studies have been published in IEEE Trans. on Information Forensics and Security, IEEE Trans. on Industrial Informatics, ACM Trans. on Cyber-Physical Systems, IEEE Internet of Things J., Information Sciences, IEEE GLOBECOM, and IEEE ICC, etc.

Jinguo Li currently works at the College of Computer Science and Technology, Shanghai University of Electric Power. Jinguo does research in Information Science, Data Structures and Computer Security and Reliability. Their most recent publication is 'Secure, flexible and high-efficient similarity search over encrypted data in multiple clouds'.