Detection of offending text for cryptic metaphors and sensitive references

Guanghui Chang, Ronghui Zhang*, Jiahui Luo

School of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

*S211201035@stu.cqupt.edu.cn

Abstract. Text data, as the main carrier of information dissemination, is filled with some offending and harmful content. Due to the development of time and culture, the offending content tends to use obscure language forms when expressing. In addition the compliance information that references sensitive keywords also greatly increases the detection complexity. This makes traditional text detection methods face great challenges. To solve the above problems, we constructed a detection dataset containing three offending categories. A detection method based on Natural Language Processing (NLP) technology and two detection strategies are also designed, and trained and compared on various types of advanced neural network models. The experimental results show that the obscure features and deep semantics can be obtained through learning, and also prove the effectiveness of our method.

Keywords: Offending Text Detection • NLP • Neural Networks

1 INTRODUCTION

At present, the volume of global data is growing rapidly at an annual growth rate of over 59 per cent, with unstructured data such as text accounting for 80 per cent of the total. And text, as the main carrier of information dissemination, contains some illegal and harmful contents, such as rumours, fraudulent information, political tendency and violence and terrorism. In addition, with the rapid development of artificial intelligence technology, a large amount of text data automatically generated by large language models further floods human society, exacerbating the possibility of the emergence of harmful and offending content. In this zeitgeist, it becomes particularly important to provide efficient, accurate and intelligent methods for detecting offending texts. However, with the continuous evolution of network culture, the expressions of offending texts are also changing and upgrading. In order to avoid systematic detection, the offending texts are more and more inclined to use more obscure language forms such as metaphors and puns when expressing themselves. In addition, compliance information that references sensitive keywords also increases the difficulty of detection, such as negative news reports, sex knowledge popularisation, and the publication of criminal methods. These

A. Haldorai et al. (eds.), *Proceedings of the 2024 3rd International Conference on Artificial Intelligence, Internet and Digital Economy (ICAID 2024)*, Atlantis Highlights in Intelligent Systems 11,
https://doi.org/10.2991/978-94-6463-490-7_59

compliance messages are superficially similar to non-compliance messages and thus are prone to misinterpretation. To address these challenges, we construct a special offending text detection dataset and combine it with NLP-related techniques for model training, so as to improve the understanding and recognition of obscure and complex semantic features. The contributions of this paper are as follows:

(1) Based on real Internet text data, a training dataset containing three violation categories, namely pornography, gambling and politics, is constructed, and its data contains a large number of obscure language forms and sensitive references.

(2) Two different violation information detection strategies are explored and their advantages as well as differences are analysed for academic research as well as engineering applications respectively.

(3) The advantages and disadvantages of the currently commonly used neural network architectures are analysed and compared, providing a reference for their application in offending text detection.

2 RELATED WORK

Offending text is a broad concept that includes, but is not limited to, a wide range of content such as hate speech, political bias, etc. Davidson et al [1] consider hate speech to be offensive speech directed at a protected identity group. Pavlopoulos et al [2] define toxicity as a general term covering offensive, abusive, and hateful language. In the early days, offending text detection mainly relied on keyword-based methods to extract features and combined with machine learning classifiers such as logistic regression, fuzzy methods and decision trees. With the development of deep learning techniques, researchers have begun to shift to the use of word embedding-based feature extraction methods and have achieved excellent results. However, due to the complexity of the task itself, it still faces various challenges such as data construction and robustness.

In the data construction for offending text detection, there are two main types of biases: labelling bias and lexical bias. Labelling bias stems from people's subjectivity in determining whether a text is offending or not. yin et al [3] pointed out that non-expert annotators are more inclined to mislabel content as hate speech. hartvigsen et al [4] started to use large pre-trained language models to generate the training data, but generative diversity and data distribution are still potential problems. Lexical bias manifests itself in the form of models that often lead to misclassification due to the excessive frequency of certain words or phrases. Dixon et al [5] found that for identity terms such as gender and ethnicity are unevenly distributed across different lengths of text, and therefore used length-sensitive up-sampling to balance the dataset.Badjatiya et al [6] mitigated bias-sensitive words in the task by identifying and replacing them in the model. stereotype bias. Zhou et al [7] utilised two proposed data filtering methods to obtain the training sample set. In addition, some studies remove bias from detection models at the modelling level by adding importance weights [8] and multi-task learning [9].

In addition to data bias, lack of robustness is a common problem in offending text detection [10-11].Ilan et al [12] augmented an offending speech dataset with real data

collected by an online platform to improve generalisability. Arango et al [13] also pointed out that the high accuracy of the existing studies may be overestimated due to data overfitting and user distribution bias, highlighting the need to pay more attention to the data's generalisation ability and the effect of user distribution. Wullach et al [14] proposed a method for generating large amounts of synthetic hate speech data using pre-trained language models. Ludwig et al [15] investigated the generalisation ability of deep learning models on different target groups of offending speech and evaluated the effectiveness of three unsupervised domain adaptation strategies.

3 PROPOSED METHODOLOGY

3.1 Data Set Construction

In order to solve the two major problems in real detection scenarios, there are two major problems: obscure language forms are difficult to discriminate and compliance information that references sensitive information is prone to lead to misjudgement. We start from the data level and construct a specialised dataset for supervised training to address the above problems. The composition of the dataset is shown in Fig. 1.

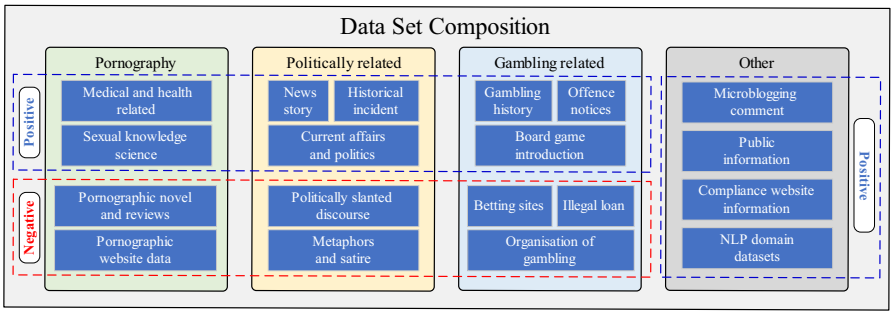

Fig. 1. Components of the data set

We selected three more common violation categories, namely pornography, gambling and politics, to construct the dataset. Negative samples are collected from the offending links and websites, and the relevant accesses have been reported with the relevant departments and permission has been granted. It is worth noting that we selected special positive samples. For the pornography category, we collected from health websites and sex knowledge science websites, and for the gambling category, we collected from news reports on gambling-related offences, board game introductions, and the history of gambling. These positive samples contain offending keywords, but the overall semantics do not affect people negatively. For the category of political involvement, due to the presence of more obscure forms of expressions such as metaphors, puns and antithesis in the negative sample examples. Therefore, in the construction of positive samples, we collected some historical events, current affairs and their related

comments. In order to adapt to the real detection, we also constructed a separate "other" category with all positive samples to improve the generalisation of the detection.

The acquired data were first pre-processed, which included removing meaningless characters and codes, intercepting excessively long data, and so on. Then a manual review was performed, which included correcting mislabelling, removing ambiguous data. Each piece of data contained a category label and an violation label.

3.2 Detection Strategy

Considering that offending text detection is different from simple tag categorisation, fine-grained sentiment and semantic changes also need to be considered. Therefore we divide this detection task into two phases, domain classification and violation classification. Domain classification is used to distinguish the categories of the text to be detected in order to determine the offending domain of the offending text. Violation classification, on the other hand, is used to identify semantic changes and distinguish between the presence of implicit violations and sensitive references.

Initially, we used three separate models for violation classification as a separated detection strategy because we were not sure whether there was similarity in the violation features of the three violation categories. However, considering the practicality and efficiency of detection, it is not practical to train a separate model for each violation category in real detection, so we use "pornographic", "gambling" and "political" as our strategy. Therefore, we merge the positive samples from "pornography", "gambling" and "politics" into the "other" category, and train the violation classification at the same time of domain classification as a fusion detection strategy.

3.3 Training Design

Although the preprocessing step solves the phenomenon of uneven distribution of some data lengths. However, since the actual detection requires batch processing of data, there are still cases where the length exceeds the limit. And the direct truncation will lead to the phenomenon of semantic fragmentation. For this problem we first slice the text $S = (w_1, w_2, \ldots, w_n)$ by the maximum input length, i.e. $S = \mathrm{S}_{1,i} + \mathrm{S}_{i,j} + \cdots + \mathrm{S}_{q,n}$, where $\mathrm{S}_{i,j}$ represents the text segment consisting of the ith character w_i to the jth character w_j in the text. $\mathrm{S}_{i,j}$ are then fed into the model to obtain the respective embedding vectors $a_1, a_2, \ldots, a_n$, where $a_i \in \mathbb{R}^{p \times d}$, p is the number of characters, d is the length of the embedding vectors, and n is the number of cuts in the text segment.

$$w_i = \frac{exp(s_i)}{\sum_{j=1}^{n} exp(s_j)} \tag{1}$$

The a_i are then converted to the same length and dimension by a linear transformation and attention scores are computed using for each converted vector a_i'.In turn, the function Softamx is used for s_i as shown above to assign an attentional weight w_i to each a_i. finally the weighted sum is computed using the attentional weights to obtain the final fused vectors, the computational formula is shown below:

$$a_{attention} = \sum_{i=1}^{n} w_i \cdot a_i \tag{2}$$

In classification tasks, fully determined category labels tend to lead to overconfidence in the model, resulting in overfitting. Therefore we use label smoothing technique to improve the generalisation ability and robustness of the model by adjusting the distribution of true labels. That is, a smoothing factor ϵ is set, and for K categorical categories, the labels after using label smoothing are shown in the following equation:

$$y_i' = \begin{cases} 1 - \epsilon + \frac{\epsilon}{K} & if\ i = j \\ \frac{\epsilon}{K} & else \end{cases} \tag{3}$$

where y_i' is the target distribution after label smoothing, and ϵ is a decimal number between 0 and 1 indicating the degree of smoothing. In the case of using label smoothing, the cross-entropy loss function is similarly computed for the adjusted label y_i', as shown below, where is the loss value computed by $\mathcal{L}$ for a single sample and p_i is the probability distribution predicted by the model.

$$\mathcal{L} = -\sum_{i=1}^{K} y_i' \, log(p_i) \tag{4}$$

4 EXPERIMENTS

4.1 Data Sets and Parameters

The experiments are conducted entirely using the constructed dataset, and the relevant information of the dataset is shown in Table 1.

Table 1. Information on the number of data sets

Category	Length	Positive	Negative	Totals
Pornography	902	4934	4154	9088
Gambling	1218	2469	2731	5200
Politics	1285	4421	4918	9939
Other	928	11281	0	11281

It can be seen that we constructed about 35,000 training data, and the number of positive and negative samples is basically balanced for the three violation categories.We use three metrics, Precision (P), Recall (R) and F1 score (F1), for evaluation, and the ratio of 8:1:1 is used for the division of the training, validation and test sets, with the number of training rounds set to 5, the batch size to 32, the learning rate to 0.001, and the maximum input length set to 500.

4.2 Experimental Results and Analysis

The models chosen for the experiments include three main categories, the traditional machine learning model SVM, the classical neural network architectures including

fully connected layers, CNNs, RNNs, and attention mechanisms, and the pre-trained language model BERT.

Table 2. Experimental results for the three categories of violations detected

Category	Positive			Negative		
	P	R	F1 %	P	R	F1 %
Pornography	92.35	95.14	93.72	94.31	91.02	92.64
Gambling	92.86	88.22	90.48	89.25	93.35	91.25
Politics	88.72	81.25	84.82	84.33	90.26	87.19

Table 2 demonstrates the detection results of the three offending categories, and the higher F1 level proves that the cryptic features, as well as the deep semantic features of the sensitive references, are learnable. This particular feature can be migrated to other detection models. Meanwhile, since the detection accuracy of the political category is at a lower level among the three, it is evident that the capture of obscure features needs to be further improved. While the pornography category has the highest detection accuracy because the keyword features are too significant.

Table 3. Comparison of detection results for each model using a split detection strategy

Models	Classification of Fields			Categories of Violation		
	P	R	F1 %	P	R	F1 %
SVM	96.56	97.47	97.01	75.64	73.31	74.45
FC	96.71	95.92	96.28	91.53	91.19	91.36
CNN	97.81	97.54	97.67	93.46	93.00	93.23
RNN	94.89	94.29	94.57	89.62	88.91	89.26
RNN+CNN	97.64	97.23	97.42	93.60	93.25	93.42
RNN+Attention	96.99	96.66	96.82	91.88	91.66	91.77
Transformer	92.72	93.48	93.07	81.73	81.23	81.48
Bert	98.79	99.11	98.95	96.46	95.89	96.17

Table 3 shows the comparison of the results of each model using the separated detection strategy. It can be seen that domain classification relies more on keyword features, so the SVM model using word frequency as a feature also has a good classification effect, but when facing the classification of violations that contain special linguistic forms such as cryptic metaphors, sensitive references, and fine-grained semantic features, BERT, which has been pre-trained on a large scale and has a larger number of parameters, still maintains the best detection effect. Table 4 demonstrates the comparison of the results of each model using the fusion detection strategy, and it can be seen that there is no significant decrease in the detection accuracy after fusion, and therefore the same features exist in the three offending categories.

From the model comparison, it can be seen that the RNN is not as effective as models such as FC and CNN that capture local features, due to the excessive length of some of

the data, which may cause the RNN to have the problem of disappearing or exploding gradients. The RNN+Attention strengthens the model's ability to focus on the key information by the Attention mechanism, but there is only a weak performance improvement. In addition, the Transformer architecture may need more careful tuning in this task. The pre-trained BERT model still achieves the best results.

Table 4. Comparison of detection results of each model using fusion detection strategy

Models	P	R	F1 %
SVM	76.15	74.39	75.26
FC	93.92	93.97	93.92
CNN	94.29	94.30	94.26
RNN	90.96	91.23	91.08
RNN+CNN	94.85	93.82	94.29
RNN+Attention	94.05	93.24	93.59
Transformer	86.68	88.95	87.32
Bert	96.18	96.46	96.32

5 CONCLUSIONS

In this paper, we start from two major problems: the use of implicit metaphors and other linguistic forms in offending text detection is difficult to discriminate, and references to compliance information of sensitive content are prone to misjudgement. A specialised dataset is constructed at the data level to address these two challenges, and a detection method as well as two detection strategies are designed based on NLP-related techniques. The experimental results show that obscure features and deep semantic features can be obtained through data learning, while the proposed detection method has good detection performance. In addition, this paper also compares different model architectures, analyses their advantages and disadvantages in the detection domain and special feature learning, and provides a reference for offending text detection.

In the future, we plan to further explore the robustness improvement of the detection model and focus on the detection methods based on less sample learning.

REFERENCES

1. Davidson, T., Warmsley, D., Macy, M., Weber, I. (2017). Automated hate speech detection and the problem of offensive language. In: Proceedings of the international AAAI conference on web and social media. Montreal. pp. 512-515. https://doi.org/10.1609/icwsm.v11i1.14955
2. Pavlopoulos, J., Sorensen, J., Dixon, L., Thain, N., Androutsopoulos, I. (2020). Toxicity Detection: Does Context Really Matter?. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online. pp. 4296-4305. https://doi.org/10.48550/arXiv.2006.00998

3. Yin, W., Zubiaga, A. (2021). Towards generalisable hate speech detection: a review on obstacles and solutions. PeerJ Computer Science, 7, e598. https://doi.org/10.48550/arXiv.2102.08886
4. Hartvigsen, T., Gabriel, S., Palangi, H., Sap, M., Ray, D., Kamar, E. (2022). ToxiGen: A Large-Scale Machine-Generated Dataset for Adversarial and Implicit Hate Speech Detection. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics. Dublin. pp. 3309-3326. https://doi.org/10.48550/arXiv.2203.09509
5. Dixon, L., Li, J., Sorensen, J., Thain, N., Vasserman, L. (2018). Measuring and mitigating unintended bias in text classification. In: Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. New York.pp. 67-73. https://doi.org/10.1145/3278721.3278729
6. Badjatiya, P., Gupta, M., Varma, V. (2019). Stereotypical bias removal for hate speech detection task using knowledge-based generalizations. In: The world wide web conference. New York. pp. 49-59. https://doi.org/10.1145/3308558.3313504
7. Zhou, X., Sap, M., Swayamdipta, S., Choi, Y., Smith, N. A. (2021). Challenges in Automated Debiasing for Toxic Language Detection. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume. Online. pp. 3143-3155. https://doi.org/10.18653/v1/2021.eacl-main.274
8. Kennedy, B., Jin, X., Davani, A. M., Dehghani, M., Ren, X. (2020). Contextualizing Hate Speech Classifiers with Post-hoc Explanation. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Online pp. 5435-5442. https://doi.org/10.18653/v1/2020.acl-main.483
9. Vaidya, A., Mai, F., Ning, Y. (2020). Empirical analysis of multi-task learning for reducing identity bias in toxic comment detection. In: Proceedings of the International AAAI Conference on Web and Social Media. Limassol. pp. 683-693. https://doi.org/10.1609/icwsm.v14i1.7334
10. Markov, T., Zhang, C., et al. (2023). A holistic approach to undesired content detection in the real world. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vancouver. pp. 15009-15018. https://doi.org/10.1609/aaai.v37i12.26752
11. Garg, T., Masud, S., Suresh, T., Chakraborty, T. (2023). Handling bias in toxic speech detection: A survey. ACM Computing Surveys, 55(13s), 1-32. https://doi.org/10.1145/3580494
12. Ilan, T., Vilenchik, D. (2022). HARALD: Augmenting Hate Speech Data Sets with Real Data. In: Findings of the Association for Computational Linguistics: EMNLP 2022. Abu Dhabi. pp. 2241-2248. https://doi.org/10.18653/v1/2022.findings-emnlp.165
13. Arango, A., Pérez, J., Poblete, B. (2019). Hate speech detection is not as easy as you may think: A closer look at model validation. In: Proceedings of the 42nd international acm sigir conference on research and development in information retrieval. New York. pp. 45-54. https://doi.org/10.1145/3331184.3331262
14. Wullach, T., Adler, A., Minkov, E. (2021). Fight Fire with Fire: Fine-tuning Hate Detectors using Large Samples of Generated Hate Speech. In: Findings of the Association for Computational Linguistics: EMNLP 2021. Punta Cana. pp. 4699-4705. https://doi.org/10.18653/v1/2021.findings-emnlp.402
15. Ludwig, F., Dolos, K., Zesch, T., Hobley, E. (2022). Improving generalization of hate speech detection systems to novel target groups via domain adaptation. In: Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH). Seattle. pp. 29-39. https://doi.org/10.18653/v1/2022.woah-1.4