A Bilingual Multi-type Spam Detection Model Based on M-BERT

Jie Cao*, Chengzhe Lai†
School of Cyberspace Security, Xi'an University of Posts and Telecommunications, Xi'an, China
Email: *Jaycaoxupt@gmail.com, †lcz_xupt@163.com

GLOBECOM 2020 - 2020 IEEE Global Communications Conference | 978-1-7281-8298-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/GLOBECOM42002.2020.9347970

***Abstract*—Spam has harassed Internet users for a long time, and how to detect spam accurately and efficiently is a critical problem. As yet, there are lots of research works proposed to detect spam, e.g., black and white lists, machine learning methods, and deep learning content-level measures, etc. Based on previous works, we find that most of methods' accuracy can reach 0.95 when they focus on one type and one language spam. Nevertheless, nowadays, people will receive spam messages of different types, different sources, and even different languages. Toward this, we develop a novel model, which is based on Google multilingual bidirectional encoder representations from transformers (M-BERT). Meanwhile, we design a brand new bilingual multi-type spam dataset to train our model. Particularly, we utilize optical character recognition (OCR) to extract text from image-based spam. Through the experiment, we find that the proposed model's accuracy can reach 0.9648, which outperforms the comparison models. In terms of time overhead, the proposed model only costs 0.3168 seconds per training step, which is an acceptable overhead. Therefore, these analysis results demonstrate that our approach can detect bilingual multi-type spam effectively.**

***Index Terms*—Spam detection, natural language processing, deep learning, BERT**

I. Introduction

Due to the blossom of the traditional Internet and mobile Internet, the amount of Internet users in China has increased rapidly in the past 10 years. According to the "44th Statistical Report on the Development of the Internet in China", until June 2019, the amount of Chinese Internet users has reached 854 million. This rapid development also provides convenience for information dissemination, and the channel of information dissemination include social media, email, Wechat platform, and other applications. In the meantime, these channels also attract malicious people to disseminate spam messages because of their vast amount of users. The contents of these spam mostly consist of merchandise advertisements, and unsolicited push information of platforms registered by users, meanwhile, there will be potential attackers who will add phishing websites in the messages to scam users [1]. Based on a previous report in 2010 [2], 8% of 25 million uniform resource locators (URLs) of Twitter direct to phishing websites and scammers, which are listed on popular blacklists. This report announced that the click rate of Twitter spam reaches 0.13% and 0.0003%˜0.0006% in email spam. In fact, since both of them have large user bases, we must consider their impact at the same time. These spam's flood harass people's life, occupy network bandwidth and computer performance, seriously disrupt people's work, and threaten people's property safety. This evidence suggests that an accurate spam detection technique is extremely desirable.

There are two types of spam, i.e., text-based and image-based spam. Before 2006, most of the spam was text-based, but after that, a large number of image-based spam appeared. Spammers embed spam text into the image and escape from the detection of text filter. Also, according to language, spam can be divided into Chinese spam, English spam, and other languages. Therefore, how to deal with these spam is a challenging issue.

In essence, the detection of spam is a binary classification. To solve this problem, researchers have designed many useful methods. At present, there are mainly two kinds of detection directions. The first one is to seek out the spammers and deleting their accounts [3], [4]. However, when spammers find that their accounts have been blocked, they will create new accounts and continue to disseminate spam. Therefore, another robust direction is content-level spam detection [5]–[8], which focuses on spam's content. Because content-level spam detection does not need to analyze user's features, it can find spam on the network more timely and comprehensively.

Given the superiority of content-level spam detection, we propose a novel content-level spam detection model, which is based on M-BERT [9], fine-tunes it with a brand new dataset, and adopts original data preprocessing method. Ultimately, our model achieves excellent performance in detecting bilingual multi-type spam.

The main contributions of this paper are as follows:

Firstly, we introduce a new spam detection scenario named bilingual multi-type, and design a detection model based on M-BERT. Secondly, we use the OCR technique to extract the text in spam images and merge them with text-based spam to build a dataset. Finally, through experiments, we find that the detection accuracy of the model after training can reach 0.9648, which is significantly higher than that of the existing models.

II. Related Work

In this section, we describe the related work in two areas: text-based spam detection and image-based spam detection.

A. Text-based Spam Detection

One of the schemes for spam filtering is blocking the spammer. Lee *et al.* [3] proposed a method to find spammer

978-1-7281-8298-8/20/$31.00 ©2020 IEEE

on Twitter, the authors analyzed spammers' behavior over time and summarize common characteristics to mine spammer. Shen *et al.* [4] established a generalized spammer detection model by jointing multiple features such as URL, user tags from real-world Twitter datasets to seek out spammer.

Some classical machine learning algorithms are content-level spam detection methods. The support vector machine (SVM) is a typical classification algorithm, which has been used in spam detection [5]. Sahami *et al.* applied Naive Bayes to the field of email spam detection [6]. Nevertheless, the detection effect of the above two machine learning methods is mediocre. Carreras *et al.* [7] used the boosting tree to filter spam email, and this method's performance exceeds the Naive Bayes.

It is believed that deep learning is the most universal and accurate method to detect spam currently, and researchers have also proposed many approaches based on deep learning to detect spam. Jain *et al.* [10] proposed a framework called sequential stacked CNN-LSTM model (SSCL) for spam detection, which combined the convolutional neural network (CNN) and long short-term memory (LSTM). In this model, the text was transformed into a vector through word2vec [11], and this process also used semantic dictionaries to enhance model performance. They use CNN to extract sentence features and use LSTM to obtain sentence sequence information. When the model's drop out rate is equal to 0.5, the accuracy in the SMS spam dataset is 99.01% and in the Twitter dataset is 93.71%. Madisetty *et al.* [12] proposed an ensemble method based on deep learning and traditional feature-based model. They made use of five CNNs with different embedding methods (Glove [13], word2vec), and one feature-based model used n-gram features, user features, and content features. This model integrates the advantages of spammer features and content-level detection, finally, its recall in the HSpam dataset reached 95.0%.

B. Image-based Spam Detection

Compared with text-based spam, the research of image-based spam started relatively late, because there are three main difficulties for detecting image-based spam: (1) Lacking image-based spam datasets; (2) The embedded image text is distorted to resist the filtering of the detector maliciously; (3) The processing of images for computers is complicated.

For image-based spam, researchers have proposed two detection approaches: (1) Based on the text embedded in the image; (2) Based on the features of the image itself.

In [14], the author used the OCR technique to extract the text in the image. Then the text was classified by the text-based approaches mentioned earlier.

The researchers also found that the spam image is different from the normal image, so a detection method based on the features of the image itself is proposed. Aradhye *et al.* [15] found that the spam image has larger text area, less color saturation and heterogeneity of non-text area than normal image. Wu *et al.* [16] found the proportion of text area and the existence of the banner slogan have differences between normal image and spam image. According to differences, the above authors all trained the SVM classifier to detect spam.

Despite there are adequate literatures about detecting text-based spam and image-based spam. However, the previous researches always focused on one type, one source, one language spam, and they did not consider them as an ensemble. Therefore, there is a demand to establish a model, which can adapt to more challenging and complex circumstances, to detect bilingual multi-type spam.

III. METHODOLOGY

Our novel bilingual multi-type model based on M-BERT is presented detailedly in this section.

A. Basic BERT Model

Text classification is an essential part of the natural language processing (NLP) task, and the emergence of BERT [9] has brought a considerable breakthrough in NLP. Devlin *et al.* proposed BERT in 2018, and the framework of BERT is shown in Fig. 1. BERT uses bidirectional transformer encoders, which employ a multi-head attention mechanism, to get contextual information. Through unsupervised deep bidirectional pre-training, BERT can represent different contextual meanings of the same word excellently, but the previous model, like word2vec, cannot realize this function.

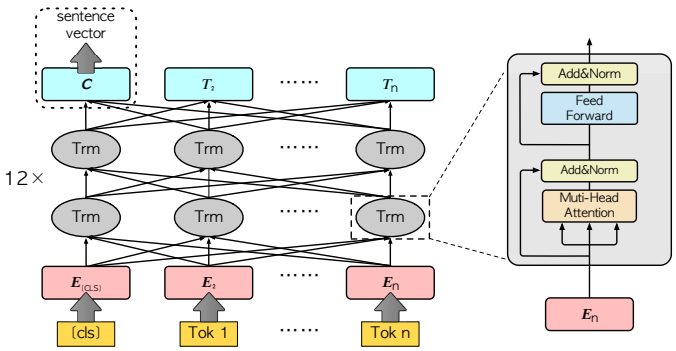

Fig. 1. The framework of BERT to do text classification.

The two main steps of BERT are as follows.

(1) Input processing

Before embedding, a [CLS] tag is added at the beginning of the sentence to indicate that this is a start, and a [SEP] tag is added to manifest the boundary of two sentences. During the embedding process, the sentence goes through three embedding layers. The token embedding finishes word segmentation and encodes word via vocab file. The segment embedding helps BERT to differentiate two sentences in a sentence pair. The position embedding helps BERT to get the sequence information of word. When we add the above three results, we obtain the final input embedding.

(2) Pre-training strategies

- Masked LM: This Strategy is a cloze test that the BERT model use [mask] tag to mask 15% words in the training dataset randomly. However, there are few [mask] tags

appearing in the fine-tuning process, so in 80% of cases, the [mask] tag will be selected to replace the selected word; in 10% of cases, the model replaces selected words with random words; in the remaining 10% of cases, the model keeps the sentence unchanged. The mission of training is to predict the masked word precisely.

- Next Sentence Prediction: For QA tasks, the model needs to understand the relationship between sentences. Thus the second strategy is to determine whether two sentences can be connected semantically.

After passing the BERT, a sentence will be encoded into a sentence vector, which is represented by the tag [CLS]'s vector actually, which is shown in Fig. 1. We notate this 1×768 dimensions vector as $\boldsymbol{C}$. $\boldsymbol{C}$ will pass the softmax classifier to get the predicted probability p_i. We will calculate the cross-entropy loss via p_i and real label y_i, and the calculation is shown in formula (1), (2). The fine-tuning process ought to train parameters of BERT and 768×2 dimensions parameter matrix $\boldsymbol{W}$ of the softmax layer by minimizing the loss.

$$p_i = softmax(\boldsymbol{CW}) \qquad p_i \in [0, 1] \tag{1}$$

$$Loss = -y_i \log p_i + (1 - y_i) \log(1 - p_i) \quad y_i \in \{0, 1\} \tag{2}$$

B. Proposed Model Based on M-BERT

M-BERT(base) has 12 transformer layers, 12 attention heads, 104 languages, and 110 million parameters. M-BERT is the most crucial part of the model proposed in this paper. Our model is shown in Fig. 2. It consists of the following two parts:

(1) Data preprocessing: For image-based spam, we use OCR to extract embedded text. Combining image text and original text-based spam, we achieve the bilingual multi-type dataset. The specific operation of data preprocessing will be described in Section IV.

(2) Fine-tuning: We fine-tune M-BERT through the training dataset and get the model, which is suitable for our spam detection task.

IV. EXPERIMENT DESIGN

This section presents the design of dataset and comparison models, and we also describe the metrics selection and hardware configuration.

A. Bilingual Multi-type Dataset

The bilingual multi-type spam dataset used in this paper consists of six public datasets, which are shown in Table 1. Fig. 3 and Fig. 4 show some examples of the dataset we built.

During the data preprocessing, we use OCR to extract text from the image-based dataset. The extracted text combining with text-based spam is cleaned by regular expression to remove special symbols that M-BERT cannot recognize. In addition to data cleaning, we also introduce the following three operations:

(1) No-text image padding: During the OCR extraction process, we find that there are some non-spam images without embedding any text. Therefore, in the experiment, we use the

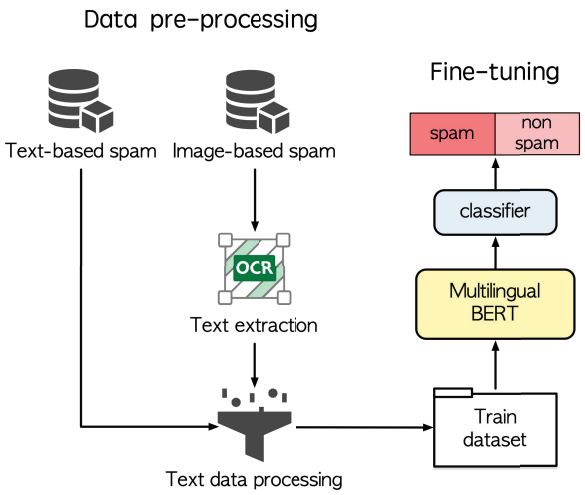

Fig. 2. A bilingual multi-type spam detection model.

Text	label
贵公司负责人(经理/财务)您好：我公司是深圳市东讯实业有限公司，我公司实力雄厚，	1
联普翻译公司是中国译协及日本商工会员单位。犊有联普、韦伯斯特、羊城翻译三大品	1
第一次在水木发文，在家版，这个我潜水最多的地方，心情真的很不好，很想倾诉一下	0
置顶#代理须知#拒绝泛滥，我不会每天一直更新图片的，图片有专门的地方，为了不影	1
爆美来袭实拍颜色赞不赞超美喔质量保证绝对一手货源本人招收代理#西游记吉尼斯纪	1
回想起我们的整个青春，最让人想哭的事情莫过于这三件：看周星驰的电影，听周杰伦	0
Subject: enron methanol meter : this is a follow up to the note i gave you on monday ,	0
Subject: vocable brand new stock for your attentionvocalscape inc the stock symbol is :	1
I actually look cute today	0
I want to have positive mind, flawless skin, and nice personality.	0
If you see, Tsung's flaming skull flies into a group including Tanya	0
THEY MAKE A CAMPAIGN PROMISE https://t.co/QaV9Uno6BQ	1
LOL this shit never gets old	0
:. Hahahahah	0
Hey guys Check this out: Kollektivet Don't be slappin' my penis I think that they deserv	1
Hey everyone, I am a new channel and will post videos of book reviews and music on th	1

Fig. 3. Text-based spam examples.

sentence "words result num = 0." as the text of this part of the non-spam images, to prevent the empty sequence from inputting into the model.

(2) Segment and splicing: Limited by GPU performance and experiment duration, we control the text length to less than 250 characters, so we need to segment long text. In this experiment, for the texts longer than 250 characters, the first 126 characters and the last 124 characters are selected for

Fig. 4. Chinese image-based spam expmple(right), English image-based spam example(left).

TABLE I
DATASET COMPONENT

Dataset Type	Used quantity
Chinese text-based email spam [17]	5000
Chinese text-based social media spam from sina weibo [18]	1300
Chinese image-based spam [19]	3000
English text-based email spam [20]	5000
English text-based social media spam from Twitter and YouTube review [21], [22]	6800
English image-based spam [23]	3000

TABLE II
POSITIVE AND NEGATIVE SAMPLE SIZE

	Spam	Non-spam
Train	7942	6338
Dev	2634	2126
Test	2611	2149

splicing, so as to help the model to obtain the information of the front and back context;

(3) Dataset division: We divide the preprocessed dataset into a train, dev, and test files according to the ratio of $3:1:1$. The "Train" file is used to fine-tune M-BERT; the "Dev" file is used to optimize model parameters; the "Test" file is used to evaluate model performance. The amount of the positive and negative samples of the above files are shown in Table II.

B. Contrastive Model

In this part, we describe two models for comparing with our model.

1) TextCNN: CNN has made significant achievements in the field of computer vision, and the reason why CNN has superior image classification ability is that it can generalize and extract features of the image. Therefore, it was also applied to text classification, and Kim *et al.* [24] proposed the TextCNN model to complete text classification.

We choose TextCNN to detect bilingual multi-type spam, and the parameters of it are shown in Fig. 5. Towards to TextCNN, we need to conduct the following steps:

- Firstly, we should apply word embedding, in the experiment, the embedding dimensions are 128, and we get the $n \times 128$ sentence matrix (n is the sentence length);
- Then we use 3 sizes $(2, 4, 6)$ filters to do convolution, and each size has 128 filters. We get 384 feature maps;
- These feature maps pass max-pooling layers, and we merge the maximal value of each feature map;
- The merged vector is calculated by a fully connected layer and softmax function; we choose cross-entropy as loss. In the end, we get the predicted probability of the input sentence.

2) BERT-CNN: According to Section III, we know that M-BERT is a more accurate language representation model than word2vec. Therefore, we can utilize M-BERT to produce token-level vectors and use these vectors as word embedding to train the TextCNN model. This BERT-CNN model leverages

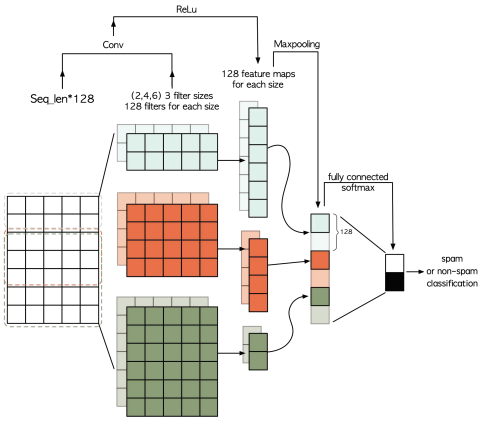

Fig. 5. Using TextCNN to detect spam.

both CNN's excellent feature extraction capability and BERT's superior language representation capability.

The structure of BERT-CNN is shown in Fig. 6. The TextCNN part is unchanged, and the main change is that we use the output of M-BERT's last layer transformer encoder to be the input of TextCNN. As shown in Fig. 1, the output of M-BERT is $O = \{C, T_1, T_2, \ldots, T_n\}$, and $T = \{T_1, T_2, \ldots, T_n\}$ will be the input of TextCNN. Therefore, the TextCNN's input dimension is $n \times 768$.

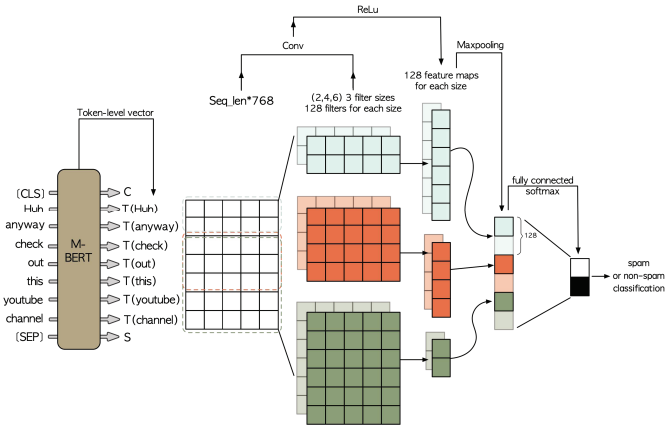

Fig. 6. Using BERT-CNN to detect spam.

C. Evaluation Metrics

We use cloud GPU to train and test the model, and the configuration is shown in Table III.

TABLE III
HARDWARE CONFIGURATION

Hardware	Settings
Operation system	Ubuntu 18.04.4 LTS
Deep learning framework	Tensorflow 1.14
CPU	Intel(R) Xeon(R) Gold 5218
RAM	64-core 2.3 GHz
GPU	GeForceRTX2080Ti×2

TABLE IV
CONFUSION MATRIX

	Spam	Non-spam
Predicted as spam	TP(True Positive)	FP(False Positive)
Predicted as non-spam	FN(False Negative)	TN(True Negative)

To evaluate the models' performance, we use recall, precision, accuracy, and F1 score to measure the pros and cons of models. Table IV shows the distribution of spam judgments, and the formulas (3)-(6) give the calculation method of metrics.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

$$F1 = \frac{2TP}{2TP + FN + FP} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{6}$$

V. RESULTS

In the end, we obtain the value of metrics of our model, which are shown in Table V. These metrics are all above 0.96, so we convince that the model proposed in this paper can filter spam efficiently.

A. Comparison and Analysis

The training loss and accuracy of TextCNN are shown in Fig. 7, and the testing loss and accuracy are shown in Fig. 8. According to these two figures, we find that the training accuracy is infinitely close to 1 after 2500 steps, but the testing accuracy is stable around 0.91 after 2000 steps. The testing accuracy of TextCNN is 5.68% lower than that of our model. From the above, the TextCNN's testing accuracy is 0.09 lower than its training accuracy. This gap shows that the TextCNN exists overfitting, and this problem can also be proved by the phenomenon in Fig. 8 that the testing loss decreases before 2000 steps, whereas increases after 2000 steps. We have already implemented the dropout to avoid this problem, so we have to believe that TextCNN is flawed.

However, observing our model's training and testing accuracy and loss in Fig. 9, the training and testing accuracy are very close, and the testing loss fluctuation is at a stable low value. Thus it can be inferred that our model does not have the overfitting problem.

TABLE V
OUR MODEL'S METRICS VALUES

Metrics	Value
Recall	0.9648832
Precision	0.9641181
Accuracy	0.9646465
F1	0.9607115

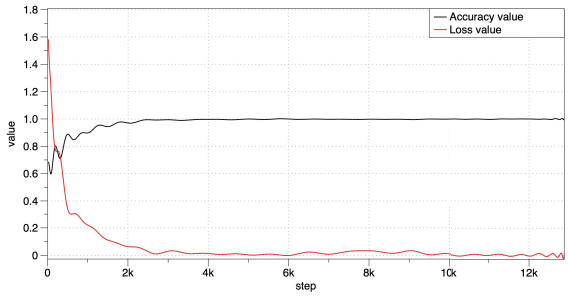

Fig. 7. TextCNN training loss and accuracy. (Polynomial function fitting)

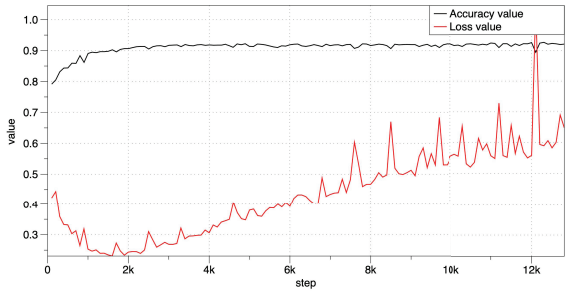

Fig. 8. TextCNN testing loss and accuracy. (Polynomial function fitting)

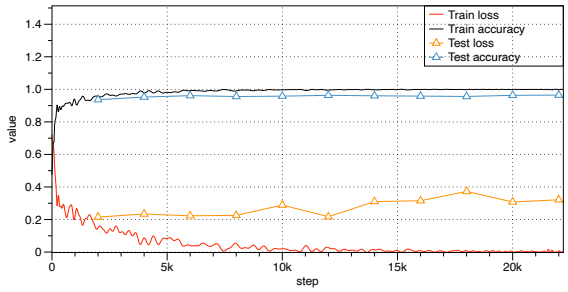

Fig. 9. BERT training and testing loss and accuracy.

The comparison of our model with the BERT-CNN model in four metrics is shown in Fig. 10, and the value of four metrics of BERT-CNN are all lower than our model obviously. It can prove that using BERT's token-level vector as the input to TextCNN does not have a positive impact on spam detection.

In time consumption, the TextCNN costs 0.2185 seconds per training step, our model costs 0.3168 seconds per training step,

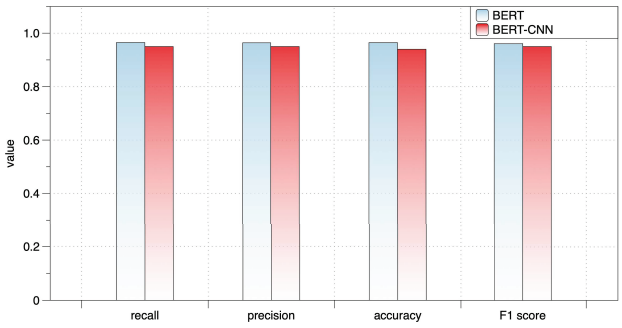

Fig. 10. Comparison of the four metrics of our model and the BERT-CNN.

and BERT-CNN costs 2.5167 seconds per training step. This order corresponds with the complexity of these three models.

By comparison, we convince that our model takes into account both remarkable spam detection capability and relatively low time consumption, so it can undertake the bilingual multi-type spam detection task. Nonetheless, we cannot deny that it still exists the following flaws:

(1) For non-spam images without text, we cannot extract text from them by OCR, so we propose using "words result num = 0." as the text context of these images. The premise of doing this is that we find that all the image-based spam in our dataset contains text information. However, in real life, many spam images have no textual information, such as pornographic and violent images. Therefore, our model needs to update to deal with no-text image-based spam.

(2) The English image-based spam dataset is severely outdated, so that our model may not handle the latest English image-based spam effectively. Besides, the proportion of image-based spam dataset is relatively small, so later we ought to expand its size.

VI. CONCLUSION

In this paper, we focus on the detection of bilingual multi-type spam and propose a model based on M-BERT to solve this problem. By introducing M-BERT, we solve the bilingual problem of spam, and by the OCR technique, we can extract text from image-based spam to help us deal with the multi-type problem of spam. Besides, in order to show the performance of our proposed model, we design TextCNN and BERT-CNN model to be comparison. Finally, the experiment results demonstrate that the value of metrics of our model, which can reach 0.96, are the highest. In terms of time overhead, our model consumes 0.3168 seconds per training step, and this value slightly more than TextCNN but much less than BERT-CNN. As a consequence, our model can restrain bilingual multi-type spam disseminating effectively. In future work, we plan to expand image-based spam dataset and compress the size of our model further to enhance the accuracy and running speed.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China under 2017YFB0802002, and the Innovation Ability Support Program in the Shaanxi Province of China under 2017KJXX-47.

REFERENCES

[1] K. S. Adewole, N. B. Anuar, A. Kamsin, K. D. Varathan, and S. A. Razak, "Malicious accounts: Dark of the social networks," *Journal of Network & Computer Applications*, vol. 79, no. 1, pp. 41–67, 2017.
[2] C. Grier, K. Thomas, V. Paxson, and C. M. Zhang, "@spam: the underground on 140 characters or less." 2010.
[3] K. Lee, B. D. Eoff, and J. Caverlee, "Seven months with the devils: A long-term study of content polluters on twitter," in *Fifth international AAAI conference on weblogs and social media*, 2011.
[4] H. Shen, F. Ma, X. Zhang, L. Zong, X. Liu, and W. Liang, "Discovering social spammers from multiple views," *Neurocomputing*, vol. 225, pp. 49–57, 2017.
[5] H. Drucker, D. Wu, and V. N. Vapnik, "Support vector machines for spam categorization," *IEEE Transactions on Neural networks*, vol. 10, no. 5, pp. 1048–1054, 1999.
[6] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A bayesian approach to filtering junk e-mail," in *Learning for Text Categorization: Papers from the 1998 workshop*, vol. 62. Madison, Wisconsin, 1998, pp. 98–105.
[7] X. Carreras and L. Marquez, "Boosting trees for anti-spam email filtering," *arXiv preprint cs/0109015*, 2001.
[8] J. Martinez-Romo and L. Araujo, "Detecting malicious tweets in trending topics using a statistical analysis of language," *Expert Systems with Applications*, vol. 40, no. 8, pp. 2992–3000, 2013.
[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding."
[10] G. Jain, M. Sharma, and B. Agarwal, "Spam detection in social media using convolutional and long short term memory neural network," *Annals of Mathematics and Artificial Intelligence*, vol. 85, no. 1, pp. 21–44, 2019.
[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.
[12] S. Madisetty and M. S. Desarkar, "A neural network-based ensemble approach for spam detection in twitter," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 4, pp. 973–984, 2018.
[13] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
[14] G. Fumera, I. Pillai, and F. Roli, "Spam filtering based on the analysis of text information embedded into images," *Journal of Machine Learning Research*, vol. 7, no. Dec, pp. 2699–2720, 2006.
[15] H. B. Aradhye, G. K. Myers, and J. A. Herson, "Image analysis for efficient categorization of image-based spam e-mail," in *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*. IEEE, 2005, pp. 914–918.
[16] C. Wu, K. T. Cheng, Q. Zhu, and Y. Wu, "Using visual features for anti-spam filtering," 2005.
[17] https://plg.uwaterloo.ca/ğvcormac/treccorpus06/, 2006, [Online].
[18] https://archive.ics.uci.edu/ml/datasets/microblogPCU, 2015, [Online].
[19] https://tianchi.aliyun.com/competition/entrance/231685/information, 2018, [Online].
[20] https://www.kaggle.com/venky73/spam-mails-dataset, 2018, [Online].
[21] T. C. Alberto, J. V. Lochter, and T. A. Almeida, "Tubespam: Comment spam filtering on youtube," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2015, pp. 138–143.
[22] T. A. Almeida, T. P. Silva, I. Santos, and J. M. G. Hidalgo, "Text normalization and semantic indexing to enhance instant messaging and sms spam filtering," *Knowledge-Based Systems*, vol. 108, pp. 25–32, 2016.
[23] M. Dredze, R. Gevaryahu, and A. Elias-Bachrach, "Learning fast classifiers for image spam." in *CEAS*, 2007, pp. 2007–487.
[24] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.