Booter Blacklist Generation Based on Content Characteristics

Wang Zhang[1,2], Xu Bai[1,2], Chanjuan Chen[3], and Zhaolin Chen[4](✉)

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
zhangwang@iie.ac.cn

[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

[3] China National Machinery Industry Corporation, Beijing, China

[4] Nanjing University of Aeronautics and Astronautics, Nanjing, China
zhaolin_in_chen@hotmail.com

Abstract. Distributed Denial of Service (DDoS) attacks-as-a-service, known as Booter or Stresser, is convenient and low-priced for ordinary people to launch DDoS attacks. It makes DDoS attacks even more rampant. However, until now there is not much research on Booter and little acquaintance with their backend infrastructure, customers, business, etc. In this paper, we present a new method which focuses on the content (text) characteristics on Booters websites and selects more discriminative features between Booter and non-Booter to identify Booters more effectively in the Internet. The experimental results show that the classification accuracy of distinguishing Booter and non-Booter websites is 98.74%. In addition, our method is compared with several representative methods and the results show that the proposed method outperforms the classical methods in 66% of the classification cases on three datasets: Booter websites, 20-Newsgroups and WebKB.

Keywords: Booter service · Feature selection · Text classification

1 Introduction

Distributed Denial of Service (DDoS) attacks, which create a huge volume of illegitimate traffic to jam the network and interrupt the network resource, is one of the biggest menaces for network security. DDoS attacks have existed for many years and continuously grown in both frequency and power. In 2014, CloudFlare reported a 400 Gbps NTP amplification attack on one of their customers [10]. Recently, Arbor Networks reported a 1.7 Tbps memcached amplification attack on an unnamed customer of a US-based service provider [4]. It is believed that Booters account for a large portion of the attack traffic in such mega attacks in recent years [3].

Activity of DDoS-as-a-service or DDoS-for-hire websites, also called Booter or Stresser, is not an accident. According to [12], it is the fact that (1) booters

H. Gao et al. (Eds.): CollaborateCom 2018, LNICST 268, pp. 529–542, 2019.
https://doi.org/10.1007/978-3-030-12981-1_37

provide a friendly interface and remove the need of technical skills to perform attacks, (2) booters are public in the Internet and easy to find by using Google or Bing and (3) they usually offer very affordable prices due to fierce commercial competition. Thus, Booters are also considered to be the indication of new period of the DDoS attack evolution. Despite the serious threat of Booters to the Internet, until now there is not much research on Booter and we know little about the ecosystem of these Booter services. Prior work points out that Booter blacklist generation is a promising approach to mitigate the challenge of the Booter services and show the effectiveness of the blacklists [15]. The prior work developed a Booter blacklist generation system containing three components: The crawler firstly collects suspect Booter URLs in the Internet; Secondly, the scraper acquires the suspect Booter URL information based on fifteen proposed characteristics; Finally, the classifier identifies whether a suspect URL is a Booter website on account of the scraped URL information. We observe that Booter websites often use similar content (text) in their webpages and we consider that content (text) characteristics on Booters websites are also effective to identify Booters. Therefore, we present a new method which classifies Booters based on content (text) characteristics. Also, we propose a new feature selection algorithm to improve the performance of text classification. Our main contributions are listed as follows:

- We develop a new Booter classifier based on content (text) characteristics which enrich the methods of identifying Booters.
- We propose a feature selection algorithm, which selects more discriminative features with the minimal number, to improve the performance of text classification.

The rest of this paper is organized as follows. The related work is discussed in Sect. 2. The details of our approach is described in Sect. 3. The experimental results and discussion are presented in Sect. 4. Finally, the paper is concluded in Sect. 5.

2 Related Work

Santanna et al. [15] designed a methodology for Booter blacklist generation and demonstrated the value of the Booter blacklist. Until now, their methodology has already found 519 Booters [13], which is of great benefits to the mitigation of Booter services. Karami et al. [5] investigated underlying technical and business structure of Booter services from the leaked data of three major booters and the payment obstruction to their services in cooperation with PayPal. Krämer et al. [6] designed a novel honeypot that can simulate amplifiers and be of assistance to monitor amplification DDoS attacks. Due to the important location of amplifiers, many methods of the mitigation of amplification DDoS attacks can be explored based on the honeypot amplifiers. Krupp et al. [7] developed methods to uncover the infrastructures behind amplification DDoS attacks by using fingerprint to the scanners and TTL-based trilateration techniques, which is also beneficial to the detection of back-end infrastructures of Booter services. Krupp et al. [8] construct a novel method that can attribute DDoS attacks to the honeypot operators including Booter services based on their honeypot amplifiers. Noroozian et al. [9] analysed the data captured from their honeypot amplifiers and provided us an in-depth investigation and explanation of victimization patterns, which is of assistance to understand the ecosystem of commoditized DDoS attacks. Santanna et al. [14] subscribed DDoS attacks from fourteen Booters to capture the real attack data and performed an analysis of attack characteristics of fourteen Booter services. The above works are very insightful and significant, but we also need more novel and in-depth research about the mitigation of Booter services or amplification DDoS attacks.

3 The Proposed Approach

We firstly describe the overall structure of our Booter list generation system in Sect. 3.1. Then, we describe the details of our feature selection method in Sect. 3.2.

3.1 Booter List Generation System

Our system contains two components (see Fig. 1): a crawler and a classifier. The crawler collects the suspect Booter URLs and related webpages. The classifier identifies whether a suspect URL is a Booter website based on the content (text) characteristics of the webpage. The crawler firstly collects the suspect Booter URLs from Google search engine by using relevant keywords. The total number of suspect Booter URLs is 718, which contain 51 Booter URLs. Then, the crawler acquires webpages based on the suspect Booter URLs. Sometimes, the webpage of a URL is missing, in this case, the crawler acquires webpages from web cache, which is always provided by search engines. The classifier contains three steps: feature preprocessing, feature selection and classification. In the step of feature preprocessing, we extract the content (text), remove stop words and use bag-of-words model to preprocess the above webpages. However, the feature vector of every document (webpage) using the bag-of-words model is sparse and high dimensional. Therefore, feature selection is a very important step for text classification and it ensures that the features which are most relevant to particular class labels can be picked out for model training. In the step of classification, we use Linear Support Vector Machine (LSVM) and Multinomial Naïve Bayes (MNB), which are efficient classifiers in text categorization, to classify Booters.

3.2 Feature Selection Method

Our method is a filter-based feature selection method, which just relies on the properties of the data and independent of any classification algorithm. There are some commonly used feature selection methods such as Information Gain [11], improved Gini Index [16] and Chi-square [19]. Information Gain and Chi-square

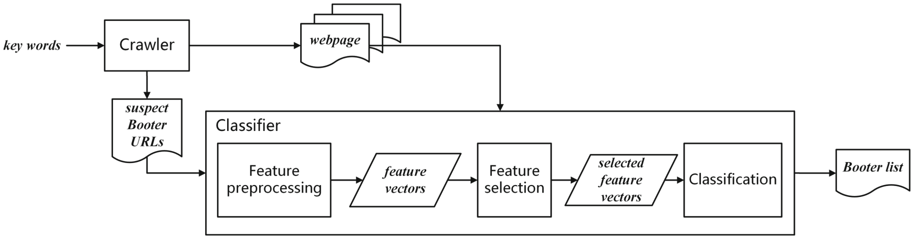


Fig. 1. The overview of Booter list generation system

are the two most effective feature selection methods [19]. Improved Gini index is an improved feature selection based on Gini index and it is reported that improved Gini Index perform more effective than Information Gain and Chi-square [16]. The feature selection methods usually consider the probability of class c_i when term t_k is present or absent, and select the representative terms of a class. However, they may ignore the differences in the distribution of different categories on a feature (term). Inspired by [17], we use centroid u_{ik} and standard deviation sd_{ik} as the representative of the distribution of class c_i on a feature (term) t_k. Mathematically, we define global inter-category distance as:

$$GD_k = \frac{1}{\sum_{i=1}^{N} sd_{ik}} \sum_{i=1}^{N} \sum_{j=i+1}^{N} |u_{ik} - u_{jk}| \tag{1}$$

where N is the number of categories. We now present the algorithm based on global inter-category distance as follows (see Algorithm 1).

Algorithm 1. Global inter-category distance algorithm

```
Input: $D$ - the preprocessed data set, $K$ - the requested number of
       features
Output: $S$ - the selected feature subset
foreach class $c_i$ do
    foreach term $t_k$ do
        obtains the centroid $u_{ik}$ and standard deviation $sd_{ik}$ of class $c_i$ ;
    end
end
foreach term $t_k$ do
    calculates the $GD_k$ of term $t_k$ by using Equation (1);
end
arranges all terms in descending order based on their $GD_k$;
selects top-$K$ terms into $S$;
return $S$;
```

Global inter-category distance algorithm firstly obtains centroid u_{ik} and standard deviation sd_{ik} for each class, then calculates the sum of the distance between different category pairs (Eq. (1)), and finally selects top-K features (terms) based on the score of our metric method. However, the above method may have a problem that sometimes a class is lack of their representative features and is difficult for the classifier to distinguish it. The global inter-category distance method may neglect the distance between a specific category and others, and cause an imbalance problem in text categorization [18]. Thus, it is necessary to ensure the balance of representative features for each class. To solve this problem, We define type-based inter-category distance as:

$$TD_{ik} = \frac{1}{\sum_{i=1}^{N} sd_{ik}} \sum_{j=1}^{N} |u_{ik} - u_{jk}| \tag{2}$$

For each class c_i, we calculate the distance TD_{ik} between this class and others, then average the requested number of features to each class to ensure the balance of their representative features. It ensures that every class obtains equal and enough representative features. We now prestent the feature selection algorithm based on type-based inter-category distance as follows (see Algorithm 2).

Algorithm 2. Type-based inter-category distance algorithm

```
Input: $D$ - the preprocessed data set, $K$ - the requested number of
       features
Output: $S$ - the selected feature subset
averages the requested number of features and sets the selected number of
each class $c_i$ as $n_i$;
foreach class $c_i$ do
    foreach term $t_k$ do
        obtains the centroid $u_{ik}$ and standard deviation $sd_{ik}$ of class $c_i$ ;
    end
end
foreach class $c_i$ do
    foreach term $t_k$ do
        calculates the $TD_{ik}$ of term $t_k$ by using Equation (2);
    end
    arranges all terms in descending order based on their $TD_{ik}$;
    selects top-$n_i$ terms into $S$;
end
return $S$;
```

The feature selection algorithm based on type-based inter-category distance ensures the balance of representative features for each class. However, we also want to pick out the features that are discriminative for all of the categories besides selecting the balanced and representative features for each class. Thus, we combine global distance with type-based distance, and define combined inter-category distance as:

$$CD_{ik} = \frac{2GD_k}{N(N-1)} + \frac{TD_{ik}}{N-1} \tag{3}$$

We replace the Eq. (2) in Algorithm 2 with Eq. (3) to get another feature selection algorithm and compare these three methods in Chap. 4.

4 Experiments

In this Section, we use three datasets to fully verify the presented feature selection algorithm, and show the experiment results on Booter websites, 20-Newsgroups and WebKB datasets in Sects. 4.1, 4.2 and 4.3, respectively. Finally, we discuss the above experiments in Sect. 4.4.

4.1 Booter Websites

The total number of the collected suspect Booter URLs is 718, which contain 51 Booter URLs. We also acquired webpages based on the suspect Booter URLs. We extracted the content (text), removed stop words and used bag-of-words model to preprocess the webpages. After that, the dimension of the features is 66448. This dataset is small, thus, we adopted LeaveOneOut in this experiment. According to [15], we define classification accuracy metrics as following:

- True positive (T_P): The number of Booter websites are correctly classified as Booter
- True negative (T_N): The number of non-Booter websites are correctly classified as non-Booter
- False positive (F_P): The number of non-Booter websites are incorrectly classified as Booter
- False negative (F_N): The number of Booter websites are incorrectly classified as non-Booter.

$$CAR = \frac{T_P + T_N}{n} \tag{4}$$

$$FP_{er} = \frac{F_P}{n} \tag{5}$$

$$FN_{er} = \frac{F_N}{n} \tag{6}$$

Where n is the total number of the collected suspect Booter websites. CAR is classification accuracy rate, FP_{er} is false positive error rate, FN_{er} is false negative error rate. In order to evaluate the performance of the proposed method, we used Linear Support Vector Machine (LSVM) and Multinomial Naïve Bayes (MNB), which are efficient classifiers in text categorization.

The performance curves of LSVM classifier are drawn in Fig. 2. We compare global distance method with Information Gain, improved Gini Index and Chi-square. This data set has only two categories, and it's no need to use type-based distance method or combined distance method, which are suitable for multi-category tasks. Figure 2(a) shows that the CAR performance of using global distance method is superior to other feature selection methods when the number of selected features is 50 and greater than 1000. It acquires the highest value, 97.91%, when the number of selected features is 5000. Figure 2(b) indicates that FP_{er} of using global distance method is less than other feature selection methods when the number of selected features is 50 and greater than 1000. It reaches 0.69%, the lowest value, when the number of selected features is 5000. Figure 2(c) shows FN_{er} performance based on global distance method is less than other feature selection methods when the number of selected features is greater than 800. It reaches 1.25%, the lowest value, when the number of selected features is 6000. Thus, the experiments show that global distance method using LSVM classifier produces highest CAR values in 7 out of 12 cases, lowest FP_{er} values in 7 out of 12 cases, and lowest FN_{er} values in 7 out of 12 cases.

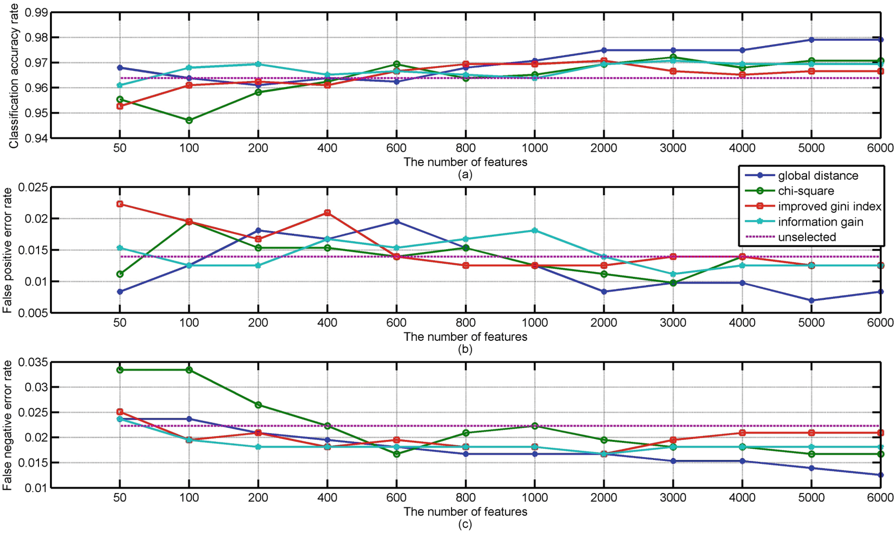


Fig. 2. The performance curves of LSVM classifier on Booter websites. (a) The curves of classification accuracy rate; (b) The curves of false positive error rate; (c) The curves of false negative error rate

The performance curves of MNB classifier are drawn in Fig. 3. Figure 3(a) shows that the CAR performance of using global distance method is superior to other feature selection methods except when the number of selected features is 6000. It acquires the highest value, 98.74%, when the number of selected features is 800. Figure 3(b) indicates that FP_{er} of using global distance method is less than other feature selection methods in all cases. It reaches 0.0%, the lowest value, when the number of selected features is from 600 to 6000. Figure 3(c) shows FN_{er} performance based on global distance method is less than or equal to other feature selection methods when the number of selected features is 400, 600, 800, 3000, 4000. It reaches 1.25%, the lowest value, when the number of selected features is 600. Thus, the experiments show that global distance method using MNB classifier produces highest CAR values in 11 out of 12 cases, lowest FP_{er} values in 12 out of 12 cases, and lowest FN_{er} values in 5 out of 12 cases.

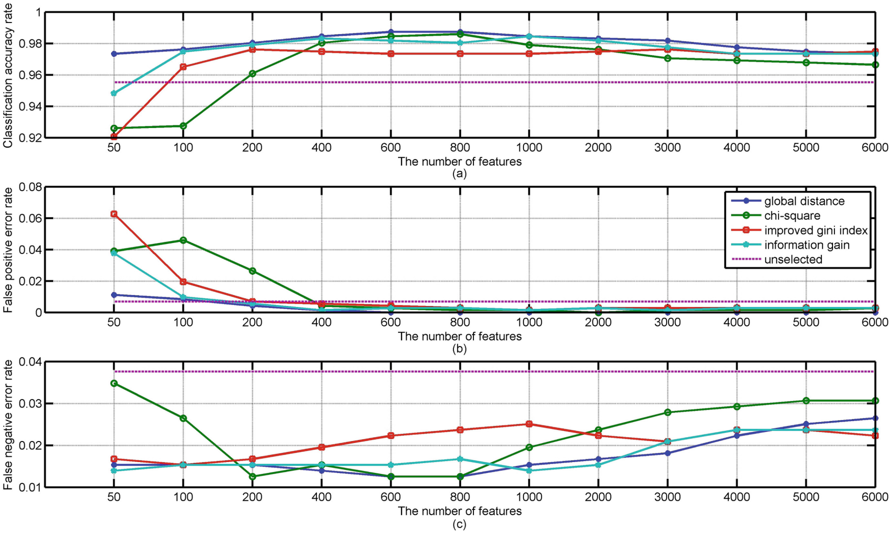
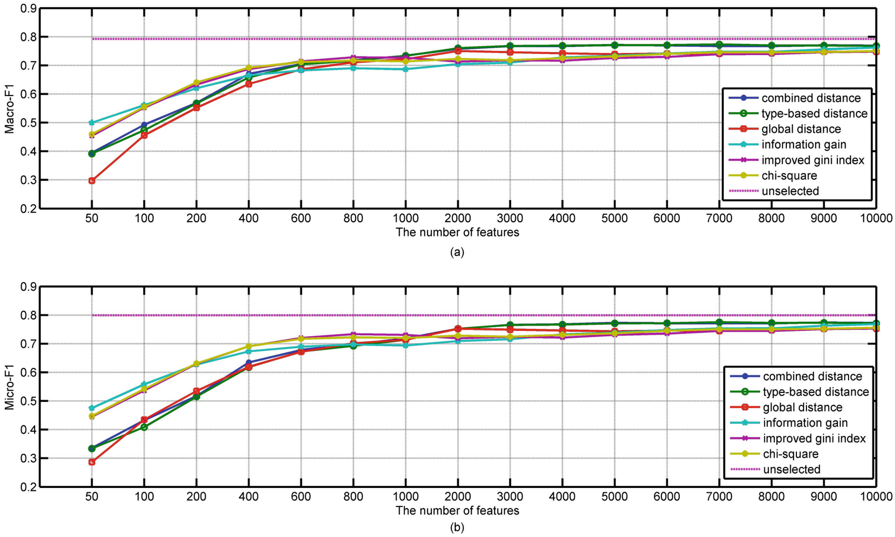


Fig. 3. The performance curves of MNB classifier on Booter websites. (a) The curves of classification accuracy rate; (b) The curves of false positive error rate; (c) The curves of false negative error rate

4.2 20-Newsgroups

The 20-Newsgroups [2] dataset collects about 20,000 newsgroup documents and is evenly divided into 20 different categories. It is a popular data set for experiments in text categorization. In this experiment, we used bydate version of the data set, which contains 18846 documents and is sorted by date into training (60%) and test (40%) sets. We removed stop words and used bag-of-words model to preprocess the data set. After that, the dimension of the features is 129326. Macro-F1 and Micro-F1 were used to evaluate the performance of different feature selection methods.

The performance curves of LSVM classifier on 20-Newsgroups are drawn in Fig. 4. Figure 4(a) shows that the macro-F1 performance of using combined distance method is superior to the three classical feature selection methods when the number of selected features is greater than 1000. Among the three distance methods, combined distance method and type-based distance method are always superior to global distance method. Combined method is a bit superior to type-based method when the number of selected features is small, and there is no obvious difference between the two methods when the number of selected features is large. Figure 4(b) shows that the micro-F1 performance of using combined distance method is superior to other three feature selection methods when the number of selected features is greater than 2000. Among the three distance methods, combined distance method and type-based distance method are superior to global distance method except when the number of selected features is 100, 200 and 800. Combined method is a bit superior to type-based method when the number of selected features is small, and there is no obvious difference between the two methods when the number of selected features is large. Thus, the experiments show that distance method using LSVM classifier produces highest macro-F1 values in 10 out of 16 cases and highest micro-F1 values in 9 out of 16 cases.

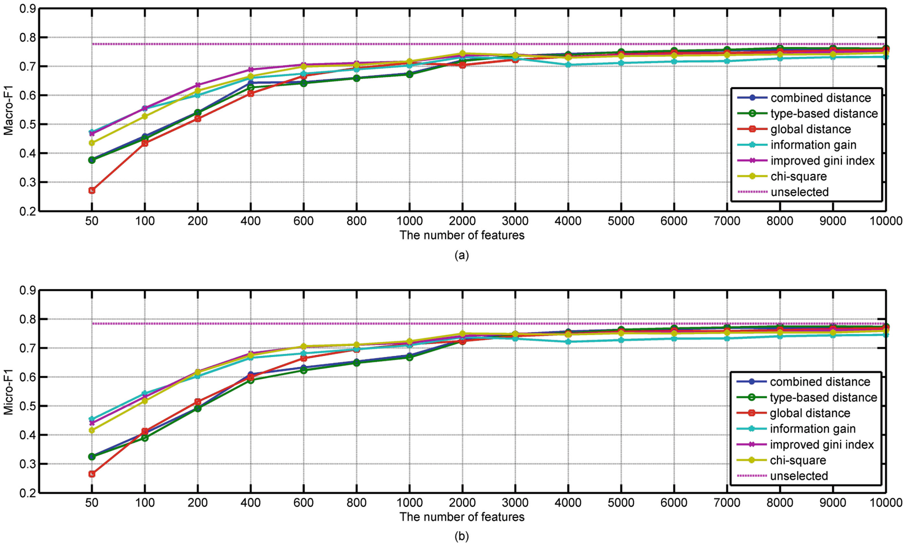


(a)

(b)

Fig. 4. The performance curves of LSVM classifier on 20-Newsgroups. (a) The curves of Macro-F1; (b) The curves of Micro-F1

The performance curves of MNB classifier on 20-Newsgroups are drawn in Fig. 5. Figure 5(a) shows that the macro-F1 performance of using combined distance method outperforms other three feature selection methods when the number of selected features is greater than 4000. Among the three distance methods, combined distance method and type-based distance method outperform global distance method except when the number of selected features is 600, 800 and 1000. Combined method is a bit superior to type-based method when the number of selected features is small, and there is no obvious difference between the two methods when the number of selected features is large. Figure 5(b) shows that the micro-F1 performance of using combined distance method is superior to other three feature selection methods when the number of selected features is greater than 4000. Among the three distance methods, CD method and TD distance method outperform GD method except when the number of selected features is 100, 200, 600, 800 and 1000. Combined method is a bit superior to type-based method when the number of selected features is small, and there is no obvious difference between the two methods when the number of selected features is large. Thus, the experiments show that distance method using MNB



(a)

(b)

Fig. 5. The performance curves of MNB classifier on 20-Newsgroups. (a) The curves of Macro-F1; (b) The curves of Micro-F1

classifier produces highest macro-F1 values in 7 out of 16 cases and highest micro-F1 values in 7 out of 16 cases.

4.3 WebKB

The WebKB [1] dataset is also a popular data set for experiments in text categorization, which collects 8282 webpages from four different college websites. These webpages are unevenly divided into 7 categories: student (1641), faculty (1124), staff (137), department (182), course (930), project (504), other (3764). In the experiment, we just selected 4 categories: course, faculty, project and student. We removed stop words and used bag-of-words model to preprocess the data set. After that, the dimension of the features is 48909. Macro-F1 and Micro-F1 were used to evaluate the performance of different feature selection methods. 10-fold validation was adopted in this experiment.

The performance curves of LSVM classifier on WebKB are drawn in Fig. 6. It can be seen from Fig. 6(a) that the macro-F1 curve of using combined distance method is always higher than other three classical feature selection methods except that the number of selected features is 100. Among the three distance methods, these curves are indented and intertwined. However, averaged macro-F1 value of TD method is higher than GD method, and CD method is higher than TD method. Figure 6(b) shows the same situation as Fig. 6(a). Thus, the experiments show that distance method using LSVM classifier produces highest macro-F1 values in 9 out of 10 cases and highest micro-F1 values in 9 out of 10 cases.

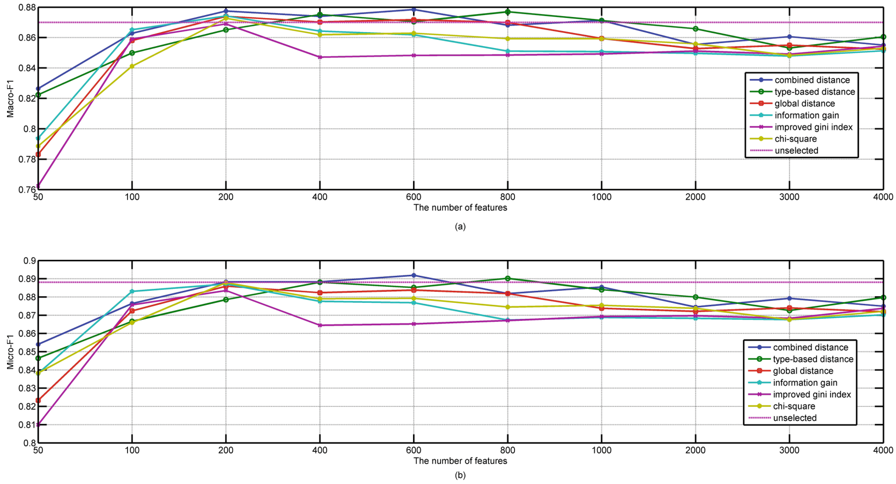


Fig. 6. The performance curves of LSVM classifier on WebKB. (a) The curves of Macro-F1; (b) The curves of Micro-F1

The performance curves of MNB classifier on WebKB are drawn in Fig. 7. It can be seen from Fig. 7(a) that the macro-F1 curve of using combined distance method is higher than other three classical feature selection methods except that the number of selected features is 2000, 3000, 4000. The macro-F1 curves of the three distance methods are also indented and intertwined. However, averaged macro-F1 value of TD method is higher than GD method, and CD method is higher than TD method. Figure 7(b) also shows the same situation as Fig. 7(a). Thus, the experiments show that distance method using MNB classifier produces highest macro-F1 values in 7 out of 10 cases and highest micro-F1 values in 7 out of 10 cases.

4.4 Discussion

The results of Booter websites, 20-Newsgroups and WebKB show that our method outperforms the other metrics in 68.05%, 51.56% and 80.00% cases, respectively. In general, our method produced the highest F1 values in 66% of the classification cases. In the experiment of Booter websites, MNB classifier is more effective than LSVM classifier, and it acquired the highest CAR value, 98.74%. In the experiment of balanced dataset like 20-Newsgroups, we observe that the distance method is not very effective compared with the classical algorithms when the number of selected features is small, however, the distance method perform more effectively and get close to the upper unselected curves earlier when the number of selected features increases. In the experiment of skewed dataset lisk WebKB, we observe that the distance method is very effective in most cases. Among three distance methods, combined distance method and type-based distance method outperform global distance method in most

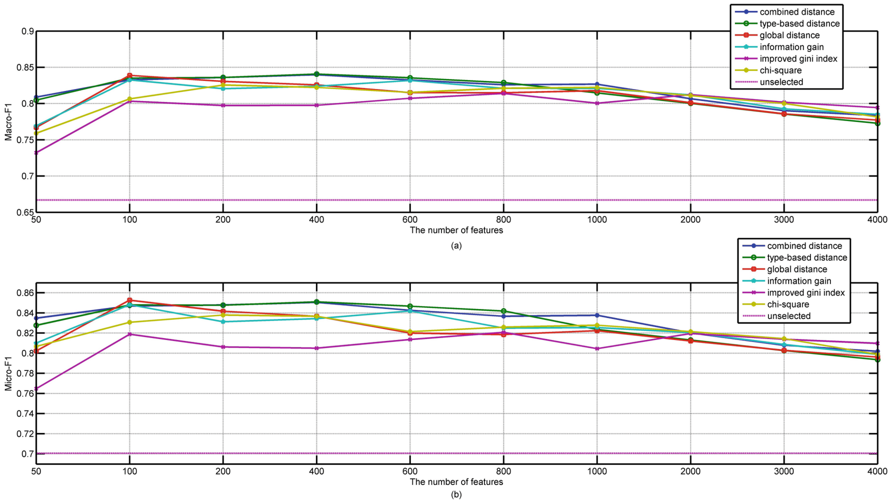


Fig. 7. The performance curves of MNB classifier on WebKB. (a) The curves of Macro-F1; (b) The curves of Micro-F1

cases especially when the number of selected features is very small, and they improve the imbalance problem. In general, combined distance method is also a bit superior to type-based distance method.

5 Conclusion

Booter is increasingly becoming a popular way to launch DDoS attacks, however, there is not much research on Booter and we know little about the ecosystem of these Booter services. In this paper, we develop a new Booter classifier based on text characteristics, which is different from previous work and enrich the methods of identifying Booters. The experiments show that the Booter classifier based on text characteristics has a classification accuracy of 98.74%. We also propose a new feature selection algorithm, which uses the distance between the different categories on a term and select more discriminative features, to improve the performance of text classification. The proposed method is superior to the several classical methods on Booter websites, 20 newsgroups and WebKB dataset in 66% of the classification cases.

Acknowledgement. This paper is Supported by National Key Research and Development Program of China under Grant No. 2017YFB0803003 and National Science Foundation for Young Scientists of China (Grant No. 61702507).

References

1. The 4 universities data set (1998). http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/. Accessed 4 June 2018
2. Home page for 20 newsgroups data set (2008). http://www.qwone.com/~jason/20Newsgroups/. Accessed 4 June 2018
3. Akamai: Third quarter 2016 state of the internet/security report (2016). https://www.akamai.com/us/en/about/news/press/2016-press/akamai-releases-third-quarter-2016-state-of-the-internet-security-report.jsp. Accessed 4 July 2018
4. Goodin, D.: US service provider survives the biggest recorded DDoS in history (2018). https://arstechnica.com/information-technology/2018/03/us-service-provider-survives-the-biggest-recorded-ddos-in-history/. Accessed 4 July 2018
5. Karami, M., Park, Y., McCoy, D.: Stress testing the booters: understanding and undermining the business of DDoS services. In: Proceedings of the 25th International Conference on World Wide Web, pp. 1033–1043. International World Wide Web Conferences Steering Committee (2016)
6. Krämer, L., et al.: AmpPot: monitoring and defending against amplification DDoS attacks. In: Bos, H., Monrose, F., Blanc, G. (eds.) RAID 2015. LNCS, vol. 9404, pp. 615–636. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26362-5_28
7. Krupp, J., Backes, M., Rossow, C.: Identifying the scan and attack infrastructures behind amplification DDoS attacks. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1426–1437. ACM (2016)
8. Krupp, J., Karami, M., Rossow, C., McCoy, D., Backes, M.: Linking amplification DDoS attacks to booter services. In: Dacier, M., Bailey, M., Polychronakis, M., Antonakakis, M. (eds.) RAID 2017. LNCS, vol. 10453, pp. 427–449. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66332-6_19
9. Noroozian, A., Korczyński, M., Gañan, C.H., Makita, D., Yoshioka, K., van Eeten, M.: Who gets the boot? Analyzing victimization by DDoS-as-a-Service. In: Monrose, F., Dacier, M., Blanc, G., Garcia-Alfaro, J. (eds.) RAID 2016. LNCS, vol. 9854, pp. 368–389. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45719-2_17
10. Prince, M.: Technical details behind a 400 Gbps NTP amplification DDoS attack (2014). https://blog.cloudflare.com/technical-details-behind-a-400gbps-ntp-amplification-ddos-attack/. Accessed 4 July 2018
11. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
12. Santanna, J.J.: DDoS-as-a-Service: investigating booter websites. Ph.D. thesis. University of Twente, Enschede, The Netherlands (2017). https://doi.org/10.3990/1.9789036544290
13. Santanna, J.J.: Booters (black)list and ecosystem analysis (2018). https://jjsantanna.github.io/booters_ecosystem_analysis/. Accessed 4 July 2018
14. Santanna, J.J., et al.: Booters—an analysis of DDoS-as-a-Service attacks. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management, IM, pp. 243–251. IEEE (2015)
15. Santanna, J.J., de Vries, J., de O. Schmidt, R., Tuncer, D., Granville, L.Z., Pras, A.: Booter list generation: the basis for investigating DDoS-for-hire websites. Int. J. Netw. Manag. **28**(1), e2008 (2018)
16. Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., Wang, Z.: A novel feature selection algorithm for text categorization. Expert Syst. Appl. **33**(1), 1–5 (2007)

17. Yan, J., et al.: OCFS: optimal orthogonal centroid feature selection for text categorization. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 122–129. ACM (2005)
18. Yang, J., Qu, Z., Liu, Z.: Improved feature-selection method considering the imbalance problem in text categorization. Sci. World J. **2014**(3) (2014)
19. Yang, Y., Pedersen, J.O.: A comparative study on feature selection in text categorization. In: ICML, vol. 97, pp. 412–420 (1997)