Mitigating False Positives in DGA Detection for Non-English Domain Names

Huiju Lee
School of Cybersecurity, Korea University
Seoul, Republic of Korea
kkj0118@korea.ac.kr

Huy Kang Kim
School of Cybersecurity, Korea University
Seoul, Republic of Korea
cenda@korea.ac.kr

***Abstract*—The existing machine learning and deep learning-based domain generation algorithm (DGA) detection methods often encounter false positives due to domain names representing non-English. To this end, we propose a DGA detection method that includes a domain name embedding approach capable of effectively representing the linguistic patterns in domain names. We here focus on Chinese domain names among non-English-based domain names. The proposed method consists of three steps as follows: (1) subword-based domain name embedding, (2) statistical feature extraction, and (3) deep learning-based detection. Experimental results demonstrate that our method overcomes misclassification of legitimate Chinese domains as DGA domains, thereby enhancing the overall performance of the DGA detection model.**

***Index Terms*—Botnet, Deep learning, Domain Generation Algorithm, Embedding, Subword segmentation**

I. Introduction

Domain generation algorithm (DGA) is an algorithm used to generate many domain names randomly to establish communication between bots and command and control servers. In response to security measures that block hard-coded domain names in malware, attackers started using dynamically generated domain names via DGAs. Recently, the range of malicious activities through malware by attackers has broadened. Consequently, DGAs can be used in PCs and IoT devices, drones, or vehicles, posing significant safety and security risks. DGA domain names are difficult to identify, so various studies have proposed machine learning and deep learning-based DGA detection methods [1]–[4]. While existing methods achieve high detection rates, they still suffer from false positives on non-English domain names—such as zhangqiaokeyan.com, the Chinese scientific research platform 掌桥科研 (*zhangqiaokeyan*). In China, domain names are typically created using Pinyin, the Romanization system for the Chinese language.

The reason non-English-based benign domain names are incorrectly identified as DGA domain names in existing DGA detection methods is as follows: First, non-English domain names exhibit significantly different linguistic features compared to English-based ones. We selected Chinese among non-English languages and analyzed 2-grams for benign English, benign Chinese, and DGA domain names to verify it (see Fig. 1). The x-axis indicates the top 10 2-grams in terms of frequency ratio within each domain name category. The figure clearly shows differences in the types and frequency ratios of 2-grams occurring in three categories. However, existing studies potentially assume that benign domain names are based on the English language. Second, the majority of deep learning-based DGA detection models embed domain names using characters. Such simplistic embedding might make it difficult to represent subword-level patterns beyond the character level in domain names.

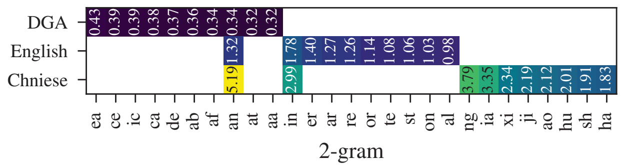


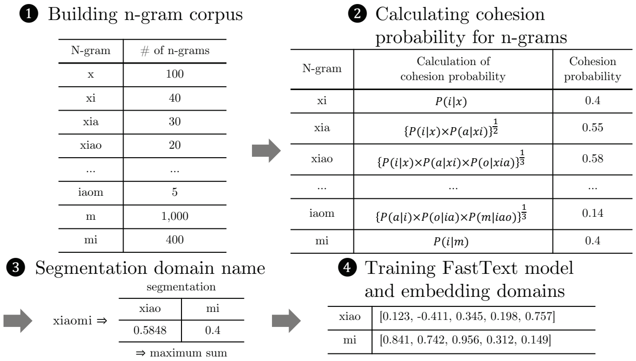
Fig. 1: Top 10 2-gram populations [%] over three domain categories. Each category is clearly distinguishable. A blank means not available (N/A) in the top 10 for each category.

In this study, we propose a new DGA detection method that utilizes an advanced embedding approach to more accurately represent non-English domain names' linguistic patterns and reduce false positives. By segmenting and embedding domain names into meaningful patterns and contexts that capture subword n-grams beyond the character level, our deep learning-based model can effectively distinguish benign Chinese domain names from DGAs. This method aims to improve the detection accuracy for Chinese-based benign domain names, which comprise a significant portion of global domain traffic and are frequently generated using Pinyin.

II. Methodology

The proposed method consists of three main stages: embedding, feature extraction, and detection. The embedding stage involves segmenting the domain name into subwords and vectorizing it. Fig. 2 details the embedding stage. In ❶, a corpus for benign domain names is built based on n-grams and frequency counts. Since our subword criterion is syllables, we consider $n \in [1..4]$. In ❷, the corpus is used to calculate the cohesion probability [5] of each n-gram. Cohesion probability represents how frequently a word appears clustered together in the corpus. Our method uses cohesion probability to infer the boundaries of the subwords composing the domain name. In ❸, the segmentation of the domain name is determined by the combination of subwords in which the sum of the cohesion probabilities of each subword is the maximum from the combination of all possible subwords in the domain name. Finally, in ❹, the segmented domain names are embedded using the FastText [6] model. The FastText model enables embedding that represents the context between subwords. The feature extraction stage involves extracting four statistical features

2833-292X/24/$31.00 ©2024 IEEE
DOI 10.1109/DSN-S60304.2024.00042



❶ Building n-gram corpus
❷ Calculating cohesion probability for n-grams
❸ Segmentation domain name
❹ Training FastText model and embedding domains

N-gram	# of n-grams
x	100
xi	40
xia	30
xiao	20
...	...
iaom	5
m	1,000
mi	400

N-gram	Calculation of cohesion probability	Cohesion probability
xi	$P(i\|x)$	0.4
xia	$\{P(i\|x)\times P(a\|xi)\}^{\frac{1}{2}}$	0.55
xiao	$\{P(i\|x)\times P(a\|xi)\times P(o\|xia)\}^{\frac{1}{3}}$	0.58
...	...	...
iaom	$\{P(a\|i)\times P(o\|ia)\times P(m\|iao)\}^{\frac{1}{3}}$	0.14
mi	$P(i\|m)$	0.4

xiaomi ⇒ segmentation

xiao	mi
0.5848	0.4

⇒ maximum sum

xiao	[0.123, -0.411, 0.345, 0.198, 0.757]
mi	[0.841, 0.742, 0.956, 0.312, 0.149]

Fig. 2: Proposed four-stage domain name embedding process

from the domain name string to create a feature vector. This stage exists to effectively represent random-based DGA, which typically have lower subword frequencies, making embedding challenging. To do this, various lexical-based statistical features capable of distinguishing between benign and DGA domain names were selected and trained using the LightGBM model. Subsequently, the importance of each feature output by the model was compared, and four features were chosen—domain name length, frequency of transitions between vowels and consonants, frequency of transitions in vowels, consonants, and vowels order, and frequency of transitions between English characters and non-English characters. In the detection stage, a convolutional neural network (CNN)-long short-term memory (LSTM)-based detection model followed by a dense layer with Sigmoid activation is trained with the embedded training domain names and statistical feature vectors. When given the domain name vector and feature vector as an input, the trained model determines whether the domain name is benign or DGA.

III. Experimental Results and Future Work

In this section, we compare the performance of the proposed method with existing methods to evaluate its detection capabilities. Existing methods [1], [2] employ character-based embedding and use LSTM, CNN, and CNN-LSTM-based detection models, respectively. Xu *et al.* [3] converted domain names into n-gram representations and used the CNN-based detection model. Yun *et al.* [4] suggested another method that tokenizes domain names based on n-grams.

We collected the benign domain name list from the Alexa Top 1 Million Domains, Majestic Million. For DGA domain names, we collected data from DGArchive [7] and UMUDGA [8]. We used 11 DGA families—banjori, ccleaner, cryptolocker, dircrypt, dyre, locky, matsnu, pushdo, qadars, ramdo, and symmi. These families are based on various generation methods and include DGA domain names that could be confused with domain names based on the Chinese language. We labeled whether benign domain names are English-based or Chinese-based to evaluate the detection accuracy for benign domain names. As a result, 74,675 benign and 100,925 DGA-generated domain names were included. In total, 20% of the collected dataset was used as the test dataset.

The detection performance of the methods was evaluated using accuracy, precision, recall, and F1-score. The experimental results are shown in Fig. 3. Compared to the existing methods, the proposed approach shows enhanced accuracy (*i.e.,* F1-score ≈ 0.99) for benign Chinese and English domain names. Furthermore, the proposed method outperforms the existing five methods for benign and DGA domain names. It is worth noting that the accuracy of Chinese domain names is improved by up to 0.12. While the existing methods showed lower detection accuracy (F1-score ≈ 0.92) for Chinese domain names than English domain names, the proposed method also exhibits high performance for Chinese as well as English domain names. Domain names that induce false positives in the existing methods (*e.g.,* hehuoxinbaiyou, huxianqiu and mianfeiku) often include subwords that are frequently used in Chinese but not in English. Taken together, we can conclude that the proposed DGA detection method can reduce false positives for benign Chinese domain names and enhance the overall performance of DGA detection.

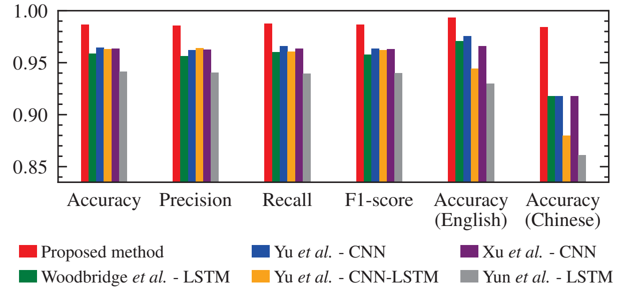


Fig. 3: Performance comparison. The proposed method (red) outperforms the existing DGA detection methods.

This paper is the first attempt to consider Chinese-based benign domain names in DGA detection. In future work, we aim to expand the proposed method to other non-English-based domain names towards practical DGA detection.

Acknowledgement

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant RS-2021-KA162334).

References

[1] J. Woodbridge, H. S. Anderson, A. Ahuja, and D. Grant, "Predicting domain generation algorithms with long short-term memory networks," *arXiv preprint arXiv:1611.00791*, 2016.

[2] B. Yu, J. Pan, J. Hu, A. Nascimento, and M. De Cock, "Character level based detection of DGA domain names," in *Proc. IJCNN*. IEEE, 2018.

[3] C. Xu, J. Shen, and X. Du, "Detection method of domain names generated by DGAs based on semantic representation and deep neural network," *Comput. Secur.*, vol. 85, pp. 77–88, 2019.

[4] X. Yun, J. Huang, Y. Wang, T. Zang, Y. Zhou, and Y. Zhang, "Khaos: An adversarial neural network DGA with high anti-detection ability," *IEEE Trans. Inf. Foren. Sec.*, vol. 15, pp. 2225–2240, 2019.

[5] H. Kim, "soynlp: Python package for Korean natural language processing," 2019. [Online]. Available: https://github.com/lovit/soynlp

[6] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the association for computational linguistics*, vol. 5, pp. 135–146, 2017.

[7] F. FKIE. (2022) DGArchive. [Online]. Available: https://dgarchive.caad.fkie.fraunhofer.de/welcome/

[8] M. Zago, M. G. Pérez, and G. M. Pérez, "UMUDGA: A dataset for profiling algorithmically generated domain names in botnet detection," *Data in Brief*, vol. 30, p. 105400, 2020.