Research on Visual Malicious Code Classification Based on Improved Faster R-CNN

Liang Zhen [1], Yuntao Zhao [*1]

[1]*Shenyang Ligong University, College of Information Science and Engineering, Liaoning Shenyang 110159, China*

Abstract

With the development of Internet technology, the number of malicious software is increasing and malicious attacks are becoming rampant. Therefore, the research on malicious software has a great application prospect. This paper proposes a visual malicious code classification method based on improved Faster R-CNN. Rank & Sort Loss is used to optimize the loss function of the Faster R-CNN model, in order to reduce the number of hyperparameters, improve the performance of the model, and make it more robust to the problem of class imbalance in training. The experimental results show that using the improved Faster R-CNN detection method has a further improvement in the recognition accuracy of malicious code grayscale images compared with the classic Faster R-CNN detection method.

Keywords

Malicious code visualization, Rank & Sort loss, Faster R-CNN network, object detection

1. Introduction

In 2020, according to the detection and dissemination of malicious programs by CNCERT / CC (National Internet Emergency Center), more than 42.98 million samples of malicious programs were found in the whole year. According to the above report, malicious code has great harm. Based on the above situation, it is of great significance to study malicious code. At this stage, there have been many researches on malicious code analysis and detection methods. In 2011, L.Nataraj et al. proposed a malicious code visualization method and classification method, which converts binary data files of malicious code into texture images, and classifies malicious software through KNN model and Euclidean distance[2]. Through experimental demonstration, this method can effectively improve the detection speed of malicious code, and also ensure the accuracy level of traditional static detection methods. On this basis, many researchers have begun to try to use the detection and classification method based on malicious code image to train the appropriate malicious code classifier. For example, Zhang Jinglian et al. proposed a malicious code classification technology based on feature fusion, which extracts and fuses the features by extracting the opcode instructions and grayscale image texture of malicious code, and uses Random Forest (RF) to classify the malicious code families[3]. The above methods all visualize the malicious code, convert the malicious code into a grayscale image for further classification, and achieve good results.

With the application of Convolutional Neural Network (CNN) in object detection, more and more researchers have put forward a series of achievements. For example, Ross B. Girshick proposed the Faster R-CNN network[5]. For target detection in grayscale images of malicious code, we need to detect the position of the section (.text) where the core feature opcodes are located in the grayscale image. The loss of the traditional Faster R-CNN model is composed of two parts: classification loss and regression loss. During the training process, many hyperparameters will be generated, which requires human and material resources to adjust the parameters. At the same time, the imbalance of data distribution will

ISCIPT2022@7th International Conference on Computer and Information Processing Technology, August 5-7, 2022, Shenyang, China
EMAIL: 771282215@qq.com (Liang Zhen); corresponding author: zhaoyuntao2014@163.com (Zhao Yuntao)
ORCID: 0000-0002-5479-7268 (Liang Zhen); 0000-0002-2627-8276 (Zhao Yuntao)

also affect the detection effect of the model. Therefore, this paper optimizes the loss function for the above problems, and introduces Rank & Sort Loss to achieve better detection results.

2. Related Work

2.1. Research on Malicious Code Visualization

Static disassembly of malicious code to achieve the visualization of malicious code. IDA Pro is used for static disassembly, by importing malicious software into IDA Pro to get malicious code binary file (.bytes file) and assembly file (.asm file).On this basis, the obtained binary executable file is used as input data, and regard it as the original Bytes binary stream. A hexadecimal number can be considered as a combination of binary number, and four binary numbers can be converted to a hexadecimal number. Because the range of hexadecimal number is only between 0 and 16, which corresponds to two hexadecimal numbers of 256 pixel value of grayscale image, this method can convert the original data into a simple gray image. The sequence of binary streams corresponding to the gray level of each 8-bit pixel is segmented and then arranged into a sequence to form the corresponding grayscale image. Figure 1 shows the schematic diagram of malicious code visualization.

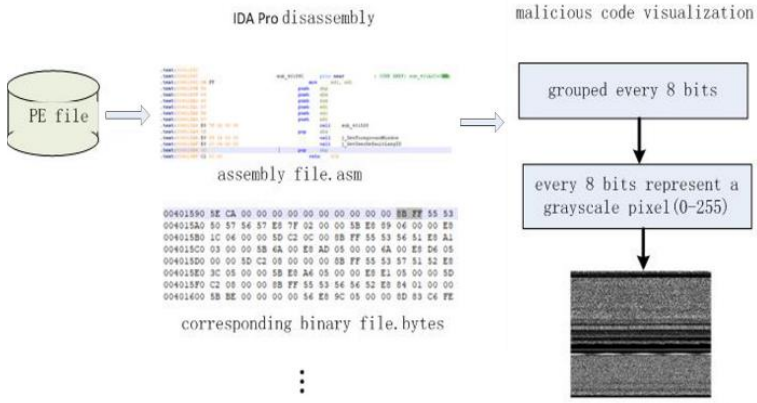

Figure 1: Visualization of malicious code

2.2. Faster R-CNN Network Model

Faster R-CNN is a deep learning network model based on region proposal network. The function of target positioning is added on basis of the CNN network model. Compared with the RCNN and Fast R-CNN detection networks, the Faster R-CNN network implements an end-to-end network training mode, so that the CNN for generating the proposal window and the CNN for object detection share operations. The structure of Fast R-CNN mainly includes backbone extraction network, RPN region proposal network, region of interest pooling layer and classifier. The Faster R-CNN network first processes the pictures, obtains the common feature layer, and then obtains the suggestion frame, then uses the suggestion frame to intercept the common feature layer, and adjusts some feature layers after the interception to the same size through the ROI pooling layer, and finally Perform classification and regression. The network structure is shown in Figure 2.

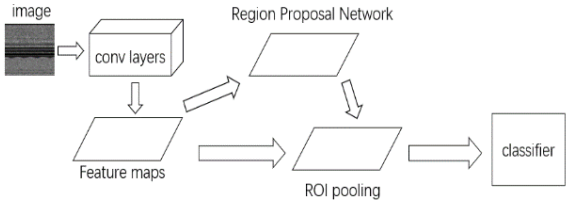

Figure 2: Faster R-CNN network structure

2.3. Rank & Sort Loss

Rank & Sort (RS) Loss [8] was proposed by K. Oksuz et al in 2021. Rank & Sort Loss is composed of Rank loss and Sort loss. Rank loss enables all positive samples to be sorted before negative samples, and only the negative samples with higher scores are selected for calculation. Sort loss uses IoU as the classification label, so that the positive samples in the prediction box can be sorted by continuous values as the label. Besides, Rank & Sort Loss does not require multi-task weight or coefficient adjustment. The definition of the loss function is shown in formula (1):

$$L_{RS} = \frac{1}{|P|}\sum_{i\in P}(l_{RS}(i) - l_{RS}^{*}(i)) \tag{1}$$

Where P is the collection of positive sample, $l_{RS}(i)$ is the sum of rank error and sort error at this time, $l_{RS}^{*}(i)$ is the sum of the target rank error and the target sort error, using equations (2) and (3) respectively express:

$$l_{RS}(i) = \frac{rank^{-}(i)}{rank(i)} + \frac{\sum_{j\in P} H(x_{ij})(1-y_j)}{rank^{+}(i)} \tag{2}$$

$$l_{RS}^{*}(i) = l_{R}^{*}(i) + \frac{\sum_{j\in P} H(x_{ij})[y_j \geq y_i](1-y_j)}{\sum_{j\in P} H(x_{ij})[y_j \geq y_i]} \tag{3}$$

Where i and j are the sample numbers, y is the score label, and $rank(i)$ is the number of all positive samples and negative samples that are greater than or equal to the positive sample's classification score; $rank^{-}(i)$ is the number of all negative samples that is greater than or equal to the positive sample's classification score; $rank^{+}(i)$ is the number of all positive samples that is greater than or equal to the positive sample's classification score, x_{ij} is the classification score, $H(x)$ is the unit step function, $l_{R}^{*}(i)$ is the target rank error.

3. Task Description

The training process for the whole network mainly includes the following steps:

1. Convert the malicious code into a grayscale image dataset and preprocess the dataset. The first step is to use the disassembly tool IDA Pro to disassemble, turn the program used for detection into a binary file, and map it into a corresponding grayscale image through visualization. Modify the label names to the labels of 6 types of malicious code images.
2. Feature extraction is performed on the input image through the backbone extraction network. In this paper, ResNet50 is used as the backbone extraction network. ResNet50 has two basic blocks, Conv Block and Identity Block. The main difference between the two is that Conv Block performs convolution operations on the residual edge, while Identity Block does not perform convolution. ResNet50 contains 1 Conv Block, and the number of Identity Blocks is 3, 4, 6, and 3, respectively.
3. The features output by the backbone extraction network are sent to the proposal box, where one convolution channel number is 18, which is used to predict whether each prediction box contains an object, and the other convolution channel number is 36, which is used to adjust the prior box, get a suggestion box. Then, the classification and regression are carried out through the ROIPooling layer. The improved method is to replace the cross entropy loss in the classification loss with the RS loss, and the regression loss adopts the GIoU loss. The weighted parameter of the regression loss of the improved model is the RS loss divided by the regression loss.

4. Experimental Process and Analysis

4.1. Experimental Data Set

This experiment selects malicious sample data from the Kaggle platform of the Microsoft Malware Security Defense Center. There are 1839 samples from 6 malicious code sample families, as shown in Table 1.

Table 1
Malicious sample data

Malicious code family name	Number of training samples	Type
Ramnit	354	Worm
Vundo	278	Trojan
Tracur	295	Trojan
Gatak	243	back door
Obfuscator.ACY	400	malicious advertisement
Lollipop	269	malicious advertisement

4.2. Experimental Environment

Table 2
Experimental environment

Experimental platform	Information
Operating system	ubuntu16.04
Graphics card	RTX 2080Ti
Drive	Nvidia 430.34
CUDA	9.0
Language	Python 3.6
Framework	PyTorch

4.3. Analysis of Experimental Results

In the experiment, the data were randomly divided into training set and test set according to 8:2. The accuracy rate and recall rate are selected as the evaluation criteria. Table 3 shows the accuracy rate and recall rate of each class of the traditional machine learning classifier, the traditional Faster R-CNN network framework and the improved Faster R-CNN network framework. The experimental results show that the Faster R-CNN network model is better than the traditional machine learning classification model, and the improved Faster R-CNN network model is further improved on the basis of the traditional Faster R-CNN network model. After the introduction of RS Loss, the accuracy of the model is increased by 1.9 percentage points compared with the original model. At the same time, the complexity of the model is measured by Floating point Operations (FLOPs) and Parameters. The results show that the amount of Parameters of the model is reduced by 15.37% compared with the original model, and the amount of FLOPs is reduced by 23.58%. It shows that the introduction of RS loss can greatly reduce the amount of parameters and effectively reduce the complexity of the model.

Table 3
Malicious sample data

Classification method	Accuracy	Recall
KNN	0.611	0.427
RF	0.889	0.836
Faster R-CNN	0.894	0.877
RS+Faster R-CNN	0.913	0.895

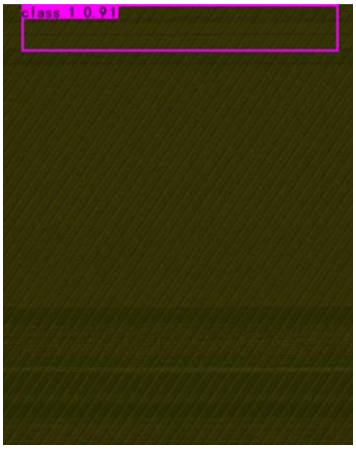

Figure 3: inspection effect diagram

5. Conclusion

In order to further improve the detection and classification effect of malicious code, this paper proposes a malicious code classification model based on improved Faster R-CNN. The method of using Rank & Sort loss function effectively reduces the number of hyperparameters. In the process of model training, there is no need to repeatedly adjust the hyperparameters. We only need to adjust the learning rate to improve the model performance, avoiding the complex parameter adjustment process and one loss dominant situation. The experimental results show that the model is more feasible and effective. In the following work, the detection ability of the model can be further improved by adding Attention Mechanism and Data Augmentation.

6. Reference

[1] CNCERT, China Internet network security report 2020 [r/ol] [2021-07-21]. https://www.cert.org.cn/publish/main/upload/File/2020%20Annual%20Report.pdf.
[2] Nataraj L, Karthikeyan S, Jacob G, et al. Malware Images: Visualization and Automatic Classification. ACM, 2011.
[3] Zhang Jinglian, Peng Yanbing. Research on malicious code classification based on feature fusion[J]. Computer Engineering, 2019,45 (08): 281-286+295. DOI: 10.19678/j.issn. 1000-3428.0051790.
[4] Liu Yashu, Wang Zhihai, Hou Yueran, Yan Hanbing. Visualization and automatic classification of malicious code with enhanced information density[J]. Journal of Tsinghua University (Natural Science Edition), 2019,59 (01): 9-14.
[5] Ren S, He K, Girshick R , et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017, 39(6):1137-1149.
[6] Saxe J, Berlin K . Deep neural network based malware detection using two dimensional binary program features[C]// International Conference on Malicious & Unwanted Software. IEEE, 2015.
[7] Ibrahim Ghafir, Vaclav Prenosil. Malicious File Hash Detection and Drive-by Download Attacks[J]. 2016.
[8] K Oksuz, Cam B C, Akbas E, et al. Rank & Sort Loss for Object Detection and Instance Segmentation[J]. 2021.
[9] Wang Yinglong, Huang Zuyuan, Liu Ailian, Lichuan Testing of malicious code detection method based on texture feature[J]. Mobile communication, 2017,41 (13): 46-49.
[10] Kunwar R S, Sharma P. Malware Analysis: Tools and Techniques[C]// International Conference on Information & Communication Technology for Competitive Strategies. 2016.
[11] Jinpei Yan, Yong Qi, Qifan Rao. Detecting Malware with an Ensemble Method Based onDeep Neural Network[J].2018.