이미지 기능을 사용하여 맬웨어를 분류하는 CNN 모델
(CNN Model to Classify Malware Using Image Feature)

카방가 †  김창훈 ‡
(Espoir K. Kamundala) (Chang Hoon Kim)

요 약 인터넷에 발생하는 악성코드는 매우 심각한 위협 요소이며, 악성코드를 이용한 공격이 전 세계적으로 전파되며 심지어 쉽게 더 지능적으로 변조되고 있다. 그러므로 악성코드를 정확하게 탐지하는 방법이 중요하다. 지금까지 알려진 악성코드 대응방법은 악성코드를 탐지하여 삭제하거나 혹은 처리한다고 알고 있다. 이러한 악성코드를 탐지하기 위하여 악성코드에 따른 분류를 하여야 한다. 악성코드를 잘 분류하는 것은 알려진 악성코드를 더 잘 탐지할 수 있다는 것과 같다. 기존 연구들을 보면 같은 카테고리에 속하는 악성코드는 반응방법이 비슷하게 보이는 경향이 있다는 것을 입증한다. 그리고 많은 새로운 악성코드들이 기존에 있던 악성코드로부터 만들어진다는 것을 증명한다. 따라서 악성코드를 종류에 따라서 분류하는 것은 탐지하는 것과 지나치게 아주 중요한 작업이다. 그러므로 악성코드 분류 기술은 젤한 요구된다. 본 논문에서는 주어진 종류에 따라 악성코드를 분류하기 위한 컨볼루션 인공신경망 모델을 구축한다. 이 모델은 악성코드 데이터 세트를 25 개의 종류로 분류하는 실험에서 98%의 정확도를 보인다. 본 연구의 목적이 다른 목표로 더 많은 양의 악성코드 파일에 적용되며 더 높은 정확도를 보이는 것이다.

키워드: 악성코드, 악성코드분류, CNN, 머신러닝

Abstract Malware programs are common threats in the information and technology society. It has been proven that a number of developed malwares cripples the victim’s computer as well as launching malicious attacks. Therefore, it is important to find a reasonable technical way to counter these attacks. Malware can be easily detected by checking whether a file has a malicious code inside the source code, if you detect a malicious code inside your content, then take an appropriate action by eliminating the threat. The first countermeasure to take is to delete the file or follow any other action defined by an Anti-malware software. After a file is infected, means it can be classified to its corresponding family based on its behavior in the infected system. In this paper, we use Convolutional Neural Network to classify malware binaries using image features. Our work relies on the previously conducted research on malware visualization, whereby we used the dataset consisted of about 9,500 samples of 25 different malware families. The built architecture achieved an accuracy of 98%.

Keywords: malware, malware classification, convolutional neural network, machine learning
1. Introduction

The huge increase in malware attack is a direct consequence of the increased in the number of electronic devices and internet access. In fact, according to [2], the number of internet users has gone from less than 1% to almost 40% in only 20 years. Figure 1 shows how the number of internet users increases over the last 20 years.

The access to internet and the use of electronic devices such as mobile phones, digital audio video players, game consoles, computers, etc. have therefore increased the risk to be infected by a malicious software, known as malware. Even though the number of new malware have decreased during the last three quarters of the year 2016, according to McAfee Labs Threats Report of June 2017 [3], it is important to mention that every single day, hundreds of new malware are created and this constitutes a very big threat to the information technology environment. The newly created malware are usually very similar to the existing ones.

Figure 2 shows that the total number of malware is about to cross seven hundred million.

The term malware is very broad because there are many different types of malware according to the harm that they cause to the infected system. Among them, there are:

- virus: it is a type of malware that is capable of spreading itself from the infected computer to other computers and cause severe disfunctions to files in the systems
- worm: this is a standalone program capable of spreading itself through the network directly.
- Trojan: This type of malware appears to be normal working file or program while it hides malicious content that can harm a system.
- Adware: an adware usually directs a system user to unwanted links containing advertisements or download of malicious files.
- Spyware: It always unauthorized access to a system and enables to the attacker theft of important information or modification of files.
- Rootkit: It can either control or access a computer without the knowledge of the owner and the security program.
- Backdoor: it is created so that unauthorized remote access to a system can be secured and it helps to compromise a system’s normal authentication procedures.
- Botnet: It performs some automatic operations in the infected system. It gets command from the attacker through a command-and-control server.

Apart from the above listed types of malwares, there are many others that exist and perform different actions in the systems.

Two major operations can be performed on the malware: detection and classification. Detection means being able to say whether a file or a program is malicious or benign and classification is done in order to classify malware to their corresponding categories regarding their behavior or the harm that they cause.

Traditional methods that include signature based, heuristic-based or also specification-based techniques have long been used for the task of malware detection and classification [9]. However, these techniques have shown considerable limitations, such that not detecting or classifying new malware, high rate of false positive and false negative. These limitations
have encouraged researchers to use machine learning techniques since they have proven to be more robust and efficient. Using machine learning techniques, it is possible to detect and classify new variant of malware and to reduce the rate of false positive and false negative.

But it is also very important to mention that, regardless of how efficient and robust machine learning techniques can be, they present also certain vulnerabilities to adversarial data, as it has been shown in [12,13]

2. Related Work

Classifying malware is important in the way that it helps to understand the way malware function and therefore it can be easy to build strong and efficient architectures to fight them. Malware classification research started more than 20 years ago. The work proposed in [4] in 1994 is among the first ever proposed classifier inspired from the nature.

Later, several other techniques were used for malware classification. [10] analyzed malware and extracted features based on the behavior of the malware to classify them into their families, with an accuracy of 88%. In [11], the author developed an automatic classification architecture that uses basic code blocks of disassembled binaries to represent the malware samples.

Two decades later, because of the increasing number of new malware that become more and more sophisticated, intelligent method of classification came into action with machine learning algorithms. Machine learning methods appear to be very efficient and produce best result compared to old methods. Work conducted in [5] used machine learning and data mining technique for detection and classification of malware executables. They compared several techniques including naive Bayes, decision trees, support vector machine and boosting for malware classification. They concluded that boosted decision tree performed better compared to other techniques. [6] proposed several architectures that included a Long–Short Term Memory language model and a Gated Recurrent Unit language model, and the experimental results showed an improvement of 31.3% in true positive rate compared to the system used in [7]. In their work, [7] used recurrent neural network, where the language model of the malware through executed instruction was learnt.

The biggest and most important inspiration of our work comes from [1]. In their work, they were able to visualize malware as grayscale image in the range of [0, 255] where 0 represents the black color and 255 represents white. They observed that the obtained images presented different sections and each section meant a particular information about the malware. Figure 3 shows an example of an obtained image with different sections.

From the above picture, the top most section contains the executable code, the second and third sections contain uninitialized code and initialized data and the last section contains the module resources. They used GIST to compute image texture and k-nearest neighbors with Euclidean distance for classification where they obtained 98% accuracy.

3. Proposed Model

Figure 4 shows describes our built architecture. This Convolutional Neural Network architecture has the following properties:

a) An input layer: After converting the binary files into images, these have different shapes. We therefore need to reshape them into 128×128×1 dimension, before they are used as input for our model. PIL package, which is an image library of Python was used to obtain vectors of images.

b) Three convolutional layers: We inserted 3 con-
volutional layers for which, at Rectified Linear Units layers, we used a 2-dimensional convolutional layer.

The Rectified Linear Units were used instead functions such as tanh or sigmoid because it plays an important role in preventing the vanishing gradient problem might occur in the lower layer of the network. After each convolutional layer, an activation layer is used. We use a max-pooling layer between each convolutional layer.

c) Two fully-connected layers: After we have inserted the obtained images in the input layer and after they have crossed the convolutional layers, we obtain an array of vectors. These vectors need to be converted to a single layer, called as fully-connected layer. We use two fully-connected layers to avoid loss of data while directly down sampling all the vectors.

d) An output layer: the output layer is made of 25 neurons that correspond to 25 families of malware available for the used dataset.

We used cross entropy loss function for classification task and Adam optimizer for optimization. The advantage of using cross entropy loss function is that it avoids to slowing down the learning and Adam optimizer is very effective because it can reach good results quickly.

The advantage of using convolutional neural networks is that they are good at extracting features and they can be applied to the whole images at a time. Therefore, the idea of converting malware binaries into images is an important innovation for working those intending to use convolutional neural networks.

4. Dataset and Experimental Result

Our model was evaluated on set of 9,458 images extracted from malware binaries from 25 different families. This dataset is available on [8]. Looking closely, we can notice images extracted from malware of same family look similar to each other. Among the available dataset, 8,500 samples were used for training and the remaining for testing.

Figure 5 and Table 1 describes the available malware families and the number of sample for each family.

Figure 6 shows two images that are extracted from 2 different malware of the same family and we can notice that the texture are very similar.
<table>
<thead>
<tr>
<th>Family Name</th>
<th>Class</th>
<th>No. of samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allaple. L</td>
<td>Worm</td>
<td>1591</td>
</tr>
<tr>
<td>Allaple. A</td>
<td>Worm</td>
<td>2949</td>
</tr>
<tr>
<td>Yuner. A</td>
<td>Worm</td>
<td>800</td>
</tr>
<tr>
<td>Lolyda. AA 1</td>
<td>PWS</td>
<td>231</td>
</tr>
<tr>
<td>Lolyda. AA 2</td>
<td>PWS</td>
<td>184</td>
</tr>
<tr>
<td>Lolyda. AA 3</td>
<td>PWS</td>
<td>123</td>
</tr>
<tr>
<td>C2Lop. P</td>
<td>Trojan</td>
<td>146</td>
</tr>
<tr>
<td>C2Lop.gen/G</td>
<td>Trojan</td>
<td>298</td>
</tr>
<tr>
<td>Instantaccess</td>
<td>DIELor</td>
<td>431</td>
</tr>
<tr>
<td>Swizzor.gen/l</td>
<td>Trojan</td>
<td>132</td>
</tr>
<tr>
<td>Swizzor.gen/E</td>
<td>Trojan</td>
<td>128</td>
</tr>
<tr>
<td>VBA.T</td>
<td>Worm</td>
<td>408</td>
</tr>
<tr>
<td>Fakerean</td>
<td>Rogue</td>
<td>381</td>
</tr>
<tr>
<td>Aluron.gen/J</td>
<td>Trojan</td>
<td>198</td>
</tr>
<tr>
<td>Malex.gen/J</td>
<td>Trojan</td>
<td>136</td>
</tr>
<tr>
<td>Lolyda.AT</td>
<td>PWS</td>
<td>159</td>
</tr>
<tr>
<td>Adialer.C</td>
<td>DIELor</td>
<td>125</td>
</tr>
<tr>
<td>Wnítrim. BX</td>
<td>Trojan Downloader</td>
<td>97</td>
</tr>
<tr>
<td>Dialplatform.B</td>
<td>DIELor</td>
<td>177</td>
</tr>
<tr>
<td>Dontov.A</td>
<td>Trojan Downloader</td>
<td>162</td>
</tr>
<tr>
<td>Obfuscator. AD</td>
<td>Trojan Downloader</td>
<td>142</td>
</tr>
<tr>
<td>Agent. FYI</td>
<td>Backdoor</td>
<td>116</td>
</tr>
<tr>
<td>Autorun. K</td>
<td>Worm. AutoT</td>
<td>106</td>
</tr>
<tr>
<td>Rbot.gen</td>
<td>Backdoor</td>
<td>138</td>
</tr>
<tr>
<td>Skintrim. N</td>
<td>Trojan</td>
<td>80</td>
</tr>
</tbody>
</table>

Our method was evaluated based classification accuracy which is the number of correct predictions made divided by the total number of predictions, multiplied by 100, to get the percentage value. The experiment produces an accuracy of 98%, which is quite promising yet needs improvement. This result is similar to the one achieved in [1] where they used K-nearest neighbors with Euclidean distance for classification.

V. Conclusion

In this work, we built a convolutional neural network for classifying malware binaries using images extracted from them. Our work was inspired by the work implemented by Nataraj et al. of visualizing malware into gray-scale images. The importance converting malware binary into an image is that different section of a binary can be easily seen when it is visualized, malware of same family look alike when seen as images and are different from other families. Our model, which is based on convolutional neural network achieved a promising accuracy, but this can always be improved in our future works.

References

Mr. Espoir K. Kamundala earned his master's degree student in computer and information engineering at Daegu University at the Republic of Korea in 2018. He earned a bachelor degree in computer applications in 2015 at CMR Institute of Management Studies (affiliated to Bangalore University) in Bangalore, India. His research area includes machine learning, malware detection and classification and forensic science.

Mr. Chang Hoon Kim is an associate professor at Daegu University, Republic of Korea. He earned his bachelor of computer science engineering in 2001, a master degree of computer and information engineering in 2003 and a Ph. D in computer science engineering in 2006, both at Daegu University in the Republic of Korea. Professor Kim has published many papers in international and domestic journals and has attended many international and domestic conferences for presentations. His research area includes network security, system security and artificial intelligence.