Deep Android Malware Detection

Lyujiuyang, Dec 29, 2021

Info

- Authors: Niall McLaughlin, Jesus Martinez del Rincon, BooJoong Kang. CODASPY 2017.
- Code: Deep-Android-Malware-Detection (Lua, Torch)

Introduction

Target

Malware classification.

Motivation

Manually examining cannot easily scale to large numbers of applications. Many of sota methods are reliant on expert analysis to design the discriminative features.

Contribution

- propose a malware detection method that uses a convolutional network to process the raw Dalvik bytecode of an Android application, which is very efficient.
- Larger n can be performed for n-gram. No need for mannual features.

Methodology

Disassembly of Android Application

Use $baksmali$ to obtain the smali files that contain the human-readable Dalvik bytecode of the application, then extracting the opcode sequence from each method, discarding the operands.

A Dalvik bytecode example:

```
nop 00
move 01
move/from16 02
move/16 03
move-wide 04
move-wide/from16 05
move-wide/16 06
move-object 07
```

Network Architecture (CNN)

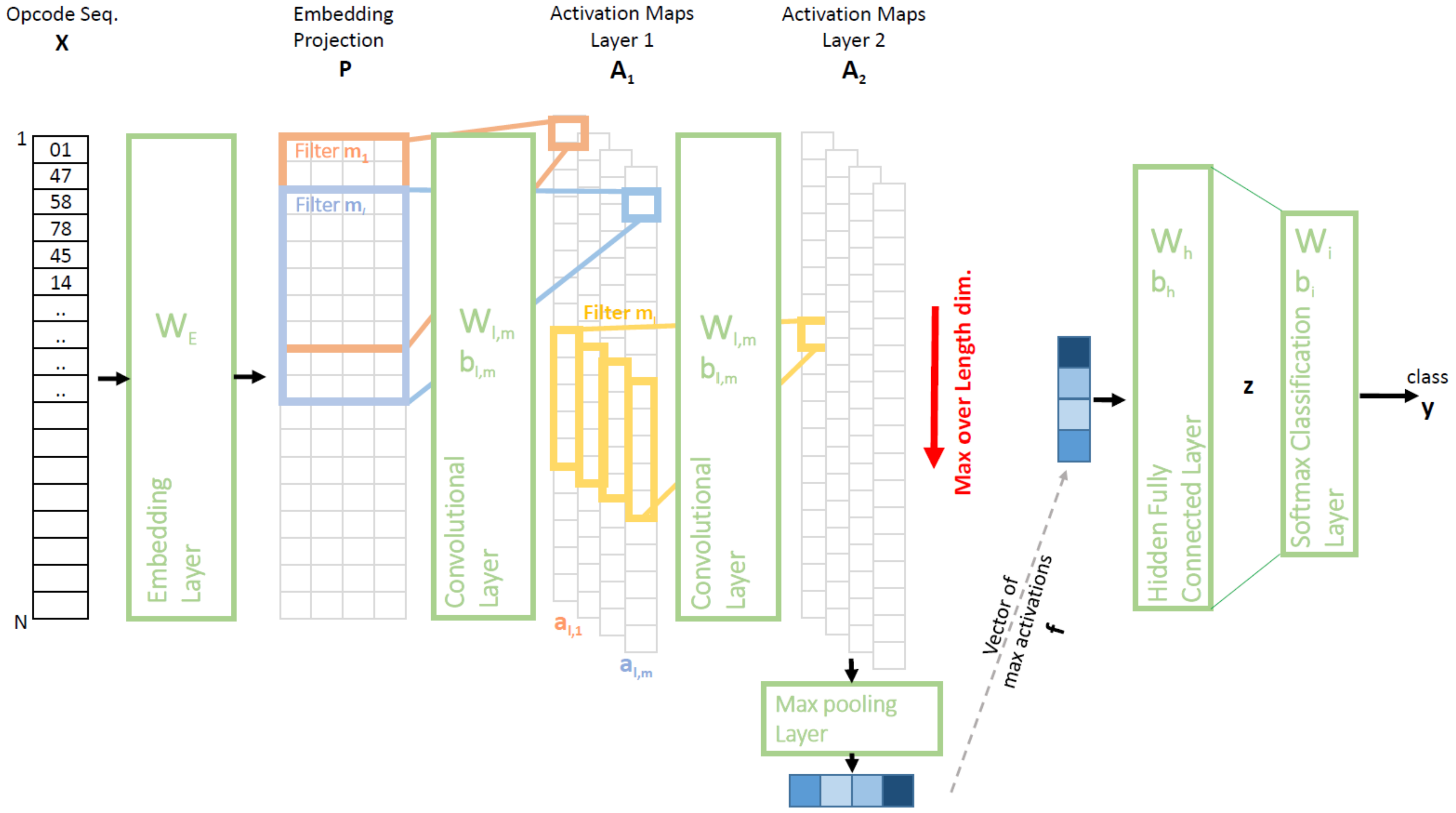

Figure 2: Malware Detection Network Architecture.

Opcode Embedding Layer

Input: a sequence of opcode instructions encoded as one-hot vectors.

Output: a dense matrix with n×p dimension.

Convolutional Layers

$$a_{l,m} = \operatorname{relu}\left(\operatorname{Conv}(P)_{W_{l,m},b_{l,m}}\right)$$

$$A_l = \left[a_{l,1}\left|a_{l,2}\right|\ldots \mid a_{l,m}\right]$$

max-pooling

$$f = \left[\max\left(a_{L,1}\right)\left|\max\left(a_{L,2}\right)\right|\ldots \mid \max\left(a_{L,m}\right)\right]$$

Classification Layers (MLP)

$$z = \operatorname{relu}\left(W_h f + b_h\right)$$

$$p(y = i \mid z) = \frac{\exp\left(w_i^T z + b_i\right)}{\sum_{i'=1}^{I} \exp\left(w_{i'}^T z + b_{i'}\right)}$$

Loss function:

$$C = -\frac{1}{b}\sum_{j=1}^{b}\sum_{i=1}^{I} 1\left\{y^{(j)} = i\right\}\log p\left(y^{(j)} = i \mid z^{(j)}\right)$$

Using RMSProp as optimizer.

$$\Theta^{(t+1)} = \Theta^{(t)} - \alpha\frac{\partial C}{\partial \Theta}$$

Experiment

Dataset: Android Malware Genome, 2 datasets provided by MacAfee.

Metrics: accuracy, precision, recall and **f-score**.

Results: performs a little bit better than sota methods, low in realistic datasets.

Other

- A demonstration of deep learning model for malware detection. More ML models can be considered.