Deep Android Malware Detection and Classification

Vinayakumar R[1], K.P Soman[1] and Prabaharan Poornachandran[2]

[1]Centre for Computational Engineering and Networking (CEN), Amrita School of Engineering, Coimbatore, Amrita Vishwa Vidyapeetham,

Amrita University, India.

[2]Center for Cyber Security Systems and Networks, Amrita School of Engineering, Amritapuri, Amrita Vishwa Vidyapeetham,

Amrita University, India.

Outline

- Introduction
- Methodology
- Description of the data set and Results
- Summary
- Future Work
- References

Introduction

- Android is the most commonly used mobile platform for smartphones and the current market leader with a market share holding nearly 87.6% [1].
- As the usage of smart phones surge past the personal computers (PC's), the malware writers also followed suit, focusing their attention creating malware for the smartphones.
- There is a sudden surge in Android malware and this sheer number of new malware instances requires newer approaches as writing signature for each malware is a daunting task.

Methodology

- Static and Dynamic analysis are the most commonly used approach.
- Static analysis collects set of features from apps by unpacking or disassembling them without the run time execution.
- Deep learning is a new field of machine learning that has the capability to obtain optimal feature representation by taking raw domain names as input [2].
- Android permissions that are collected from the static analysis are passed to recurrent neural network [3] particularly long short-term memory [4] to detect and classify the malicious apps.

Contd.

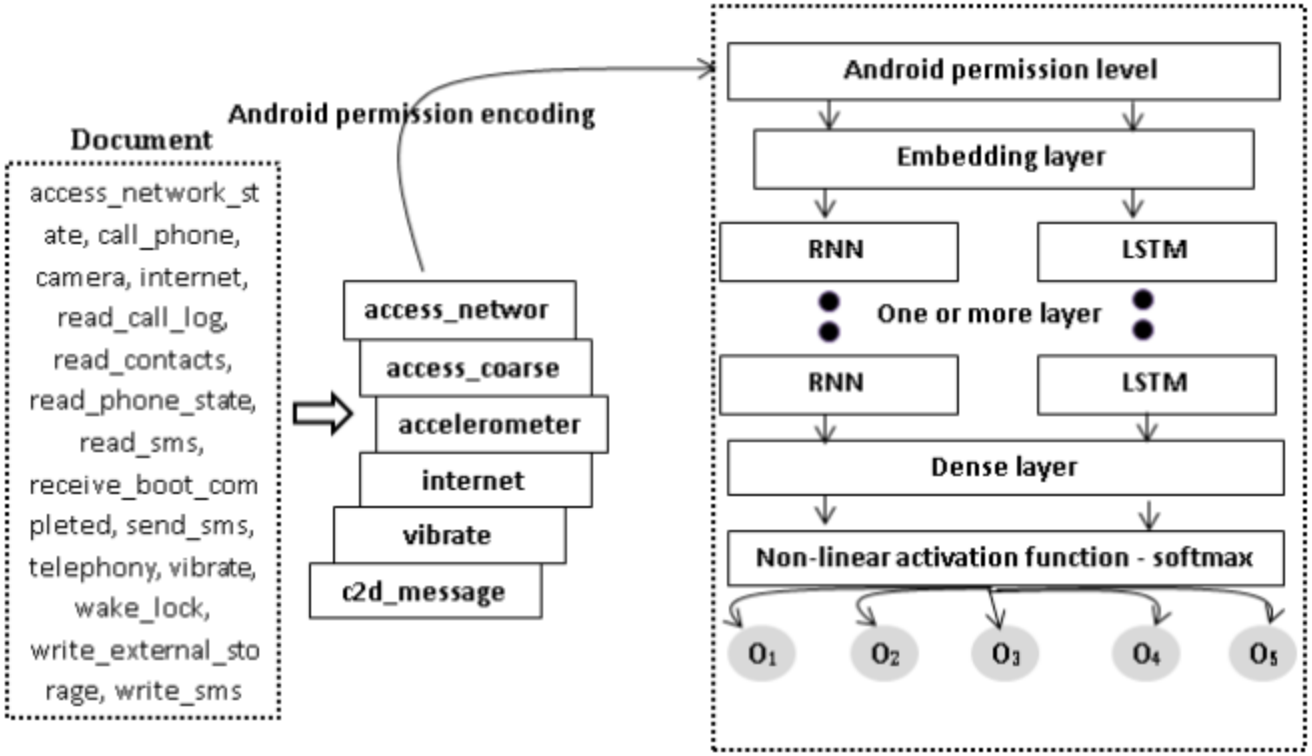

Figure 1. Proposed deep learning architecture: Maps android permissions in AndroidManifest.xml to a unique id and passed to embedding layer followed by RNN/LSTM layer and dense layer and activation layer with non-linear activation function such as sigmoid for classification.

Description of the data set and Results

The dataset is created from a set of APK (application package) files collected from the Opera Mobile Store over the period of January to September of 2014.

Table 1. Description of Data set

Total APK files	Total Permissions	Total Classes	Training samples	Testing samples
61,730	up to 583	2	30,920	30,810

Contd.

Architecture	Accuracy	Precision	Recall	F-measure
RNN 1 layer	0.981	1.000	0.977	0.988
LSTM 1 layer	0.998	0.999	0.998	0.999
RNN 2 layer	0.967	1.000	0.959	0.979
LSTM 2 layer	0.982	0.999	0.978	0.989

Table 2. 5-fold cross-validation results of RNN and LSTM networks

Contd.

Overall accuracy	Class wise accuracy	Precision	Recall	Specificity
0.897	0.897	0.91	0.96	0.628

Table 3. Summary of test results of CDMC 2016 Android malware detection using LSTM network

Summary and Future work

- LSTM network is proposed to detect and classify the malicious apps accurately by considering the Android permissions as inputs to the LSTM network.
- LSTM network has performed well in comparision to the RNN network.
- We lack in explaining the internal dynamics of LSTM network, this remained as one of significant direction towards future work.

References

[1] M Lindorfer M, M Neugschwandtner,L Weichselbaum, Y Fratantonio, V van der Veen, C Platzer, Andrubis-1,000,000 Apps Later: a view on current android malware behaviors. Third International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS), IEEE 2014 Sep 11 (pp. 3-17)

[2] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. "Deep learning." Nature 521.7553 (2015): 436-444.

[3] Elman, Jeffrey L. "Finding structure in time." Cognitive science 14.2 (1990): 179-211.

[4] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.