Chapter Five

Conclusion, Limitations and Recommendations

5.1 Conclusion of Results

1. Decision trees had the highest accuracy rate, while Naïve Bayes performance yields the lowest accuracy.

2. Applying Chi-square to the dataset ranked the columns according to their importance. A minimized version of the data was produced, thus, accuracy did not show any improvement with less columns.

3. According to Chi-square, the most important ten features are:

1. android.permission.READ_PHONE_STATE
2. android.permission.READ_SMS
3. android.permission.WRITE_SMS
4. android.permission.INTERNET
5. android.permission.ACCESS_WIFI_STATE
6. android.permission.ACCESS_NETWORK_STATE
7. android.permission.ACCESS_COARSE_LOCATION
8. android.permission.RECEIVE_BOOT_COMPLETED
9. android.permission.CHANGE_WIFI_STATE
10. android.permission.SEND_SMS

5.2 Opinions from the Field Practitioners

In order to explore the research problem from different angles that the ML analyst would not probably be much aware of, Android developers and mobile application security practitioners were being interviewed for the purpose of studying more the limitations of the proposed ML solution to the detection problem.

One Android developer for example stated that the sequence of steps for checking the security of a suspicious Android mobile application should start from the Manifest file first, followed by checking the registered components of the app, to get clear ideas of services and activities. Knowing the permissions from the same file will help in understanding what

resources are provided by the system for the app. And if the app is obfuscated, there would still be a way to read the plain xml Manifest file.

5.2 The Problem with Context

Nevertheless, relying on permissions alone is what this Android developer thinks inefficient. Another mobile security practitioner agreed on this opinion as well. The main drawback of this approach as they think is the lack of the representation of the involved context.

Analyzing an app for malicious code without having the element of “context” involved is difficult to accept. Unless the machine can know somehow the context, Android developers don’t see how the detection based on intelligence can be done.

The app can be using the permissions of combinations of permissions for legit cause. If the app for example, and sending two copies of the information to two different servers, this might be information theft.

5.3 Final Thoughts and Conclusion

Relying on Machine Learning blindly to be the only layer standing between the user and cyber attackers would not help unless the problem was approached by Android app developers and mobile application security specialists in a collaboration with the data analyst. Most of the criticisms was not targeting using ML as a technology, but on how the problem itself is represented, in terms of which data and variables are the most efficient representation of the characteristics and behaviors of malware.

Therefore, for future work, we recommend the use of datasets that contains variant variables to the used permissions. In addition to engaging domain-knowledge expert in the early stages of problem understanding, data understanding, and data preparation.