SNRU Journal of Science and Technology

Journal home page: snrujst.snru.ac.th

Malware detection for Thai Android applications using regularized logistic regression

Kamphol Promjiraprawat[1,*], Waranyu Wongseree[2]

[1]Department of Computer Engineering, Faculty of Engineering, Ramkhamhaeng University, 282 Hua Mark, Bangkapi, Bangkok 10240

[2]Department of Electrical and Computer Engineering, Faculty of Engineering, King Mongkut's University of Technology North Bangkok, 1518 Piboolsongkram Road, Bangsue, Bangkok 10800

[*]**Corresponding Author:** kamphol@ru.ac.th

Received: 6 January 2017; **Revised:** 16 March 2017; **Accepted:** 22 March 2017; **Available online:** 1 August 2017

Abstract

Android applications have widely served for this generation of smartphone customers. As a consequence, malwares have also increased and caused severe security risks. Therefore, an efficient malware detection system, especially for Android applications, has become more interesting and necessary to deal with the next generation malwares. Machine learning approach has proven its capability to identify whether an application is benign or malicious, by interpreting it as a dichotomous malware-detected classification. A permission required by each of Android application can be considered as a promising feature, despite having to take a large number of them into account. Both regularization and feature selection improve generalization performance of a classifier. In this study, Least Absolute Shrinkage and Selection Operator (LASSO) and elastic net are carried out and their performances are compared on data set of Thai Android applications and malwares. The regularized logistic regression with simultaneous feature selections provide more efficient malware detection system. The experimental results indicate that both LASSO and elastic net have their own benefits for malware classification. The LASSO with an efficient feature selection requires only 18 permissions of feature to develop the malware classification with minimum deviance and 10 permissions for a parsimonious model. The elastic net is able to detect the malware with 95% accuracy, more feature requirement notwithstanding.

Keywords: Android applications; Malware detection; LASSO; Elastic net; Regularization; Logistic regression

1. Introduction

Smartphone is an indispensable device belonging to a contemporary lifestyle with demands for the internet browsing, the social communicating, mobile banking and much more other applications. According to [1], the smartphone shipments will reach 1.80 billion units by the year 2018. Google is expected to conquer a global smartphone market having launched an Android mobile operating system which is based on Linux kernel to support various technologies of a touchscreen display. In 2016, the Android successfully dominated a mobile operating

system market with a worldwide market share of 38.30% [2]. Along with Android Studio tool which has allowed JAVA and C++ codes to be complied, a variety of Android applications or the so-called "Apps" can be developed easily and freely. Unfortunately, the more the popularity and convenience of application development, the more risk the system will deal with. McAfee found that total malwares grew around 17% from Q4 2015 to Q1 2016. With this rapid growth, there are more than 7 new threats for every second [3]. Therefore, Android smartphone system requires an automatic malware detection with various techniques that have been proposed and categorized into a signature-based detection and a machine learning approach. The signature-based detection is designed to deal with a known malware whose specific forms such as unusual traffic, malicious commands, are recognized as so-called "signature" in order to prevent similar attacks, despite worthlessness of preventing a new attack. On the other hand, the machine learning algorithm is employed to solve the malware detection problem which is defined as a binary classification model between benign application and malware. Feature selection plays an important role in determining a relevant feature. Generally, pattern recognition of Android application is a data intensive problem. An appropriate feature selection approach can mitigate not only the computational burden, but also the complexity of malware detection in a limited resource smartphone, since irrelevance features would not be taken into account. A permission required in each application has been a promising feature for malware detection, despite dealing with quite a large set of all possible permissions and a complex tool to gain desirable permissions regarding different considered applications. Several machine learning algorithms such as Bayesian classification, decision tree and support vector machine have been conducted for malware detection. The obtained solution was somewhat promising in terms of classification accuracy. However, feature selection have not been automatically analyzed in learning algorithms [4, 5].

Logistic regression is a standard classifier with a good generalization performance in machine learning approach. As regularization versions of the logistic regression, Least Absolute Shrinkage and Selection Operator (LASSO) and elastic net have been proposed to solve various classification problems with feature selection [6, 7]. LASSO offers a sparse classifier by adding a constraint of the coefficients to eliminate some of them. Elastic net retains benefits of both ridge regression and LASSO by preserving the convex optimization and allowing more features to be remained [8].

In this study, LASSO and elastic net were applied to solve the malware detection for Thai Android apps system. Various permissions required by each of the benign applications and malwares were learnt as a feature in the classification model. An optimal regularization parameter for LASSO and elastic net was selected by means of 10-fold cross-validation. Finally, the performance of both classifiers were compared by the area under the curve (AUC) of receiver operating characteristics (ROC).

2. Materials and methods

For each of the Android applications, "AndroidManifest.xml" plays an important role in description of application requirement and security. The manifest file provides essential components of the application to the Android operating system, especially scope of permission requirement. Since all applications need a particular permission

to access sensitive information and resources on smartphone in Android security protocol. Therefore, a permission required in each application is a promising feature for malware detection.

Fig. 1. depicts the process of data preparation in this study. The .apk files of benign applications and malwares were gathered and then the "Advanced Apktool" was employed to obtain "AndroidManifest.xml" for each of .apk file. The xml file has provided a tag of user-permission which is interpreted as a binary feature in the data set, as shown in Fig. 1.

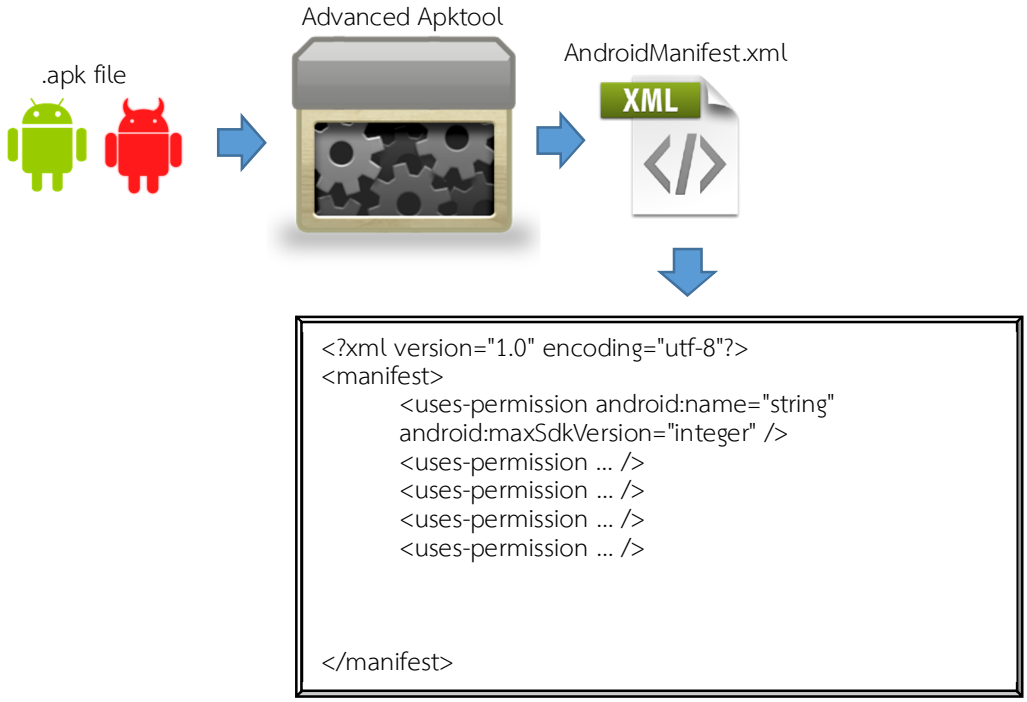

Fig. 1 Data preparation

In this study, malware detection problem was solved by the regularized logistic regression and the classification model was formulated as follows: Suppose that *N* Android applications (including both malwares and benign applications) with *M* permissions are observed. The predictor matrix, $X = \left[x_{ij} \right]_{N \times M}$, and the response vector, $y = \left[y_1, \cdots, y_N \right]^T$, are defined as follows;

$$x_{ij} = \begin{cases} 1 & ;\text{The } i^{th} \text{ application requires the } j^{th} \text{ permission} \\ 0 & ;\text{Otherwise} \end{cases}$$

$$y_i = \begin{cases} 1 & ;\text{The } i^{th} \text{ application is malware} \\ 0 & ;\text{Otherwise} \end{cases} \quad (1)$$

Therefore, the probability of the i^{th} response with respect to the logistic regression model can be defined as follows;

$$P\left(y_i=1\right)=\frac{1}{1+\mathrm{e}^{-\sum_{j=0}^{M}\beta_i x_{ij}}} \tag{2}$$

or,

$$\sum_{j=0}^{M}\beta_j x_{ij}=\ln\frac{P\left(y_i=1\right)}{1-P\left(y_i=1\right)} \tag{3}$$

where, $\beta=\left[\beta_0,\beta_1,\cdots,\beta_M\right]$ denotes the coefficient vector of logistic regression model and $x_{i0}=1$

For a positive regularization parameter, λ, the general form of the cost function for ridge regression, LASSO model and elastic net model minimizes;

$$\beta=\arg\min_{\beta}\frac{1}{N}Deviance(\beta)+\lambda\left[\frac{(1-\alpha)}{2}\|\beta\|_2^2+\alpha\|\beta\|_1\right] \tag{4}$$

where,

$$Deviance(\beta)=-2\sum_{i=1}^{N}y_i\ln\left(P\left(y_i=1\right)\right)+\left(1-y_i\right)\ln\left(1-P\left(y_i=1\right)\right)$$

$$\|\beta\|_1=\sum_{i=1}^{p}\left|\beta_j\right| \tag{5}$$

$$\|\beta\|_2^2=\sum_{i=1}^{p}\beta_j^2$$

$\|\beta\|_2$ is so-called “L2 norm” which is regularization penalty of the ridge regression by taking into account. On the other hand, LASSO penalty replaces the L2 norm by “L1 norm”, $\|\beta\|_1$, where $\alpha=1$ is adjusted. The elastic net combines both regularization penalties which are required for searching for the most appropriate α between 0 to 1 in order to construct an elastic net classifier with gaining benefits of ridge regression and LASSO model.

In this study, a 10-fold cross-validation mechanism is performed in order to select the regularization parameter. Since the malware detection entails imbalanced class between benign applications and malware, the ROC curve is used to compare the classifier performance. The ROC have been widely employed to present a trade-off between “Sensitivity” and “1 – Specificity” regarding a classifier threshold. The ROC visualization is concluded to a scalar value using the AUC. In this works, the probability density function were empirically estimated regarding classification results for each observed samples. Hence, the obtained density estimate were made use of drawing the ROC and computing the AUC which is equivalent to the Mann-Whitney U Statistic (MWUS) [9].

3. Results and Discussion

Regarding the Google Play Developer Console, all Android applications are grouped into two main categories; "Apps" and "Games". In this study, 15 application sub-categories of both were selected and reported in Table 1 [10]. The three top Thai free-downloaded apps of each sub-categories were gathered as the training set for pattern recognition of benign applications (90 Apps). 20 malware samples were observed from [11]. Regarding the observed data, 192 different permissions were taking into account as different features of the selected classifier. Therefore, this malware detection problem can be defined as a dichotomous classification problem with imbalanced data.

Table 1 Application Categories in Thai Google Play Store

Apps	Games
Art & Design, Education, Entertainment, Finance, Health & Fitness, Lifestyle, Medical, Music & Audio, News & Magazines, Personalization, Photography, Shopping, Social, Travel & Local and Weather	Action, Adventure, Arcade, Board, Card, Casino, Educational, Music, Puzzle, Racing, Role Playing, Simulation, Sports, Strategy and Others

As above mentioned in the previous section, the elastic net classifier requires finding of the most appropriate parameter, α. In this study, the minimum deviance was employed as an indicator and the experiments were repeated 1000 times. The result indicates that $\alpha = 0.039$ is the most appropriate weighting of ridge regression and LASSO.

As shown in Table 2, the elastic net outperforms LASSO model in terms of a classification accuracy, despite such an outstanding feature selection for LASSO. The minimum deviance of LASSO can be obtained by taking 18 permissions of applications into account of malware detection. On the other hand the elastic net requires much more features (92 permissions) in order to achieve the minimum deviance model. For parsimonious fitting of both models, LASSO is able to reduce the permission requirement by 56% for constructing the malware classification model which is better than the elastic net by 62%.

Table 2 Performance comparison between LASSO and elastic net

Indicator	LASSO	Elastic net
AUC	0.95	0.99
Deviance of the best model	55.31	42.61
Deviance of the parsimonious model	70.30	50.74
Accuracy	0.93	0.95
Number of variables in the best model	18	91
Number of variables in the parsimonious model	10	72

Fig. 2 describes impacts of decreasing the regularization parameter λ on the deviance. Each point in the graph refers to an average deviance corresponding to each of λ and the variance of deviance is presented in a standard error of deviance. The best model (the green dotted line) implies a model that provides the least deviance. On the other hand, the parsimonious model (the blue dotted line) offers less complexity than the best model with the satisfied deviance. In practical terms, the parsimonious model is more realistic to be deployed for malware detection in terms of both hardware and software development. The variance of deviance for both classifiers increase as the regularization parameter decrease, which leads to conclusion that higher complexity classifier is enabled for stronger feature signal.

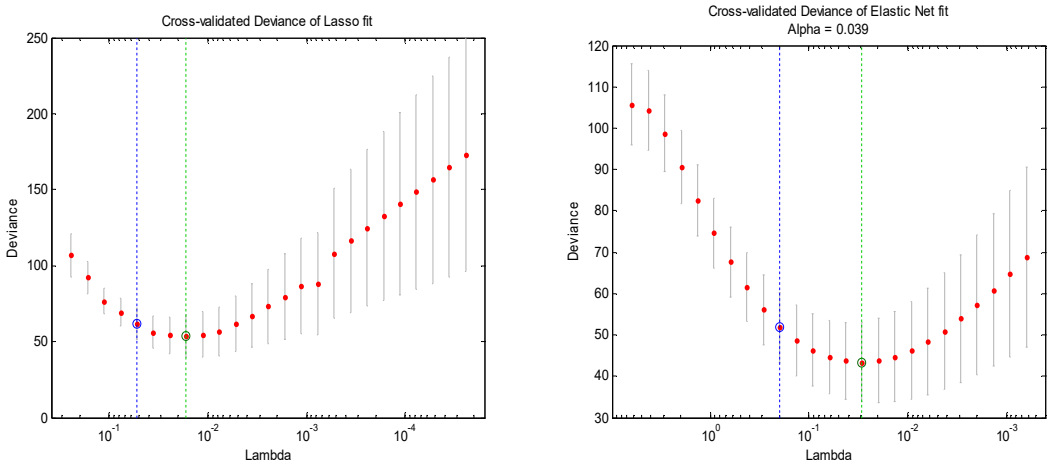

Fig. 2 Relationship between deviance and regularization parameter for LASSO and elastic net

Fig. 3. demonstrates a plot with vertical axis that presents the coefficient value determined by LASSO and elastic net. As shown in horizontal axis, the regularization parameter influences a great deal for feature selection of both the best and parsimonious models for malware classification. Feature selection schemes all begin with zero and the regularization parameter shrinks along with the greedy increment of features. Both coefficient tracks with respect to the regularization parameter are identical, nevertheless the LASSO is able to eliminate more features at the same regularization level. The selected features in elastic comprises those of ride regression and LASSO.

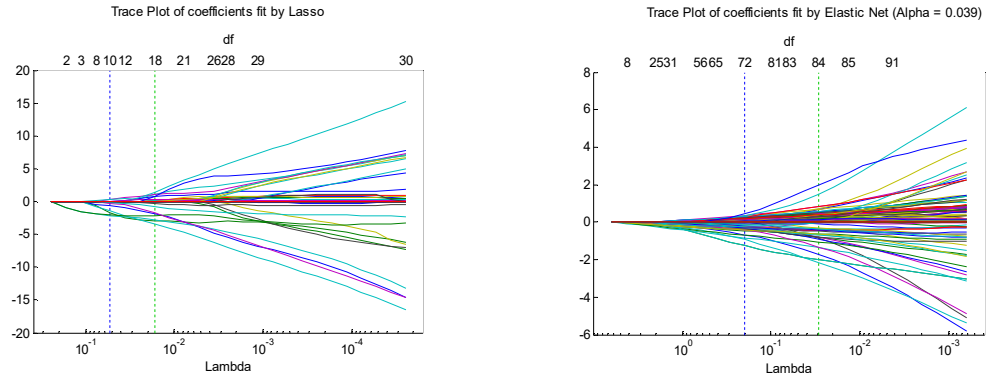

Fig. 3 Variable selection of best and parsimonious models for LASSO and elastic net

The ROC curve is able to visualize a classifier performance regardless of whether class distribution is balanced or not. The ROC curves of LASSO and elastic net are compared as shown in Fig. 4. Taking more features than the LASSO model into account, the elastic net would be superior in terms of better performance for some possible thresholds. Obviously, the AUC of the elastic net is larger. In relation to 30-times-repeated experiments, the pair t-test also indicates that elastic net was significantly better than LASSO in terms of an AUC metric. Moreover, the comparison of AUC indicates that the elastic net has higher probability of a positive instance to be randomly selected than the LASSO as well.

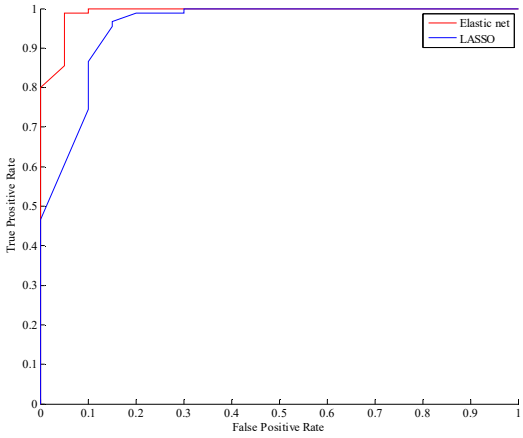

Fig. 4 ROC of LASSO and elastic net

4. Conclusion

Protection against malware which is a significant threat to Android systems plays an important role in security improvement of the next generation smartphone industry. In this article, LASSO and elastic-net, as a regularized logistic regression, have been proposed to identify malware using various permissions required by each of Android applications. In this work, malware detection was defined as an imbalanced dichotomous classification problem with 82% of benign applications.

The simulation results indicate that the LASSO provides advantages in various properties, along with the skill of determining significant permissions as well as computational performance. LASSO has suggested that efficient malware detection would take less than 10% of all permissions (192 different permissions) into account and can identify malware with a 93% accuracy. The elastic net generalizes ridge regression and LASSO simultaneously and is able to achieve an excellent classification accuracy. The elastic net has provided better AUC by 4%, compared to that of LASSO and lower deviance in both best and parsimonious models. Without an analysis of dynamic data including network traffic, memory consumption and so on, the proposed approach can be a useful alternative to improve the robustness and the efficiency of the existing malware detection system.

5. Suggestions

In this work, it has been shown that the regularization approach for a classifier can be an important tool for future malware detection. Therefore, a new approach to improve efficiency for the regularized classifier would provide great benefits in several aspects, not only for malware detection, but also a similar problem in data analysis. Moreover, more intensive experiment along with analyzing more Android permissions would help to develop malware detection tool in practice.

6. Acknowledgement

The authors would like to thank Department of Computer Engineering, Faculty of Engineering, Ramkhamhaeng University and Department of Electrical and Computer Engineering, Faculty of Engineering, King Mongkut's University of Technology North Bangkok for a facility support. The authors also would like to thank Dr Sujeetha Selvakkumaran for the manuscript reviews and valuable opinions.

7. References

[1] Global smartphone shipments forecast from 2010 to 2020, The Statistics Portal, 1 September 2016.

[2] Mobile operating systems' market share worldwide from January 2012 to June 2016, The Statistics Portal, 1 September 2016.

[3] McAfee Labs. McAfee threats report: June 2016 Technical report, McAfee, 2016.

[4] A. Feizollah, N.B. Anuar, R. Salleh, A.W.A. Wahab, A review on feature selection in mobile malware detection, Digital Investigation. 13 (2015) 22 – 37.

[5] U. Pehlivan, N. Baltaci, C. Acartürk, N. Baykal, The analysis of feature selection methods and classification algorithms in permission based Android malware detection, Computational Intelligence in Cyber Security (CICS), IEEE Symposium on IEEE, Decemeber 2014, 1 – 8.

[6] P. Filzmoser, M. Gschwandtner, V. Todorov, Review of sparse methods in regression and classification with application to chemometrics, J. Chemometrics. 26(3 – 4) (2012) 42 – 51.

[7] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society: Series B (Statistical Methodology). 67(2) (2005) 301 – 320.

[8] V. Hautamäki, T. Kinnunen, F. Sedlák, K. A. Lee, B. Ma, H. Li, Sparse classifier fusion for speaker verification, IEEE Trans. Audio, Speech, Language Process. 21(8) (2013) 1622 – 1631.

[9] T. Fawcett, An introduction to ROC analysis, Pattern Recogn. lett. 27(8) (2006) 861 – 874.

[10] Select a category for your app or game, Google Play Developer Console., https://support.google.com/googleplay/android-developer/answer/113475?hl=en#, 1 September 2016.

[11] Mobile malware mini dump, Contagio mobile, http://contagiominidump.blogspot.tw/, 1 September 2016.