Volume 2, Issue 3, March 2017 ISSN (online): 2456-0006

International Journal of Science Technology Management and Research

Available online at: www.ijstmr.com

Clustering Heterogeneous Data with k-Means for Risk assessment of Play Store Apps

Priya Malpure
Dept.of Computer Engineering
K. K.Wagh Institute of Engineering Education & Research
Nashik, India.

Komal More
Dept. of Computer Engineering
K. K.Wagh Institute of Engineering Education & Research
Nashik, India.

Vasundhara Mohite
Dept. of Computer Engineering
K. K.Wagh Institute of Engineering Education & Research
Nashik, India.

Priyanka Pawar
Dept. of Computer Engineering
K. K.Wagh Institute of Engineering Education & Research
Nashik, India

Abstract-: ***Mobile systems are facing a number of application vulnerabilities that can be combined together and utilized to penetrate systems with devastating impact. When assessing the overall security of a mobile system, it is important to assess the security risks posed by each mobile application (apps), thus gaining a stronger understanding of any vulnerabilities present. This paper aims at developing a three-layer framework that assesses the potential risks which apps introduce within the Android mobile systems. The android based platform has reached the top of the smart phone market, in the Google Apps marketplace, more than 1,192,749 applications (apps) are available and this is increasing by 40 percent monthly. Although some security tools are available it is still challenging to detect harmful apps before downloading. Framework decomposes risks into three layers namely static analysis layer, dynamic analysis layer, and behavioral analysis layer. Unsupervised feature transformation (UFT), which can transform non-numerical features into numerical features with the k-means to cluster the heterogeneous data without information loss. UFT- k-means can cluster the heterogeneous data effectively which is able to show users where the potential causes of risk exist. It will lead users to select apps without risk or with lower risk and helps to secure their system. This framework will help user to well understand the potential risks in their system. Traditional centroid-based clustering algorithms for heterogeneous data with numerical and non-numerical features result in different levels of inaccurate clustering. This is because the Hamming distance used for dissimilarity measurement of non-numerical values does not provide optimal distances between different values, and problems arise from attempts to combine the Euclidean distance and Hamming distance. In this study, the mutual information (MI)-based unsupervised feature transformation (UFT), which can transform non-numerical features into numerical features without information loss, was utilized with the conventional k-means algorithm for heterogeneous data clustering. For the original non-numerical features, UFT can provide numerical values which preserve the structure of the original non-numerical features and have the property of continuous values at the same time. Experiments and analysis of real-world datasets showed that, the integrated UFT-k-means clustering algorithm outperformed others for heterogeneous data with both numerical and non-numerical features.***

Keywords: feature transformation; k-means; clustering heterogeneous data; numerical features; non-numerical features.

I. INTRODUCTION

Mobile systems are facing a number of application vulnerabilities that can be combined together and utilized to penetrate systems with devastating impact. When assessing the overall security of a mobile system, it is important to assess the security risks

caused by each mobile applications (apps), thus gaining a stronger understanding of any vulnerabilities present. The term risk is used to denote an unsafe state or behavior of an app or system that is related to vulnerabilities [7]. In mobile systems, security assessment generally involves the risk identification, risk analysis, and risk evaluation. Most conventional clustering methods can only handle either numerical data or non-numerical data, however many real world data sets are heterogeneous, consisting of a mixture of both. As an example one of the most widely used clustering method k-means, it cannot handle heterogeneous data properly because the Euclidean distance between vectors of mixed numerical and non-numerical data cannot be measured directly. To perform heterogeneous data clustering UFT is used. In UFT non-numerical data is converted into numeric data which given as input to k-means algorithm to find out the risk value.

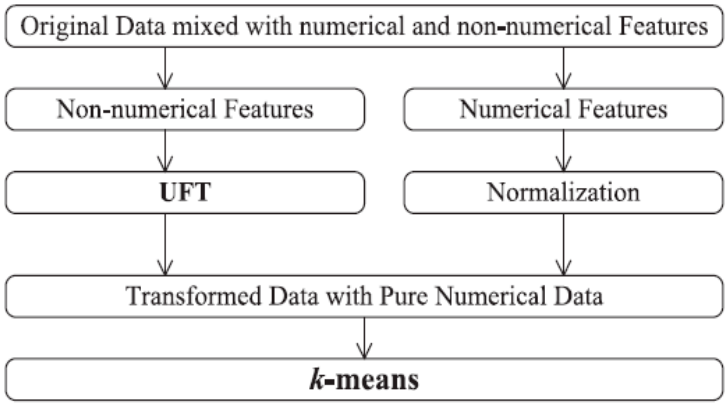

Fig.1 The experimental design of UFT-*k*-means.

A. Basic Idea:

For providing better security before downloading any mobile application, risk assessment of that app is necessary. It can be done by processing heterogeneous data using UFT. In UFT non-numerical data is converted into numeric data which further used by k-means algorithm to investigate the security risk in mobile environment and enable user to evaluate the potential risk.

B. Motivation :

Mobile malware threats have recently become a real concern. Android-based mobile systems are exposed to strong and significant security threats and many issues are reported each day. The peculiarities of mobile systems have been well known many approaches have been developed to protect them. In general, a security solution for mobile systems begins with risk assessment to determine the threats and loss expectancy. In risk assessment framework it is essential to be able to identify what each app is actually doing. A risk analysis architecture is able to accurately identify vulnerabilities, it captures the interdependencies among the vulnerabilities posed by apps and the mobile systems.

II. LITERATURE SURVEY

Multi layered Hierarchical Bayesian Network for Risk assessment in [1], the term risk is used to denote an unsafe state or behavior of an app or system that is related to vulnerabilities or threats. They proposed hierarchical Bayesian risk graph model offers a novel way to investigate the security risk in mobile environment and enables users to evaluate the potential risk. By integrating static analysis, dynamic analysis, and behavior analysis in a hierarchical framework, the risk and their propagation through each layer are well modeled by the Bayesian risk graph, which can quantitatively analyze risks faced to both apps and mobile systems [6].

Machine Learning for Android Malware Detection Using Permission and API Calls. The Google Android mobile phone platform is one of the most anticipated smartphone operating systems on the market. The open source Android platform allows developers to take full advantage of the mobile operation system, but also raises significant issues related to malicious applications (Apps).On one hand, the popularity of Android absorbs attention of most developers for producing their applications on this platform. The increased numbers of applications, on the other hand, prepares a suitable prone for some users to develop different kinds of malware

and insert them in Google Android market or other third party markets as safe applications. In this paper, we propose to combine permission and API (Application Program Interface) calls and use machine learning methods to detect malicious Android Apps.

Extensions to the k-Means Algorithm for Clustering. The k-means algorithm is well known for its efficiency in clustering large data sets. However, working only on numeric values prohibits it from being used to cluster real world data containing categorical values. In this paper we present two algorithms which extend the k-means algorithm to categorical domains and domains with mixed numeric and categorical values.

The k-modes algorithm uses a simple matching dissimilarity measure to deal with categorical objects, replaces the means of clusters with modes, and uses a frequency-based method to update modes in the clustering process to minimize the clustering cost function. With these extensions the k-modes algorithm enables the clustering of categorical data in a fashion similar to k-means. The k prototypes algorithm, through the definition of a combined dissimilarity measure further integrates the k-means and k-modes algorithms to allow for clustering objects described by mixed numeric and categorical attributes.

III. RELATED WORK

In risk assessment modeling it is essential to be able to identify what each app is actually doing. A hierarchical risk analysis architecture is able to accurately identify vulnerabilities, while a BRG can capture the interdependencies among the vulnerabilities posed by apps and the mobile systems. In this paper, hierarchical risk analysis is combined with a BRG to accurately model the risk states, propagations, and transitions.

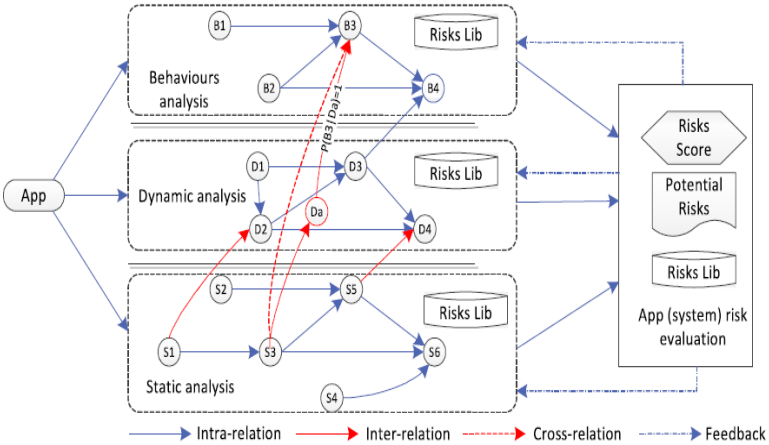

Fig.2 shows a typical HBRG model

Fig. 2 shows a typical HBRG model, which integrates hierarchical risk analysis architecture into a BRG [2]. It consists of a three-layer architecture and each layer extracts featured risks to form a directed acyclic graph (DAG). Each edge between nodes in the graph denotes the probabilistic causal dependencies. Based on the DAG, a Bayesian Network model could be created, in which each node maintains a conditional probabilities table (CPT). The parental nodes in the HBRG are assumed to be marginally independent. The relations in the HBRGs denote the transition probabilities between nodes, which are classified into three aspects: 1) intralayer relations, which connect the nodes within a layer; 2) interlayer relations, cover the connections between two adjacent layers; and 3) crosslayer relations, denoting the links which bridge between the behavior layer and static layer. In order to reduce the computation complexity, cross-layer relation can be converted into two interlayer relations by adding a virtual node concept. In Fig. 1, the cross-layer relation from $S3$ to $B3$ can be translated into two interlayer relations, where Da is the newly created virtual node and $P(B3|Da) = 1$. The virtual node Da can be seen as a copy of node $B3$ in dynamic layer without needing interrelations in the dynamic layer. By doing this, the complexity of the relation space could be significantly reduced [3].

A. Bayesian Risks Graph Model

The process that one or more vulnerabilities propagate to one or more different threats could be defined with a dependence graph. The risk states or its propagation are usually constructed as a DAG and the transitions between nodes could be modeled with local conditional

probabilities. A DAG can be modeled with a Bayesian graph model **G** = {**V**,**E**}, in which the nodes **V** denotes the variables of risks, and the edges **E** denote relations between nodes that can be described with conditional probability distributions. For each variable *Si* ∈ **V**, a conditional probability distribution *P (Si|Pa(Si))* is used to describe the transition. *Pa (Si)* denotes the parent set of *Si* in **G**. The Bayesian model reflects a conditional independence statement, which could significantly reduce the number of parameters needed [5].

IV. PROPOSED SYSTEM

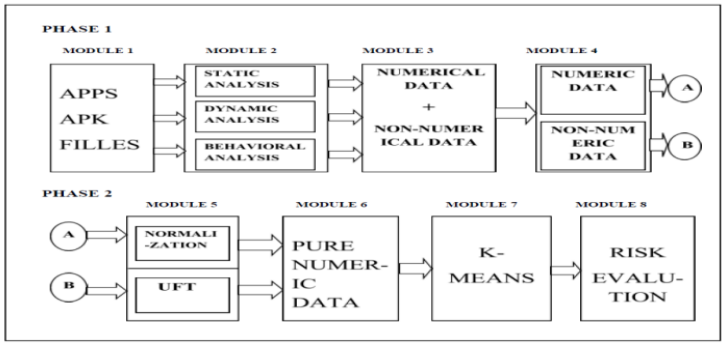

Fig 3: Decomposition of risks in apps into three layers static analysis layer

Phase 1 model decomposes risks in apps into three layers static analysis layer, dynamic analysis layer and behavioral analysis layer. Each layer focuses on specific aspects and needs. These layers helps to find the dominating risk causes. The data coming from this three layers are heterogeneous i.e. it contain the mixture of both numerical and non-numerical data. After collecting all the data, module-4 separate out it into numerical and non-numerical data then it is given for further process to phase 2. As the k-means algorithm works only on numeric data but the data coming from phase 1 contains non-numeric data also. So in phase 2 the non-numerical data is converted into numeric data by applying UFT and the numeric data is normalized using normalization technique. K-means algorithm is applied after obtaining the data from module-5.According to the types of risk the clusters are form and then finally risk evaluation is done and user get the message on screen that how much app is risky or dangerous for mobile system.

A. K-means Algorithm:

k-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other.

The next step is to take each point belonging to a given data set and associate it to the nearest center. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center. A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more. Finally, this algorithm aims at minimizing an objective function know as squared error function given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} (\|x_i - v_j\|)^2$$

where,

'$||x_i - v_j||$' is the Euclidean distance between x_i and v_j.

'c_i' is the number of data points in i^{th} cluster.

'c' is the number of cluster centers.

Algorithmic steps for k-means clustering

Let X = {$x_1,x_2,x_3,\ldots\ldots,x_n$} be the set of data points and V = {$v_1,v_2,\ldots\ldots,v_c$} be the set of centers.

1) Randomly select *'c'* cluster centers.

2) Calculate the distance between each data point and cluster centers.

3) Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers..

4) Recalculate the new cluster center using:

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i$$

where, *'c_i'* represents the number of data points in i^{th} cluster.

5) Recalculate the distance between each data point and new obtained cluster centers.

6) If no data point was reassigned then stop, otherwise repeat from step 3.

CONCLUSION AND FUTURE WORK

The system proposes a risk evaluation framework for Android mobile systems. It provide users with a friendly risk evaluation framework, which is able to show users where the potential causes of risks exist. It will lead users to select apps without risk or with lower risk and help to know how to secure their system. It is also possible that this model will allow people to well understand the potential risks in their system and create an incentive for developer to create lower-risk apps that do not contain invasive networks and avoid over-requesting permission. This paper proposed an HBRG model as a risk evaluation framework in Android mobile systems. The HBRG can provide users with a friendly risk evaluation framework, which is able to show users where the potential causes of risks exist. It will lead users to select apps with lower risk and help to know how secure their apps or systems are. It is also possible that this model will allow people to well understand the potential risks in their system and create an incentive for developer to create lower-risk apps that do not contain invasive ad networks and avoid over-requesting permissions. This paper is not the last word on the question of how to best present risk information, but future work will continue to investigate hidden risk mitigation and propagation in Android systems.

REFERENCES

[1] Hsu, C.-C.; Chen, Y.C. Mining of mixed data with application to catalog marketing. *Expert Syst. Appl.* **2007**, *32*, 12–23.

[2]. Goodall, D.W. A new similarity index based on probability. *Biometrics* **1966**, *22*, 882–907.

[3] Huang, Z. Clustering large data sets with mixed numeric and categorical values. In Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore, Singapore, 23–24 February 1997; pp. 21–34.

[4] Y. Zhou and X. Jiang, "Dissecting Android malware: Characterization and evolution," in *Proc. 33rd IEEE Symp. Security Privacy*, San Francisco, CA, USA, May 2012, pp. 95–109.

[5] Z. Wang, R. Johnson, R. Murmuria, and A. Stavrou

[6] E. Chin, A. P. Felt, V. Sekar, and D. Wagner, "Measuring user confidence in smartphone security and privacy," in *Proc. 8th Symp. Usable PrivacySecurity (SOUPS)*, Washington, DC, USA, 2012, pp. 1–16.

[7] C. S. Gates, J. Chen, N. Li, and R. W. Proctor, "Effective risk communication for Android apps," *IEEE Trans. Dependable Secure Comput.*, vol. 11, no. 3, pp. 252–266, May/Jun. 2014.