Detecting Fraud Apps in Stores

Aadit Amit Shirke[1], Sumukh Abhay Patil[2], Mrs. Sharvari Patil[3]

[1]Student, Dept. of Computer Engineering, Universal College of Engineering, Maharashtra, India
[2]Student, Dept. of Computer Engineering, Universal College of Engineering, Maharashtra, India
[3]Professor, Dept. of Computer Engineering, Universal College of Engineering, Maharashtra, India

***

Abstract - *Mobile applications are used by everyone in current day and age. These applications are generally downloaded through Play Store. Unfortunately, some of the apps are not safe. Some apps ask permissions that are not even essential for the functioning of these apps. These applications can damage our phones and there can also be data thefts. We are proposing an application which will process the information like ratings and the review of the application present in the stores using data mining. The comments/ reviews will be further analyzed using sentiment analysis. First, the comments/ reviews will be cleaned using data pre-processing. Then, we use Naïve Bayes algorithm for the sentimental analysis of these comments. This will determine whether the comment is positive or negative. If the comment turns out to be negative, the admin will be notified of these applications. The admin will then test the app manually and also gather information about the application developers, websites etc. to verify those comments. Then the admin will add remarks which will help the user to decide whether to download these applications or not.*

Key Words: Fraud, applications, reviews, comments, sentimental analysis, data mining, mobile apps.

1. INTRODUCTION

Most of us use android and IOS Mobiles these days and uses the play store or app store. Both the stores provide great number of application but unfortunately few of those applications are fraud. These applications are generally published by untrustworthy developers or individuals with harmful intentions. As we know, almost every application on the Google play store or the app store asks for various permissions in order for these apps to work. Some of these permissions are unnecessary and not even required to run these apps but they force you to allow all these permissions in order to gain access to your personal files or data. Such applications can damage our phones and there can also be data thefts. Hence, such applications must be marked, so that they are identifiable for store users. This can be done with the help of Data Mining and Sentimental Analysis to find out if the comments of a particular application are positive or negative. For this, we first use Data pre-processing using Natural Language Processing (NLP) which include procedures like Tokenization, Stop-word removal and Stemming. This cleans the data and helps in improving further steps. Then we apply Sentimental Analysis using Naïve Bayes Algorithm. We search for specific words like "fraud", "fake", "stealing data" etc. and determine how many times these particular words or phrases occur for an application and then decide whether the app is positive or negative. Also, if there exists an admin that can gather information about this application by surfing through various web sites and also gaining information about the developers, what all apps have they developed in the past, go through their terms and conditions and also testing the application to verify public reviews etc. Then the admin can add personal remarks for the application as well. This can help the users to make a better decision in whether the app is safe to download or not.

2. LITERATURE SURVEY

Detecting Fraud Apps Using Sentiment Research: It is a need to keep track and develop a system to make sure the apps present are genuine or not. The objective here is for developing a system in detecting fraud apps before the user downloads by using sentimental analysis and data mining. Sentimental analysis is used to help determining the emotional tones behind words which are expressed in online. This method is very useful in monitoring social media and helps to get a brief idea of the public's opinion on certain issues. The user cannot get correct or true reviews about the product on the internet always. We can also check for user's sentimental comments on multiple application. The reviews may be fake or genuine. Analyzing these rating and reviews together involving both user and admins comments, we can determine whether the app is genuine or not. By using this sentimental analysis and data mining, the machine is able to learn and analyze the sentiments, emotions about reviews and other texts. By use of sentimental analysis and data mining, we can analyze reviews and comments that can help to determine the correct application for both Android and iOS platforms. [1]

Forensic Analysis For the Android Applications And Detection Of Fraud Apps Using the CloudStack And Data Mining: Nowadays there are so many applications available on internet because of that user can not get correct or true reviews about the product on internet. Hence, we can check for more than two sites. The reviews may be fake on individual sites. But after comparing reviews from two sites we can get more clear idea. Hence, we can determine higher probability of getting real reviews. So here they are proposing a system to develop an android application that will take reviews from two different websites for single product and analyze them for positive negative rating. The user will give two different URLs from 2 different sites for same product to system as input for every URL Reviews and comments will be fetched separately and analyzed with NLP for positive negative rating. Then the rating will be combined together with average to give final rating for the product. The proposed system is to develop an app which will help detect fraud apps using cloudstack and data mining. To develop propose system we use two methods natural language processing and K-means algorithm. [2]

Fraud Application Detection Using Data Mining Technique: The mobile industry today is developing at a rapid rate. Since there are many apps available in market users are in fuzzy state while downloading the apps for their use. In order to advertise a particular mobile apps, the leader board of apps is the most important way in the market. An app which is at the top of the leader board leads to large number of downloads and it will gain maximum profit. So, in order to have their apps ranked as high as possible, app developers promote their apps using various ways such as advertising, offers etc. Such applications may do damage to phones and also may cause data thefts. Therefore, such applications are identified so that they will be identifiable for play store users. So, the authors of this paper are proposing an android application which will process the information, comments and three reviews of the application with natural language processing to give results. Hence, it's easier to decide fraud application [3].

3. METHODOLOGY OF PROPOSED SYSTEM

We are proposing an application which will process the information like ratings and the review of the application present in the stores using data mining. The comments/ reviews will be further analyzed using sentiment analysis. First, the comments/ reviews will be cleaned using data pre-processing. Then, we use Naïve Bayes algorithm for the sentimental analysis of these comments. This will determine whether the comment is positive or negative. If the comment turns out to be negative, the admin will be notified of these applications. The admin will then test the app manually and also gather information about the application developers, websites etc. to verify those comments. Then the admin will add remarks which will help the user to decide whether to download these applications or not.

Collecting Various App Review Dataset : First we collect Google play store dataset from the open sources like Kaggle, Google public datasets then on. The dataset contains the reviews for various application class like Social, Games, Education, Finance, News, Food.

Data Pre-Processing : In this module, we use Natural Language Processing(NLP) techniques like Tokenization, Stop word removal, Stemming to remove/ ignore unwanted text and empty spaces from the user reviews.

Sentimental Analysis Of Reviews : Here, we use algorithms like Nave Bayes to analyze sentiment in the review and categorize them as positive or negative.

Login : In this module the Admin and the user both has to login by using valid user name and password. After successfully login in, the admin can carry out certain operations.

Ranking Fraud Details : In this module, the admin will view the application details and its reviews and then personally test the application and verify if the reviews are true or not. And then add remarks for the application.

Search For App : Finally, user can enter the application name and view app details such as application name, app description, admin remarks, etc. This will help the user to decide whether to install the application or not.

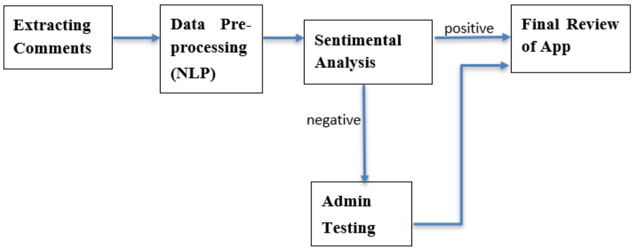


Fig -1: System Architecture

A. Data Pre-processing: This involves cleaning of data. Following are the various procedures involved in data pre-processing.

Tokenization: Tokenization means to divide a sentence into separate words.

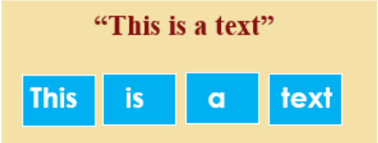


Fig -2: Tokenization

Stop word Removal: In NLP, stop-words are nothing but useless words. Few examples of stop words are: a, the, and, for etc.

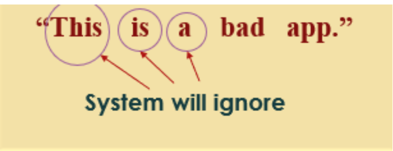


Fig -3: Stop-word Removal

Stemming; Stemming is to remove suffixes from the words and replacing it with the original words.

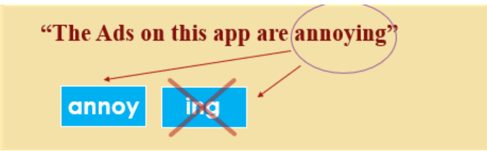


Fig -4: Stemming

B. Sentimental Analysis:

Naïve Bayes Algorithm
Step 1. Determine the data set
Step 2. Convert data set into frequency set
Step 3. Compute the prior

P(C) = Nc/N

where,
P(C) is the probability of the class.
Nc is the total count of a particular class in training set.
N is total count of the class in training set.

Step 4. Compute the conditional probabilities of each word attributes.

P(w|c) = [count(w,c)+1]/ [count(c)+|*V*|]

where,
P(w|c), is the conditional probability/ likelihood.
W, is the word attribute and c is the class.
count(w,c), is the total count of word attribute in a specific class occurring in the c class.
+1, is Laplace smoothing.
Count(c), is count of total words attribute in a class occurring in a training set.
|V|, is the total count of different word attribute.

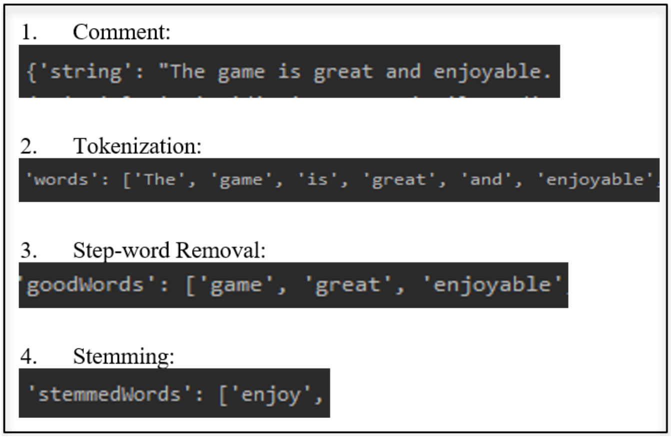
Step 5. Compute posterior probability.
$C_{MAP} = \text{argmax } P(x_1,x_2,\ldots, x_n)\ P(C),\ c\ \varepsilon\ C$

Step 6. Determine the class of the test set.

4. RESULT AND DISCUSSION

We present the output of our current progress in our proposed system. We have successfully implemented the first part of our project i.e. Data Pre-processing using Natural Language Processing (NLP). This includes processes like Tokenization, Stop-word Removal and Stemming. We have also shown how the output of the existing system looks like.

Snapshots:



1. Comment:

```
{'string': "The game is great and enjoyable.
```

2. Tokenization:

```
'words': ['The', 'game', 'is', 'great', 'and', 'enjoyable'
```

3. Step-word Removal:

```
'goodWords': ['game', 'great', 'enjoyable'
```

4. Stemming:

```
'stemmedWords': ['enjoy',
```

Fig -5: Data Pre-processing

Table -1: Comparison with Existing system

Sr No.	Factors	Existing System	Proposed System
1	Accuracy	Less	More
2	Additional information about the app	No	Yes
3	Admin remarks after manual testing	No	Yes

5. CONCLUSIONS

The result expected from the proposed system promises an effective, convenient and reliable alternative to the previous systems. Not only does it informs the user if the application is fraud or not but also gives them additional information about why exactly is the application marked as fraud, what permission does is require that are irrelevant to the functioning of the app, information about it's developers etc. This will give the user a better overall understanding of the app and help them decide whether they should download this particular application or not. Currently, we are in early stages of implementation and have successfully completed the Data Pre-processing procedure. We are currently working on sentimental analysis using Naïve Bayes algorithm and figuring out ways in which we can implement it in our project.

ACKNOWLEDGEMENT

We take this opportunity to express our deep sense of gratitude to our project guide, Mrs. Sharvari Patil, for her continuous guidance and encouragement throughout the duration of our project work. It is because of her experience and wonderful knowledge, we can fulfill the requirement of completing the project within the stipulated time. We would also like to thank our Principal Dr. J. B. Patil and the management of Universal College of Engineering, Vasai, Mumbai for providing us all the facilities and the work friendly environment. We acknowledge with thanks, the assistance provided by departmental staff, library and lab attendants.

REFERENCES

[1] Hengshu Zhu, Hui Xiong, Senior Member, IEEE, Yong Ge, and Enhong Chen, Senior Member, IEEE"detection of fraud ranking for mobile apps", IEEE Transaction and data engineering, vol 27,No 1, January 2015.

[2] Y. Ge, H. Xiong, C. Liu, and Z.-H. Zhou. A taxi driving fraud detection system. In Proceedings of the 2011 IEEE 11th International Conference on Data Mining, ICDM '11, pages 181{190, 2011.

[3] N. Spirin and J. Han. Survey on web spam detection: principles and algorithms. SIGKDD Explore. Newsl., 13(2):50{64, May 2012.

[4] K. Shi and K. Ali. Getjar mobile application recommendations with very sparse datasets. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 204{212, 2012.

[5] Mahmudur Rahman; Mizanur Rahman; Bogdan Carbunar; Duen Horng Chau. Search Rank Fraud and Malware Detection in Google Play. Published in IEEE Transactions on Knowledge and Data Engineering, pages 1329 – 1342. June 2017.

[6] Shraddha Jundhare; Padmaja Gajare; Priyanka Gadekar; Archana Aher; Shalini Wankhade. Fraud Application Detection using summary risk score. 2017 International Conference on Inventive Systems and Control ICISC, Jan 2017.