Detection of Fraud Ranking for Mobile App Using Fuzzy Logic

Esther Nowroji. S[1]
PG SCHOLER
Department of Computer Science and Engineering
Dr. N. G. P Institute of Technology
Coimbatore, India
esthernowroji@gmail.com

Vanitha. S[2]
Assistant Professor
Department of Computer Science and Engineering
Dr. N. G. P Institute of Technology
Coimbatore, India
vanitha.ngp@gmail.com

Abstract— Fraudulent activity in mobile app market means intruders or fake app developers use some techniques to increase their app rating to bring their app in top 20 list in order make more number of downloads to inflate their app sale. Knowledge engineering domain usually uses the methodologies to extract the useful knowledge from the given large data. Ongoing rapid growths of online data have created the need of KDD. Ongoing rapid growth of online rating and review system to the app, make fraud app has been launched in the mobile market and let them be downloaded and used by many users. The fraud mobile app is not worth to use and wasting device memory. Sometimes such app is created with malicious software which is harmful to the device. To avoid this situation the fraud app should be find out. In existing work, IP Address Recognition technique was used to detect the fraud ranking for single user system with dynamic user data. Though this methodology is efficient and less time consumption to detect the fraud rankings, due to advancement of the network technology many attacks techniques like IP Address spoofing and MAC Address spoofing were exist. To overcome this problem system should show the true value in detection of fraud ranking. Fuzzy Logic is form of many valued logic which produces the true value of variables. From the collected dataset, input will be fuzzified into member function. Rules are executed with input values to produce fuzzy output. Map a fuzzy output member functions into crisp output value which is used for decision. Fuzzy system produces the true value for detection of fraud ranking. Works well for large amount of data in order to increase the scalability.

Index Terms — Fuzzyfication, Defuzzyfication, member function

I. INTRODUCTION

Past view years more number of Apps has increased with their different updated version. In mobile App market, App store launch the daily App Leaderboard. Leader board shows the chart ranking of the populated Apps. App Leader board is one of the important way of promoting the mobile App.

App which posses higher rank in a Leader board will get the huge download and leads to the million dollars of revenue. So, developers has taken many way to promote their App to make their App place in a higher position in a Leaderboard to increase the App downloads. Instead of follow Traditional marketing solution, App developers fall into the deceptive activities to boost App ratings which show the drastic increase in ratings of the App in Leaderboard. App developer manipulates chart ranking on an App store. This eventual manipulation is implemented with the help of method called Human Water armies to increase the App download and ratings in a short time of periods.

There is an observation which reveals that always mobile Apps are not ranked at the high position in the Leaderboard. Leading events form the many leading sessions. In those leading events mobile Apps may have high position in the Leaderboard. Fraud ranking may happend in those leading sessions. Efficient algorithm is proposed to identify the Leading events and leading session depending on their historical records.

In the literature survey, there are many related works are available like web ranking spam detections [4], online review

spam detections [6], mobile Apps recommendation [8]. Figure 1 represents the fraud detection system. There is a problem of fraud ratings detection is not having a solution. This paper proposes the solution for detecting the fraud ratings. Instead of finding the global anomaly, challenge is to find the local anomaly to determine the fraud raking. Leading events and Leading sessions find out for getting result. Usually Leaderboard is updated daily. Leading event is time range which indicates how long App holds the rank in a Leaderboard. These different leading events form the leading session. Mining is extracting the useful information from the given collection of data. Classification is the unsupervised Approach which does not need any labeled set of data. Data aggregation is one of the important concepts in the unsupervised learning Approach.

Fuzzy logic is a form of many valued logic in that truth values of a variable will be real number 0 and 1. Fuzzy logic can be extended to handle the concept of partial truth, where the truth value is ranged between completely true and completely false. In fuzzy system, values are managed by the membership function. Fuzzifying all the input values into fuzzy membership function. Execute all application rules in the rule base to compute the fuzzy output functions. Defuzzifying the fuzzy output function to get crisp output values.

II. RELATED WORK

In mobile Application market, ranking fraud means fraudulent activities that push up Apps in popularity list. Application developer uses the shady means to increase their sales. To locate the ranking fraud, process of mining active period will take place. Leading sessions are active period. This Leading session emphasize on local anomaly not on global anomaly.

There are three evidences are investigated. They are rating based evidence, ranking based evidence, review based evidence. Investigation[1] is done by modeling the App's rating, ranking and review behavior with the help of statistical hypothesis test. Optimization based aggregation method is used to integrate these three evidences with real world App data from apple App store. To built a graph for reviews and reviewer [9] the following equation is used to determine the relationship between the two quantities x and y. The given below is called power law helpful to plot the graph for product reviews and for various reviewers.

If we take the log on both sides, we will obtain a straight line on a log-log plot. Figure 4 represents the graph. Rank aggregation used to combine [11], [14] the rankings and produce the joint ranking. Without supervision votes from the respective rankers with domain-specific expert will be aggregated. Votes of the each judge is produced by the sampling model.

Web page content spam detection is focus on these two steps. Even though Crawlers give importance to the well connected and more important page, pages are ranked in higher position by the search engine. Spam pages are numbered Approximately [4], [18], [19] according to the perception of the users. Crawler estimate the impact of web spam discard the spammed web page which has been reported. Prevalence of spam relative to number of words on page represents more than half of all the web pages contain less than 300 words in that only 12.7% of all the pages contain at 1000 words at least. Poisson Distribution of the bar graph is estimated with a mode of 2 words, a median of 281 words, and a mean of 429.2 words.

Unsupervised learning Approach for Rank Aggregation(ULARA) make the data fusion real-world task from ad hoc retrieval system. Group of shared task N participants(50) are provided with Q number of Queries(50) and return the 1000 documents[12] which is relevant to those queries. ULARA is used to combine these rankings of each group into aggregate rank function R.

Two types of rankings are used to overcome the hurdles in learning model. Depending upon the distance function definition overcome[2] the hurdles. Two types are permutation and top-k list. Further individual judges specify ranking over the k objects out of n. For example, top-10 list items could be associated with the 10 items in the first page of a result produced by the web search engine. Decomposability property will be satisfied for permutation as well as top-k list which estimate LHS efficiently.

In many Applications, rank aggregation is used like Genome Database Construction, Document Filtering, Database Middleware Construction[5], [21], [22] Spam Webpage Detection, Meta-Search. The main aim of the rank aggregation is to assigning the Real-valued score to individual entities by aggregating the every ranking provided by the base ranker. Sort the entities upon their score without change in generality. Median Rank aggregation sorts the entities upon the median of the rank in ranking list.

Hierarchical fuzzy logic systems are an active research topic in the fuzzy logic system. It is focused on reduction of the exponential number of rules in control and other applications. If N membership functions are defined for each of l inputs then inputs then the number of possible fuzzy rules is N^l. The hierarchical fuzzy system approached gains recently most interest [25]. Systems like that process inputs in lower-dimensional subspaces, combining the results in a binary tree structure. In this process, the physical interpretation and the ability to design such systems without much training is easily lost, though there are some proposals to restore it.

III. PROPOSED WORK

Due to some challenges in the existing system, fuzzy system is proposed. Existing system might be vulnerable to the IP Address spoofing attack. It does not provide any true value. This challenge can be overcome by fuzzy system whose output is truth value.

System focus on issue of shady means which means that App is promoted by fake ratings. In mobile Application market this is big issue.

A. *Fuzzy system*

In fraud detection system, App android dataset is downloaded from the internet. Dataset is processed using the predefined functions and packages in the fuzzy logic algorithm. The output of the process is crisp value.

The dataset input will be converted into the [26] fuzzy membership function output. The rules are executed with the input value which is collected from the dataset in the rule base. Finally the fuzzy output will be mapped into crisp value. This crisp value will be the 0 or 1.

System focus on issue of shady means which means that App is promoted by fake ratings. In mobile Application market this is big issue. Very few research works has been done for this issue. When an App was promoted by ranking manipulation it will be the top in the market, other users will buy the App. This activity affects other App reputation and some legal marketing campaigns, such as limited time discount.

With this modern world, many smart phone users are download Apps from internet. Money motive App developers release their App and push them to top 25 lists to increase the view and downloading to yield more money. App with ranking manipulation always has an expected ranking target and the hired marketing firms also charge money can be earned. After reaching and maintaining the expected ranking for a specified period, the manipulation [19] will be stopped and ranking of the malicious App will be decreased dramatically. As a result, the suspicious leading event may contain very short rising phase, recession phases.

Meanwhile, cost of ranking manipulation with high rating expectations is expensive due to the unclear ranking manipulation of App store and the fierce competition between App developers. The leading event of fraudulent Apps often has very short maintaining phase with high ranking positions.

Fuzzyfication algorithm and Defuzzyfication algorithm are helpful in knowledge engineering domain which uses lots of predefined methodologies to extract the useful knowledge from huge amount of data.

B. *Fuzzyfication*

The pendulum algorithm decides whether the input of the Fuzzyfication come under truth value ranges between completely true or completely false. X-Position and Y-Position value in the graphical representation is decided based on the pendulum operation procedure. Given the input is fuzzified using the pendulum algorithm. From the fuzzy logic perspective rule a conclusion is aggregated using weights and thresholds which maximizes the number of correct answers to a rule = y>a, where y is a weighted combination of rule conclusion.

Rectangular membership functions of the crisp rules are converted into the Soft trapezoidal function respective of the optimal uncertainty about 1.5%. uncertainty is sufficiently least to make the verbal interpretation of fuzzy rules still easy. The true uncertainty of psychometric scale is unknown and the reliability of the training data is hard to estimate. For single input uncertainty rules predict the one or more classes, while for large uncertainties [27] many classes have comparable probabilities. With input uncertainty set to zero crisp rules are used.

The query case in is found in the region where rules for two different classes will be overlapped. Using crisp rules, solution should be preferred to predictions of a single class only. As a result membership value of the actual case in this rule is only 38%.

```
Algorithm: Pendulum
public double getNextTheta (double d, double d_0_)

1.  double d_1_ = d - xPosition;
2.  xPosition = d;
3.  double d_2_ = d_1_ / d_0_ - xVelocity;
4.  xVelocity = d_1_ / d_0_;
5.  double d_3_ = d_2_ / d_0_;
6.  double d_4_ = d_3_ * Math.cos(previousTheta) / (mass * ropeLength);
7.  double d_5_ = 9.8 / ropeLength * Math.sin(-previousTheta);
8.  double d_6_ = d_4_ + d_5_ - damping * angularVelocity;
9.  angularVelocity += d_6_ * d_0_;
10. previousTheta += angularVelocity * d_0_;
11. if (previousTheta > 1.57)
12. previousTheta = 1.57;
13. if (previousTheta < -1.57)
14. previousTheta = -1.57;
15. return previousTheta;
```

Figure 1. Pendulum algorithm for deciding the truth values.

C. *Defuzzyfication*

```
1.  abstract public double computeCenterOfMass ()
2.  final double getMin ()
3.  return left;
4.  final double getMax ()
5.  return right;
6.  protected void clearInferenceWeights ()
7.  NumberOfWeights = 0;
8.  combinedWeight = 0;
9.  protected void addInferenceWeight (double weight)
10. combinedWeight += weight;
11. numberOfWeights++;
12. protected boolean hasWeight()
13. return numberOfWeights > 0;
14. protected double getCombinedWeights ()
15. return combinedWeight/numberOfWeights;
16. public String toString () {
17. StringBuffer b = new StringBuffer(20);
18. b.append("MF ");
```

Figure 2. Compute Center function.

Always the true will be ranged between completely true like 1 and completely false like 0. In between comparison will be done by finding the mid range value. To calculate the center value of the input value the above function is used given above in the fig.2.

D. *Membership function*

```
1.  public abstract class MemberFunction
2.  protected String name;
3.  protected double left;
4.  protected double right;
5.  protected int numberOfWeights;
6.  protected double combinedWeight;
7.  protected MemberFunction (String name, double left, double right) {
8.  if (right < left)
9.  throw new IllegalArgumentException (" lower bound ("+left+") must be
    less than upper bound ("+right+")");
10. this. left = left;
11. this. right = right;
12. this.name = name;
13. abstract public double fuzzify (double X);
14. final public String getName ()
15. return name;
```

Figure 3. Member function.

Member function is used in the universal discourse values of the input from the dataset. It is used to normalize the value to be fuzzy output value. To regulate the input value, weighting will be added to the input value to get the fuzzy output which will be mapped to the membership function. Execute all rules in the rule base.

IV. SYSTEM MODEL

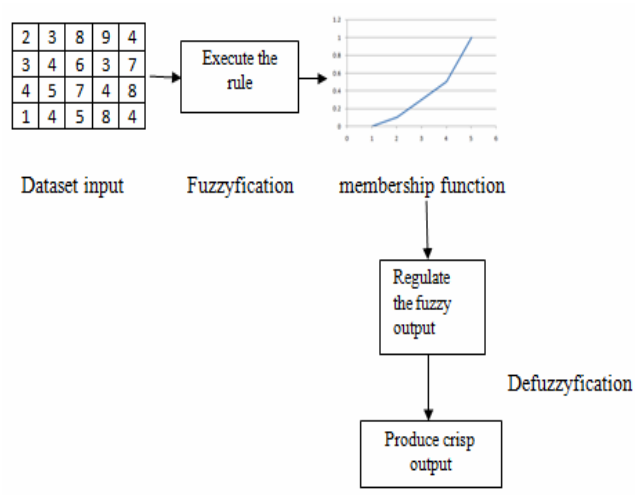

Figure 4. Overall structure of fuzzy system.

A. *Uploading the dataset*

Dataset is uploaded as a file in the fuzzy system. Dataset is saved in the drive of system through link path, particular input value for app is uploaded and processing the dataset for further operation. By default fuzzy have many predefined rules in its rule base. After the dataset is uploaded, corresponding rules are executed as defined in the above function fig. 1.

B. *Generating membership function*

Membership function is deciding the Fuzzyfication output by plotting the universal discourse value. Fuzzy logic is extended to handle the concept of partial truth, where truth value ranged between the completely true and completely false. Membership function is used to manage the linguistic variable.

C. *Produce crisp value*

Fuzzy system is the consensus of all of the input and all of the rules. Weightings can be added to each rule in the rule base and weightings can be used to regulate the degree to which a rule affects the output values. These weightings may be static or dynamic. Fuzzifying all input values into fuzzy membership functions. Execute all application rules in the rule base to compute the fuzzy output functions. Defuzzifying the fuzzy output functions to get the crisp output values.

V. EXPERIMENTAL RESULT

A. *Experimental data*

Experimental data are collected from Android app store. The data collected are around 30,000. But not those full 30,000 data are made use in project. Only 2000 data are mad use in project. The dataset is present in drive folder and selected data in dataset will be called through path then be applied with Fuzzyfication and Defuzzyfication algorithms. All methods are implemented in the NetBeans IDE 6.9.1. Input data for each app is around 350 user's ratings from dataset. Fuzzy system is worked with ten app along with its static ratings of dataset. Data can be extended to improve the scalability of the fuzzy system.

B. *Experimental Setup*

Existing system used the dynamic data provided by the user during run time. Data was stored in the database then applied with algorithm. Fuzzy system uses the static data from android app store. Nearly 35,000 user's data can be used and applied with algorithm for app in the app leaderboard. Each rating are normalized with probability value to get the fuzzy output using Fuzzyfication algorithm which contains all predefined function as inbuilt. Finally normalized value will be analyzed with graphical representation using membership

function and produce the crisp value using the Defuzzyfication which contains the predefined function as inbuilt.

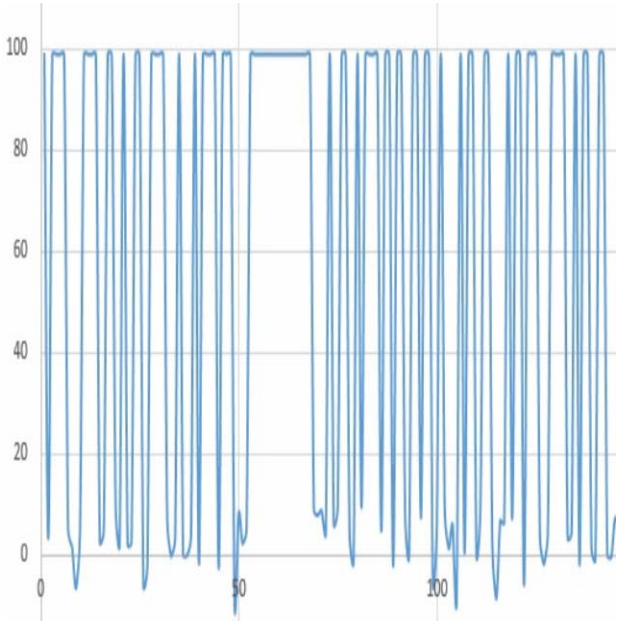

Figure 5. Fuzzy system fraud rating analysis.

From the above graphical representation fig.5 0-300 is the number of user in x-axis. 0-100 ratings provided by the each user in y-axis.

VI. CONCLUSION

Fuzzy system is mainly in the mobile app market helped to prevent download of fraud app. Each app's ratings are normalized to the probability value according to the fuzzy logic technique. Normalized probability value is defuzzified to produce the crisp value successfully. Crisp value is show casing the fraud app ratings. The probability value declared the fraud percentage of the each app in its app history.

References

[1] H. Zhu, H. Xiong, Y. Ge, and E. Chen. Ranking fraud detection for mobile Apps: A holistic view. In Proceedings of the 23rd ACM international conference on Information and knowledge management, CIKM '14, Vol 27, No. 1, Jan 2015.

[2] H. Zhu, H. Xiong, Y. Ge, and E. Chen. Ranking fraud detection for mobile Apps: A holistic view. In P roceedings of the 22nd ACM international conference on Information and knowledge management, CIKM '13, 2013.

[3] A. Klementiev, D. Roth, and K. Small. Unsupervised rank aggregation with distance-based models. In Proceedings of the 25th international conference on Machine learning, ICML '08, pages 472–479, 2008.

[4] A. Mukherjee, A. Kumar, B. Liu, J. Wang, M. Hsu, M. Castellanos, and R. Ghosh. Spotting opinion spammers using behavioral footprints. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '13, 2013.

[5] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In Proceedings of the 15th international conference on World Wide Web, WWW '06, pages 83–92, 2006.

[6] Y.-T. Liu, T.-Y. Liu, T. Qin, Z.-M. Ma, and H. Li. Supervised rank aggregation. In Proceedings of the 16th international conference on World Wide Web, WWW '07, pages 481–490, 2007.

[7] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw. Detecting product review spammers using rating behaviors. In Proceedings of the 19th ACM international conference on Information and knowledge management, CIKM '10, pages 939–948, 2010.

[8] D. M. Blei, A. Y. Ng, and M. I. Jordan. Lantent dirichlet allocation. Journal of Machine Learning Research, pages 993–1022, 2003.

[9] H. Zhu, E. Chen, K. Yu, H. Cao, H. Xiong, and J. Tian. Mining personal context-aware preferences for mobile users. In Data Mining (ICDM), 2012 IEEE 12th International Conference on, pages 1212–1217, 2012.

[10] N. Jindal and B. Liu. Opinion spam and analysis. In Proceedings of the 2008 International Conference on Web Search and Data Mining, WSDM '08, pages 219–230, 2008.

[11] G. Heinrich. Paramter stimaion for text analysis. Technical report, University of Lipzig, 2008.

[12] A. Klementiev, D. Roth, K. Small, and I. Titov. Unsupervised rank aggregation with domain-specific expertise. In Proceedings of the 21st international jont conference on Artifical intelligence, IJCAI'09, pages 1101–1106, 2009.

[13] A. Klementiev, D. Roth, and K. Small. An unsupervised learning algorithm for rank aggregation. In Proceedings of the 18th European conference on Machine Learning, ECML '07, pages 616–623, 2007.

[14] D. M. Blei, A. Y. Ng, and M. I. Jordan. Lantent dirichlet allocation. Journal of Machine Learning Research, pages 993–1022, 2003.

[15] 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.

[16] D. F. Gleich and L.-h. Lim. Rank aggregation via nuclear norm minimization. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11, pages 60–68, 2011.

[17] T. L. Griffiths and M. Steyvers. Finding scientific topics. In Proc. of National Academy of Science of the USA, pages 5228–5235, 2004.

[18] G. Shafer. A mathematical theory of evidence. 1976.

[19] 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.

[20] M. N. Volkovs and R. S. Zemel. A flexible generative model for preference aggregation. In Proceedings of the 21st international conference on World Wide Web, WWW '12, pages 479–488, 2012.

[21] Z. Wu, J. Wu, J. Cao, and D. Tao. Hysad: a semi-supervised hybrid shilling attack detector for trustworthy product recommendation. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 985–993, 2012

[22] S. Xie, G. Wang, S. Lin, and P. S. Yu. Review spam detection via temporal pattern discovery. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '12, pages 823–831, 2012.

[23] B. Yan and G. Chen. Appjoy: personalized mobile application discovery. In Proceedings of the 9th international conference on Mobile systems, applications, and services, MobiSys '11, pages 113–126, 2011.

[24] G. Heinrich. Paramter stimaion for text analysis. Technical report, University of Lipzig, 2008.

[25] A. Homaifar, E. McCormick, "Simultaneous Design of Membership Functions and Rule Sets for Fuzzy Controllers Using Genetic Algorithms", Analysis of Fuzzy system, ITOFS'95, pages 643-658, 1995.

[26] W. Duch "Uncertainty of data, fuzzy membership functions, and multi-layer perceptions" Analysis of fuzzy system: neural networks, ITONN'04, pages 1-11, 2004.

[27] J. Mendel, "Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions". NJ: Prentice Hall, 2001.