An Empirical Study of Anomaly Detection in Online Games

Phai Vu Dinh
Faculty of IT
Le Quy Don Technical University
Hanoi, Vietnam
Email: dinhphai88@gmail.com

Thanh Nguyen Nguyen
GRD Department, VNG Corporation
and Le Quy Don Technical University
Hanoi, Vietnam
Email: thanhnt@vng.com.vn

Quang Uy Nguyen
Faculty of IT
Le Quy Don Technical University
Hanoi, Vietnam
Email: quanguyhn@gmail.com

***Abstract*—In data mining, anomaly detection aims to identify the data samples that do not conform to an expected behavior. Anomaly detection has successfully been applied to many real world applications such as fraud detection for credit cards and intrusion detection in security. However, there are very little research on using anomaly detection techniques to detect cheating in online games. In this paper, we present an empirical study of anomaly detection in online games. Four unsupervised anomaly detection techniques were used to detect abnormal players. A method for evaluating the performance these detection techniques was introduced and analysed. The experiments were conducted on one artificial dataset and two real online games at VNG company. The results show the good capability of detection techniques used in this paper in detecting abnormal players in online games.**

I. Introduction

Online game is one of the most successful businesses on the Internet nowadays. However, as the online games become popular, cheating in games also grows rapidly [1]. This results in some serious impacts to the development of online game industry. Due to cheating in online games, players might receive unfair in game sets, experience unexpected advertisements or even lose money [2]. Thus, detecting and preventing cheating in online game is of great important to the growth of online game industry.

To date, there has been a large number of research that attempted to classify and detect cheating users in online games [3]. However, most cheat prevention techniques often attempt to detect if a particular cheating technique is used, and then prevent that technique. While this rule based methods can effectively prevent the known cheating techniques, they often react slowly to fast-changing cheat methods. Only recently, anomaly detection techniques were used to detect cheating users in online games [4].

In machine learning, anomaly detection approaches aim of finding samples in data that do not follow to an expected behavior [5], [6]. These samples are often referred to as anomalies or outliers. Recently, unsupervised anomaly detection techniques have been applied to detecting abnormal users in online games [4]. The results in [4] showed that anomaly detection techniques are capable of identifying cheating users in one popular online game at VNG company. However, there are several drawbacks in the previous research [4]. First, the approach for evaluating the accuracy of the detection techniques is based on using a linear classifier (logistics regression). Thus, the result may not be reliable if the data is non-linear separability. Second, the methods tested in [4] are the non-parametric models that often perform slowly. Third, the experiments was conducted on a relatively small dataset, one online game in VNG company.

978-1-4799-7492-4/15/$31.00 ©2016 IEEE

In this paper, we aim to rectify and extend the research in [4]. The main contributions of the paper are:

- We introduce the use of non-linear classifiers to evaluate the performance of unsupervised anomaly detection techniques. The experimental results show that these classifiers are better than logistic regression when using for evaluating the performance of anomaly detection approaches.
- We apply some parametric models to detecting cheating users in games. These approaches perform much faster than non-parametric models in [4] while their accuracy is also competitive.
- We test all methods on broader datasets including two online games and one artificial dataset.

The remaining of this paper is organized as follows. In the next section, we brief review some related work to cheating detection in online games. Four anomaly detection techniques examined in this paper are presented in Section III. Section III also analyses the method for evaluation the performance of unsupervised anomaly detection techniques. Section IV describes the experimental setup. The results of the experiments are presented in Section V. Section VI concludes the paper and highlights some future work.

II. Related Work

Cheating in online games is defined as *the set of activities that modify the game experience to give one player an advantage over another player(s)* [7]. This is a major problem for the game companies since it deteriorates the experience of the normal gamers and decreases their incomes [8]. Thus, detecting and stopping cheating users is critical to the development of the game companies. Recently, game developers and researchers have developed various approaches that attempt to detect and prevent cheating [9]. To date, game cheating detection techniques can be classified into three groups.

The first detection group is based on game rules. These methods attempt to define a set of legal game rules and consider all user's behavior that violates the predefined rules as cheating users [10]. There are also methods that used blacklist to detect cheating users [11]. First, they attempt to identify as many known cheating techniques as possible. After that, a matching technique is used to determine if those known cheating techniques appear in a particular game. Although, the rule based cheating detection can effectively detect the known cheating players, they often react slowly to fast-changing cheat techniques in the online game nowadays.

The second detection group is based on analysing the statistical features of game players. These methods record some important statistics of game players and report any unusual characteristic as anomalies. For instances, Chapel et al. [7] was based on probability theory and the law of large numbers to determine cheating. They assumed that each player can be assigned a score which determines the probability of the outcomes of their games, and identify cheating by observing the difference from resulting scores and expected scores. Another detection technique was proposed by Laurens et al. [12], which statistically analysed server-side behaviour of players for indications of cheating. The great advantage of statistical approaches is that they are non-intrusive to the player's privacy and can implement on all end-user system configurations. However, the statistical approaches typically reply on the assumption about the distribution of player's features. If this assumption does not hold, the performance of statistical methods may be suffered.

The third group is based on the application of anomaly detection techniques. These methods are the extension of statistical methods that apply a broader of machine learning approaches to detecting cheating players. Although, anomaly detection has been extensively applied to a wide range of problems, there is very little research of applying anomaly detection to detecting cheating in online games. To the best of our knowledge, our research in [4] was the first attempt in this research strand. The benefit of anomaly detection techniques is that they are not based on the assumption about the distribution of game data. Moreover, anomaly detection allows to identify other unusual behaviors (such as user's errors) along with cheating users. In this paper, we will extend the research in [4] by testing some novel techniques for detecting cheating users in online games. The tested techniques will be detailed in the following section.

III. Methods

This section presents four anomaly detection techniques used in this paper. After that, the approach for evaluating the accuracy of detection techniques is analysed.

A. Anomaly Detection Techniques

Since the online game data is unlabeled, we used four unsupervised anomaly detection techniques in this paper. Four tested techniques are: Local Outlier Factor, Kernel Density Estimation, K-Mean and Gaussian Mixture Model.

Local Outlier Factor (LOF): (LOF) is often considered as the baseline technique in anomaly detection [13]. The idea of LOF is to calculate the density of a sample locally instead of globally. This local density is then considered as the degree to which an object is abnormal. The detail of LOF algorithm is presented in Algorithm 1.

Algorithm 1 Process of LOF algorithm

```
1. Calculate the $k-distance$ of object $p$ ($k-distance(p)$)
   as distance between $p$ and its $k^{th}$ nearest neighbour.
2. Find the set of k-nearest neighbours of $p$
       $N_k(p) = q \subseteq D \setminus p \mid d(p,q) \leqslant k - distance(q)$
3. For each object $o \subseteq D$, calculate the reachability distance
       $rd_k(p,o) = max\{k - distance(o), d(p,o)\}$
4. Compute the local reachability density of $p$
       $l_k(p) = \left( \frac{\sum_{o \subseteq N_k(p)} rd_k(p,o)}{\mid N_k(p) \mid} \right)^{-1}$
5. Calculate the local outlier factor (LOF) value
       $LOF_k(p) = \frac{\sum_{o \subseteq N_k(p)} \frac{l_k(o)}{l_k(p)}}{\mid N_k(p) \mid}$
6. Sort objects $p$ in decreasing order of the $LOF$ value.
```

In Algorithm 1, LOF value of object p is computed as the average ratio of local reach density $l_k(p)$ and k-nearest neighbors. The ratio between $l_k(p)$ of p to those of p's k-nearest neighborsis lower meaning that the point p is far away from its nearest cluster and the higher the LOF value of p is. Therefore, the LOF value represents the degree of object being an outlier.

Kernel Density Estimation (KDE): KDE is a non-parametric method to estimate the density of data samples in a dataset [14]. A sample with low density indicates its rarity in the dataset and can be abnormal. The density of data sample x is calculated by the following equation:

$$KDE_h(x) = \frac{1}{n} \sum_p K_h(x-p) \tag{1}$$

Choosing a suitable kernel $K_h(x)$ is important for the performance of KDE. In this paper, we selected Gaussian kernel to detect abnormal users in online games.

$$K_{gaussian,h}(u) = \frac{1}{(2\pi)^{\frac{d}{2}} h^d} e^{-\frac{u^2}{2h^2}} \tag{2}$$

where h is the bandwidth of kernel; d is the dimension of data.

K-Means: K-Means is the most popular techniques used for clustering data. K-means clustering aims to partition N data samples into K clusters in which each sample belongs to the cluster with the nearest distance [15]. The algorithm starts by defining K centers and associate each point to its nearest center. After that, the K new centroids are re-calculated as the mean of the samples belong to their cluster. This process is repeated until there is no more changes in the centers. In other words, K-means aims at minimizing the objective function know as squared error function:

$$J = \sum_{i=1}^{K} \sum_{j=1}^{C_i} (||x_j - v_j||)^2 \tag{3}$$

where C_i is the number of data points in i^{th} cluster and K is the number of cluster centers.

Detecting anomaly data samples using K-means clustering replies on the assumption that normal data instances belong to large clusters, while anomalies belong to small clusters. The technique divides the dataset into a number of clusters and reports any data instance that belongs to the small size clusters as anomalous [16].

Gaussian Mixture Model (GMM): A GMM is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters [17]. A Gaussian mixture model is presented by a weighted sum of M component Gaussian densities as follows.

$$p(x|\lambda) = \sum_{1}^{M} (w_i g(x|\mu_i, \Sigma_i)) \tag{4}$$

where x is a D-dimensional data vector, w_i is the mixture weights, and $g(x|\mu_i, \Sigma_i)$ is the component Gaussian density. Each component density is a D-variate Gaussian function of the form,

$$g(x|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)} \tag{5}$$

with mean vector μ_i and covariance matrix Σ_i. The parameters of GMM ($\lambda = \{w_i, \mu_i, \Sigma_i\}$) include the mean vectors, covariance matrices and mixture weights from all component densities. A GMM model is often trained using Maximum Likelihood Estimates (MLE). The distance of a data instance to the estimated mean is then considered as the anomaly score for that instance [18].

B. Evaluation of Anomaly Detection

The evaluation of unsupervised anomaly detection has been a challenging task in the research community [19]. So far, the performance of unsupervised anomaly detection techniques has often been evaluated by using labeled data sets. In other words, the labels are not used by the algorithms during the training process, but only for evaluating their results [20]. This method is often referred to as external evaluation approach.

The shortcoming of the external methods is that they are not applicable to real world problems where the labeled data is not available. In our previous research [4] we introduced the application of classification algorithms to separate the abnormal users found by detection techniques from the normal users and consider the performance of classification algorithms as the indicator for the accuracy of anomaly detection techniques. This method was also introduced and examined by Marques et al. [21]. The method is based on the observation that abnormal samples are often distant from normal samples and can therefore be more easily separated from other observations. Thus, let S be the set of abnormal samples detected by a technique A, then the performance of A can be quantified by measuring how easy or difficult it is to separate each object in S from other objects in the dataset.

The drawback of the approach in [4] is that it used a linear classification algorithm (logistics regression) for evaluating the performance of the detection approaches. Subsequently, the comparison between various detection techniques may not be reliable if the dataset is non-linear separability. In this paper, we rectify the method in [4] by testing some non-linear classification algorithms for evaluating the performance of anomaly detection approaches. F-score was used to measure the performance of classification algorithms and this value is then considered as the indicator for the performance of anomaly detection methods. The greater value of F-score presents that the classification algorithms can performs well therefore the abnormal samples found by detection algorithms are well separated from the normal samples.

IV. Experimental Settings

We divided our experiments into two sets. The first set aims to compare and select relevant classification algorithms for measuring the performance of anomaly detection. To this end, we created an artificial dataset where normal and abnormal samples have been labeled. Four classification algorithms tested on the artificial dataset are Logistics Regression (LR), Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Random Forest (RF). Among these algorithms, LR is a linear classifier that has been used in the previous research [4]. Three other algorithms are non-linear classifiers. The implementation of the all algorithms in scikit learn software packet is used. The parameters setting for these algorithms is the default settings in scikit learn software. The details description of these algorithms and their parameters settings can be found in [22].

The second set is to evaluate the performance of various anomaly detection techniques in online games. To this end, we tested four anomaly detection techniques on two online games. Four tested anomaly detection approaches are those presented in Section III and two online games are JX2 and Chan at VNG company. They are among the most popular games at VNG company with more than one million registered users. For JX2, we used the same feature set and datasets as in [4]. For Chan game, we extracted four features from each player's record. These features are the number played games, the number of won games, the amount of money gained and lost. We also collected data from seven successive days in Chan game. The number of active players collected in a day is nearly 20000 with JX2 and about 2000 with Chan game.

The parameters setting for anomaly detection techniques is as follows. For LOF, the number of nearest neighbors used to calculate the LOF factor is set at 10. For KDE, the bandwidth of the kernel is set at 1. For K-means, we selected 15 clusters in JX2 and 6 clusters in Chan game. For GMM, we varied the number of Gaussian components from 2 to 6 and used Bayesian information criterion (BIC) [23] to select the best model. All parameters were calibrated from the early experiments for the good performance of each detection approach.

The number of abnormal users detected by three ranking methods (LOF, KDE and GMM) is set at one percent of the size of dataset. For K-means, since it is not based on ranking users, a criteria must be defined to decide if an user is anomalous. In this paper, a simple criteria in which if an user belongs to a cluster that has too few users (less than a threshold t) than this user is reported as abnormal user. In JX2 game we set $t = 80$ while in Chan game we set $t = 30$. These values of t guarantee that the number of abnormal players found by K-means is approximately equal to those found by ranking methods.

V. Results and Discussion

This section presents the result of the experiments in our paper. First, four classification techniques were verified to see whether they are reliable for evaluating the performance of unsupervised detection algorithms. After that, the effectiveness of the anomaly detection approaches is analysed.

A. Verifying Evaluation Techniques

Four classification algorithms were tested on the artificial dataset. This dataset was created so that it contains a small number of abnormal samples. The artificial dataset is visualized in Figure 1 in which two big clusters are normal sample while the samples outside these clusters are abnormal. The results of four classification algorithms on this dataset is also presented in Figure 1 where the red points are the samples that are classified as abnormal samples by the classification algorithms [1].

It can be seen from Figure 1 that LR performs unsatisfactorily on this dataset. A large number of normal samples are misclassified and become abnormal samples while nearly half of abnormal samples are not detected by LR. This is not surprising since LR is a linear classifier and can not separate the non-linear separability dataset in this figure.

In contrast to LR, three other classification algorithms perform much better. These algorithms correctly detect most abnormal samples while they do not generate any false alarm. Comparing between SVM, KNN and RF, the figure shows that KNN and RF are better than SVM. While SVM can detect most abnormal samples, it could not do so if the abnormal samples line in the area between two normal clusters. For KNN and RF, they can correctly detect all abnormal samples except one with each algorithm. Overall, the result in this section shows that the classification algorithms used for evaluating the performance of anomaly detection approaches should carefully be selected. Moreover, selecting a linear classifier like LR in [4] may lead to misleading results. Thus, in the following subsection, we only used three non-linear classifiers for evaluating the performance of the anomaly detection methods.

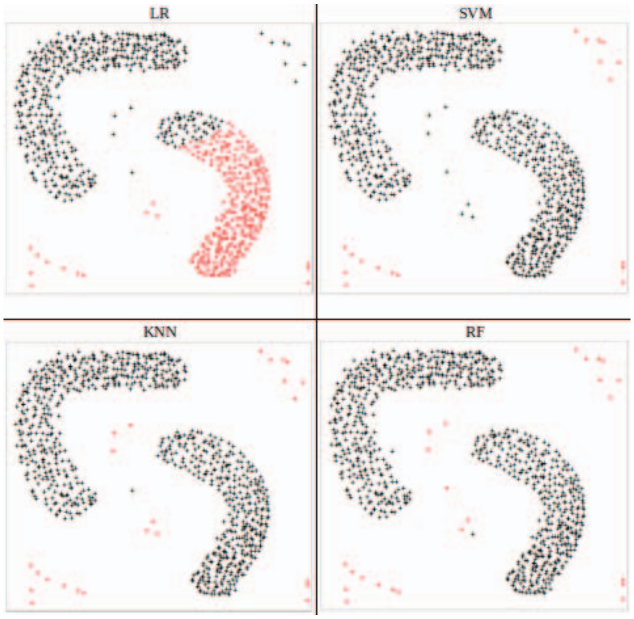

Fig. 1. Performance of four classification algorithms on the artificial dataset.

B. Performance of Anomaly Detection Techniques

The performance of anomaly detection techniques were evaluated using three metrics. The first metric is their accuracy of detecting abnormal users measured by F-score of classification algorithms applied to their output. The second metric is the number of abnormal users detected by each approach and the third metric is the computational time of the algorithms. The accuracy of four anomaly detection methods on JX2 and Chan game measured by F-score of SVM, KNN and RF is presented in Table I and Table II respectively. In these figures, the best results among four detection algorithms are printed bold faced [2].

It can be seen from these tables that all anomaly detection methods excepts LOF perform convincingly on these games. The smaller value of F-score for LOF on all configurations presents that the performance of LOF is worse than other detection approaches. The reason could be that the assumption of LOF about the local density of data is not satisfied. Comparing between KDE, K-means and GMM, the tables show that the performance of K-means and GMM is slightly better than the performance of KDE. In most experiments, the best result was often achieved by either K-means or GMM. Only in some cases, KDE obtained the best performance. However, The difference between the performance of K-means and GMM compared to KDE is only marginal.

Comparing between three evaluation techniques (SVM, KNN and RF), Table I and Table II show that they are most consistent regarding to the performance of the detection techniques. In other words, whether SVN, KNN or RF was used to evaluate, LOF is always the worst algorithm while K-means and GMM are roughly equal and they are slightly better than KDE. However, these tables also present that RF performs very well on all datasets. Subsequently, the different between F-score of four detection techniques becomes smaller

[1] We also tested these four classification algorithms on eight other datasets and their results (can be found in the site: https://bitbucket.org/qtfitmta/gameanomalydetection/downloads) are consistent with the result presented in this subsection.

[2] Eval is shorted for evaluation algorithms in these tables.

TABLE I. F-SCORE OF FOUR DETECTION METHODS ON JX2. THE BEST RESULTS ARE PRINTED BOLD FACED.

Eval	Methods	Day1	Day2	Day3	Day4	Day5	Day6	Day7
SVM	LOF	0.025	0.086	0.032	0.033	0.032	0.038	0.032
	KDE	0.853	0.563	0.670	0.764	0.597	**0.601**	**0.901**
	K-Mean	0.824	0.544	0.629	**0.814**	**0.619**	0.589	0.789
	GMM	**0.858**	**0.570**	**0.674**	0.758	0.610	0.595	0.898
KNN	LOF	0.590	0.630	0.555	0.571	0.434	0.447	0.567
	KDE	0.971	0.971	0.966	0.953	0.972	0.975	0.975
	K-Mean	**0.978**	**0.983**	**0.972**	**0.984**	**0.976**	**0.989**	**0.989**
	GMM	0.971	0.980	0.966	0.965	0.975	0.986	0.975
RF	LOF	0.974	0.974	0.972	0.959	0.964	0.973	0.974
	KDE	0.998	0.997	0.995	0.997	0.998	0.998	0.998
	K-Mean	0.998	0.998	**0.996**	**0.999**	0.998	0.998	**0.999**
	GMM	**0.999**	**0.999**	0.995	0.996	**0.999**	**0.999**	0.998

TABLE II. F-SCORE OF FOUR DETECTION METHODS ON CHAN GAME. THE BEST RESULTS ARE PRINTED BOLD FACED.

Eval	Methods	Day1	Day2	Day3	Day4	Day5	Day6	Day7
SVM	LOF	0.022	0.021	0.021	0.022	0.022	0.022	0.022
	KDE	**0.466**	0.500	0.545	0.522	0.528	0.484	0.655
	K-Mean	0.440	0.533	**0.818**	0.620	**0.561**	**0.557**	0.692
	GMM	0.429	**0.370**	0.561	**0.557**	0.442	0.538	**0.693**
KNN	LOF	0.522	0.733	0.625	0.606	0.571	0.545	0.690
	KDE	0.970	0.973	0.875	0.944	0.813	0.929	**1.000**
	K-Mean	0.955	**1.000**	0.992	0.996	0.853	**0.997**	**1.000**
	GMM	**0.989**	**1.000**	**0.997**	**0.999**	**0.995**	0.993	**1.000**
RF	LOF	0.978	0.991	0.986	0.963	0.986	0.967	0.988
	KDE	**0.993**	0.996	0.991	0.995	0.995	0.993	0.998
	K-Mean	0.989	**1.000**	**0.997**	**0.999**	0.995	0.993	**1.000**
	GMM	0.992	0.984	0.993	0.991	**0.996**	**0.995**	**1.000**

TABLE III. NUMBER OF ABNORMAL USERS DETECTED IN JX2.

Method	Day1	Day2	Day3	Day4	Day5	Day6	Day7
LOF	161	180	180	177	186	184	184
KDE	161	180	180	177	186	184	184
GMM	161	180	180	177	186	184	184
KMean	92	183	183	168	211	168	129
Ensemble1	92	131	97	131	148	139	126
Ensemble2	2	13	5	5	4	4	3

TABLE IV. NUMBER OF ABNORMAL USERS DETECTED CHAN GAME.

Method	Day1	Day2	Day3	Day4	Day5	Day6	Day7
LOF	17	18	18	18	19	15	19
KDE	17	18	18	18	19	15	19
GMM	17	18	18	18	19	15	19
KMean	14	8	12	16	11	17	18
Ensemble1	7	8	10	9	10	7	18
Ensemble2	0	0	0	0	0	0	0

if evaluated by using RF. Conversely, SVM seems not good enough to separate the abnormal users from normal users and the F-score of SVM is often smaller. Only KNN seems to be the most relevant algorithms for evaluating the performance of detection techniques among three tested classifiers.

The number of abnormal users found by each detection method on two games is presented in Table III and Table IV. In these tables we also report the number of abnormal users detected by two ensemble approaches. The first ensemble (Ensemble1) reports abnormal users if these users are detected by three methods KDE, K-means and GMM. The second ensemble method (Ensemble2) reports abnormal users if these users are considered as anomalous by three methods, LOF, K-means and GMM.

It can be seen from Table III and Table IV that the number of abnormal players detected by K-means is approximately equal to the number of abnormal users found by three ranking methods [3]. Moreover, the number of abnormal users detected by Ensemble1 is also very high and these values are only slightly less than those of four single methods. This shows that three methods, KDE, K-means and GMM found mostly the same set of abnormal users. Conversely, the number of abnormal users found by Ensemble2 is very small. This value in JX2 is often less then 10 while on Chan game it is always zero. This presents that most of the abnormal users determined by LOF is different from those of K-means and GMM. This result explains why the performance of LOF in Table I and Table II is not convincing.

TABLE V. RUNNING TIME OF FOUR DETECTION METHODS ON JX2 (MEASURED IN SECONDS).

Method	Day1	Day2	Day3	Day4	Day5	Day6	Day7
LOF	12.06	15.89	15.92	15.04	16.82	18.40	16.38
KDE	87.65	107.40	109.16	103.99	121.06	129.21	114.76
K-Mean	2.71	2.39	2.54	2.78	2.71	2.35	2.62
GMM	1.12	2.08	1.87	1.17	2.13	1.83	1.49

TABLE VI. RUNNING TIME OF FOUR DETECTION METHODS ON CHAN GAME (MEASURED IN SECONDS).

Method	Day1	Day2	Day3	Day4	Day5	Day6	Day7
LOF	0.56	0.59	0.60	0.54	0.62	0.56	0.72
KDE	1.16	1.22	1.21	1.22	1.24	0.85	1.26
K-Mean	0.15	0.14	0.12	0.14	0.14	0.11	0.13
GMM	0.31	0.35	0.39	0.29	0.30	0.33	0.32

The last metric used to evaluating the performance of anomaly detection methods is their running time. All approaches were executed on the same computer system and their execution time measured in seconds is recored and reported in Table V and Table VI. These tables show that two non-parametric methods LOF and KDE perform much slower than two parametric models, K-means and GMM. Particularly, on the problems with larger datasets, JX2, LOF is roughly 10 time slower while KDE is roughly 100 times slower than K-means and GMM. Therefore, LOF and KDE may not be applicable to online detection or to the problems where the size of dataset is very large. On the problem with smaller datasets, Chan game, LOF and KDE are still slower than K-means and GMM although the border of the difference between them is not as large as on JX2.

Overall, the results in this subsection show that the parametric detection methods (K-means and GMM) help to achieve the detection accuracy that is slightly better than the best non-parametric method, KDE, in the previous research [4]. Moreover, the parametric models also execute much faster compared to the non-parametric models. Therefore, the parametric models should be preferred to online anomaly detection or in big dataset problems.

VI. CONCLUSIONS AND FUTURE WORK

This paper presented an empirical study of anomaly detection in online games. We proposed the use of non-linear classification algorithms for evaluating the performance of unsupervised anomaly detection techniques. We tested four

[3] The number of abnormal users detected by three rank approaches are the same since in these methods, we always reported 1% users with the highest suspicious degree as anomalous.

classification algorithms on an artificial dataset. The experimental result helped to determine the relevant algorithms for evaluating the performance of anomaly detection methods.

After that, the effectiveness of four anomaly detection techniques include two non-parametric techniques in the previous paper [4] and two new parametric methods was investigated. All methods were applied to detect abnormal users in two popular games (JX2 and Chan) at VNG company. Their performance was analysed using three metrics including the detection accuracy, the number of detected abnormal users and the execution time. The experimental results showed that using the parametric methods have some advantages over the non-parametric methods in the detection accuracy and the computational time.

There are some research areas for future work which arise from this paper. First, we would like to examine and propose a better method for evaluating the performance of unsupervised abnormal detection techniques. In this paper, we followed the previous research [4], [21] in using classification algorithms to measure the accuracy of detection algorithms. However, the results in this paper showed that, various classification algorithms may leads to different results when using to evaluate the performance of abnormal detection. In the future, we would like to study and apply the internal evaluation methods in clustering algorithms to measure the performance of abnormal detection techniques.

Second, the results in this paper showed that K-means performs very convincingly on two tested online games. However, K-means has a weakness is that its performance depends on a parameter, the number of chosen cluster. In the future, we want to apply other clustering algorithms such as hierarchical clustering algorithms that help to eliminate the need of predefining the number of clusters in K-means. At the practical level, we would like to apply the tested anomaly detection methods to other games and problems to better understand their performance.

ACKNOWLEDGEMENTS

This research was funded by VNG company. The last author was partly funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.01-2014.09

REFERENCES

[1] A. B. Jeng and C. L. Lee, “A study on online game cheating and the effective defense,” in *Recent Trends in Applied Artificial Intelligence, 26th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2013, Amsterdam, The Netherlands, June 17-21, 2013. Proceedings*, vol. 7906. Springer, 2013, pp. 518–527.

[2] A. R. Kang, J. Woo, J. Park, and H. K. Kim, “Online game bot detection based on party-play log analysis,” *Computers & Mathematics with Applications*, vol. 65, no. 9, pp. 1384–1395, 2013.

[3] X. Lan, Y. Zhang, and P. Xu, “An overview on game cheating and its countermeasures,” in *Proceedings of the Second Symposium International Computer Science and Computational Technology*. ACADEMY PUBLISHER, 2009.

[4] T. T. Nguyen, A. T. Nguyen, T. A. H. Nguyen, L. T. Vu, Q. U. Nguyen, and L. D. Hai, “Unsupervised anomaly detection in online game,” in *Proceedings of the Sixth International Symposium on Information and Communication Technology, Hue City, Vietnam, December 3-4, 2015*. ACM, 2015, pp. 4–10.

[5] A. Patcha and J.-M. P. 0001, “An overview of anomaly detection techniques: Existing solutions and latest technological trends,” *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, 2007.

[6] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Computating Surveys*, vol. 41, no. 3, 2009.

[7] L. Chapel, D. Botvich, and D. Malone, “Probabilistic approaches to cheating detection in online games,” in *Proceedings of the 2010 IEEE Conference on Computational Intelligence and Games, CIG 2010, Copenhagen, Denmark, 18-21 August, 2010*, G. N. Yannakakis and J. Togelius, Eds. IEEE, 2010, pp. 195–201.

[8] S. Ferretti, “Cheating detection through game time modeling: A better way to avoid time cheats in P2P MOGs?” *Multimedia Tools Appl*, vol. 37, no. 3, pp. 339–363, 2008.

[9] J. J. Yan and B. Randell, “A systematic classification of cheating in online games,” in *NETGAMES*. ACM, 2005, pp. 1–9.

[10] M. DeLap, B. Knutsson, H. Lu, O. Sokolsky, U. Sammapun, I. Lee, and C. Tsarouchis, “Is runtime verification applicable to cheat detection?” in *Proceedings of the 3rd Workshop on Network and System Support for Games, NETGAMES 2004, Portland, Oregon, USA, August 30, 2004*, W. chang Feng, Ed. ACM, 2004, pp. 134–138.

[11] K. Huguenin, A. Yahyavi, and B. Kemme, “Cheat detection and prevention in P2P MOGs,” in *10th Annual Workshop on Network and Systems Support for Games, NetGames 2011, Ottawa, Ontario, Canada, October 6-7, 2011*, S. Shirmohammadi and C. Griwodz, Eds. IEEE, 2011, pp. 1–2.

[12] P. Laurens, R. F. Paige, P. J. Brooke, and H. Chivers, “A novel approach to the detection of cheating in multiplayer online games,” in *ICECCS*. IEEE Computer Society, 2007, pp. 97–106.

[13] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, “Lof: Identifying density-based local outliers,” in *Proc. ACM SIGMOD 2000 International Conference on Management of Data, Dalles, TX*, 2000.

[14] J. Kim and C. D. Scott, “Robust kernel density estimation,” *CoRR*, vol. abs/1107.3133, 2011.

[15] A. K. Jain and R. C. Dubes, *Algorithms for Clustering Data*. Englewood Cliffs: Prentice Hall, 1988.

[16] D. Yu, G. Sheikholeslami, and A. Zhang, “Findout: Finding outliers in very large datasets,” *Knowl. Inf. Syst*, vol. 4, no. 4, pp. 387–412, 2002.

[17] G. McLachlan and D. Peel, *Finite Mixture Models*, ser. Wiley series in probability and statistics. John Wiley & Sons, Inc., 2000, mentions MML mixture models!

[18] D. Agarwal, “Detecting anomalies in cross-classified streams: a bayesian approach,” *Knowl. Inf. Syst*, vol. 11, no. 1, pp. 29–44, 2007.

[19] A. Zimek, R. J. G. B. Campello, and J. Sander, “Ensembles for unsupervised outlier detection: challenges and research questions a position paper,” *SIGKDD Explorations*, vol. 15, no. 1, pp. 11–22, 2013.

[20] A. Zimek, E. Schubert, and H.-P. Kriegel, “A survey on unsupervised outlier detection in high-dimensional numerical data,” *Statistical Analysis and Data Mining*, vol. 5, no. 5, pp. 363–387, 2012.

[21] H. O. Marques, R. J. G. B. Campello, A. Zimek, and J. Sander, “On the internal evaluation of unsupervised outlier detection,” in *Proceedings of the 27th International Conference on Scientific and Statistical Database Management, SSDBM ’15, La Jolla, CA, USA, June 29 - July 1, 2015*, A. Gupta and S. L. Rathbun, Eds. ACM, 2015, pp. 7:1–7:12.

[22] G. Hackeling, *Mastering Machine Learning with scikit-learn*. Packt Publishing Ltd., 2014.

[23] A. Mehrjou, R. Hosseini, and B. N. Araabi, “Improved bayesian information criterion for mixture model selection,” *Pattern Recognition Letters*, vol. 69, pp. 22–27, 2016.