Chatting Pattern Based Game BOT Detection: Do They Talk Like Us?

Ah Reum Kang, Huy Kang Kim and Jiyoung Woo
Center for Information Security Technologies (CIST),
Graduate School of Information Security, Korea University,
Seoul, Rep. of Korea
[e-mail: {armk,cenda,jywoo}@korea.ac.kr]
*Corresponding author: Jiyoung Woo

Received April 9, 2012; revised July 29, 2012; revised September 7, 2012; accepted October 6, 2012; published November 30, 2012

Abstract

Among the various security threats in online games, the use of game bots is the most serious problem. Previous studies on game bot detection have proposed many methods to find out discriminable behaviors of bots from humans based on the fact that a bot's playing pattern is different from that of a human. In this paper, we look at the chatting data that reflects gamers' communication patterns and propose a communication pattern analysis framework for online game bot detection. In massive multi-user online role playing games (MMORPGs), game bots use chatting message in a different way from normal users. We derive four features; a network feature, a descriptive feature, a diversity feature and a text feature. To measure the diversity of communication patterns, we propose lightly summarized indices, which are computationally inexpensive and intuitive. For text features, we derive lexical, syntactic and semantic features from chatting contents using text mining techniques. To build the learning model for game bot detection, we test and compare three classification models: the random forest, logistic regression and lazy learning. We apply the proposed framework to AION operated by NCsoft, a leading online game company in Korea. As a result of our experiments, we found that the random forest outperforms the logistic regression and lazy learning. The model that employs the entire feature sets gives the highest performance with a precision value of 0.893 and a recall value of 0.965.

Keywords: Online game security, game bot, MMORPG, data mining, text mining

A preliminary version of this paper appeared in ICONI 2011, December 15-19, Sepang, Malaysia. This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program (NIPA-2012-H301-12-4008) supervised by the NIPA(National IT Industry Promotion Agency).

http://dx.doi.org/10.3837/tiis.2012.10.007

1. Introduction

As the online game industry grows, illegal activities in online games have drastically increased and become more diverse. Attackers on the Internet have various methods to hack online game services and gain money from it. Most of the security threats are due to the fact that game items and currency gained through game play can be sold to other players and also monetized into real currency. The major security threats in online games can be categorized into four classes: gold farming with game bots, operating a private server, system/network penetration and identity theft.

Among the various security threats in online games, the use of game bots is the most persistent threat. Game bots are automated programs that play the game on behalf of human players. Since game bots can play 24 hours a day, game bots gain cyber assets such as game money and items more efficiently than normal users. Game bots destroy the game balance and consume game contents fast. They cause honest users to feel deprived, lose interest and eventually leave the game. Thus, game bots are one of the main reasons that online game users leave and the life cycle of online games gets shortened. As the number of game bots increases, online game providers are struggling to keep their users. The efforts of game companies to prevent and detect game bots, including game monitoring, deploying security system and network monitoring, have failed to reduce the amount of cheating. These efforts can also lead to high maintenance costs and cause inconvenience for users such as collisions with other software and interruption of game play. Game companies need to adopt a bot detection method that causes no side effects for a better user experience. At the same time, the bot detection method employed should guarantee high detection accuracy.

Various methods have been proposed to detect game bots by exploiting the differences between human' behavior and bot behavior. Most previous studies on game-log analysis just focused on the analysis of playing actions or movement patterns. To the best of our knowledge, there is currently no study that attempts to analyze communication patterns as a game bot detector. Communication patterns will be able to differentiate between the bot and human player since game bots have unique and abnormal chatting patterns. They do not usually chat to avoid receiving any attention from normal users. Also, they give formal and polite responses to game masters when they get suspicions and receive a confirmation request from game masters. Typically, they deliver some command messages to control other bots in the same group. In this case, they use their own languages and repeat a limited message set.

In this study, we propose a communication pattern analysis framework for game bot detection. In Section 2, we review previous studies about the chatting pattern analysis and also provide a summary about research on game bot detection. In Section 3, we address the details of the proposed game bot detection framework based on communication pattern analysis. In Section 4, we report the experiment results of the communication pattern analysis using in-game log data from a game company. Finally, conclusions are drawn in Section 5.

2. Related Work

The research into analysis of instant messages and chat messages using text mining techniques has been widely undertaken. Bengel et al. [1] developed a text classification system for topic detection. They showed that the proposed system is effective in identifying malicious topics such as computer hacking and bomb making. Tuulos and Tirri [2] proposed a topic

classification model from chat data using a social enhanced model. Maroof [3] proposed the detection method of SPam over Instant Messaging (SPIM) and proved the effectiveness of the technique based on content-based searches and feature extraction. Hariharan [4] studied gender identification of chatters through chat conversation analysis.

The following studies proposed automatic classification models by applying text mining techniques to chat contents. Gianvecchio et al. [5] analyzed the behavior patterns of both humans and chatbots during real world chat sessions. They distinguish chatbots from humans and classify the behavior of bots into types. Classification of bots is based on the analysis of message sizes and inter-message delay times. They use an entropy-based classifier and a Bayesian-based classifier. McIntire et al. [6][7] presented a graphical and statistical analysis of communication patterns and performed studies on the behavioral patterns, message sizes and inter-message delays that distinguish chatbots from humans. Elnahrawy [8] presented a text categorization approach for automatic monitoring of chat conversations and performed a cross comparison using the Naïve Bayes, K-nearest neighbor and Support Vector Machine classifiers. Khoo and Zubek [9] mentioned that the conversational patterns of bots are their most easily recognizable characteristic. According to their arguments, this property becomes quite noticeable over an extended period of time. For example, when referring to a player, bots would use the player's full name.

Regarding game bot detection, some researchers have proposed detection methods based on the analysis of in-game logs using data mining techniques. They mainly include user behavior analysis, moving path analysis and traffic analysis. Besides these methods, human observation proof analysis and CAPTCHA (Complete Automated Public Turing Test to Tell Computers and Humans Apart) analysis are widely adopted in the game industry and are being explored by researchers.

User behavior analysis relies on the idea that there are differences between human behavior and programed bot behavior, such as their idle time or social connection [10][11]. Moving path analysis uses the fact that most bots have pre-scheduled moving paths, but humans have a greater variety in their movement patterns [12][13]. Traffic analysis uses network traffic information such as command packet timing, traffic explosiveness, network response speed, data length and traffic interval time [14]. Human observation proof analysis uses keyboard and mouse input patterns [15]. CAPTCHA analysis requests answers that can be easily solved by humans, but are hard for bots [16]. Woo et al. [17] proposed a game farmer's workshop detection method. The game farmer's workshop is a factory-size malicious group that operates numerous game bots. The authors focused on a virtual black money market that consists of gold producers, brokers, and buyers and especially the money flow between them. When combined with the game bot detection to detect gold producers that usually use game bots, the proposed model can be enhanced in revealing black money trade and finally detecting the gold farming network that constitutes the game farmer's workshop.

In our previous work [18], we proposed the party play-based bot detection method. The proposed model has a meaning that it considers social activities among gamers in game bot detection. It could detect game bots with a high precision rate, however, it cannot detect a high coverage of game bots since some bots do not participate in the party play. Because not all users do party play nor chat, the exploration on the various social activities is necessary. In the previous version of this work, we proposed the chat-based detection model of game bots and performed preliminary experiments [19]. To improve the detection performance of the previous work, we will incorporate the network feature derived from the chatting network and explore various classification algorithms on the combinations of feature sets.

We classified studies on bot detection into three categories, as shown in **Table 1**. Server-side detection methods analyze in-game logs and out-game logs. The game bots show repeated and biased patterns in their actions because they are programmed. Network-side detection methods are designed to detect the network traffic burstiness and anomalies in command timing, response time and traffic interval. The client-side method requires user involvement or security solution installation on the user-side.

Table 1. Categories of bot detection models

Taxonomy		Descriptions and related key papers
Server-side	In-game log	Combat and craft: Play pattern [10][11][18]
		Exploration: Moving path (coordinate, zone) [12][13]
		Socializing: Trading [17], Chatting [19]
	Out-game log	Windows event: Keyboard and mouse input patterns [15]
Network-side	Network traffics [14]	Command packet timing
		Traffic explosiveness
		Network response
		Data length analysis
		Traffic interval time
Client-side	Human interaction proofs (HIP)	Challenge-response test (CAPTCHA) [16]

According to the classification of user behaviors in MMORPGs into exploration, combat, craft and socializing [20], previous studies focus on explorations and combats, play patterns and movement paths. The major role of game bots is to play the game on behalf of human players. Game bots play automatically, so they show different patterns from human players. These play patterns are easy to be acquired compared to other socializing patterns. Research that looks into socializing patterns such as chatting, party play and community-based activity is lacking.

In this paper, we use in-game chat logs that have been excluded from previous work because the authors have had trouble in collecting real data from the industry. Authors can simulate bot play using bot software to get data; however, the data about socialization activity including chatting must be provided by game companies. Previous works have some limitation that their conducted dataset is not large enough to reflect plenty of social interactions, or their experiments have no validation process because of the lack of cooperation with game companies. Moreover, the analysis on chatting contents requires complex text mining skills. In this paper, the real large data about socialization activity including chatting is provided by game companies. Finally, this paper can differentiate game bots and human players through the text-mining based analysis on the communication patterns of game players.

3. Data and Text Mining Framework for Game Bot Detection

We propose a communication pattern analysis framework for online game bot detection. We pose the problem of identifying game bots as a binary classification. **Fig. 1** shows the data and text mining framework for game bot detection. First, we develop the data set combining in-game logs and chatting contents. For the entire set of users, we perform data sampling to randomly select the test data set. We then derive a well-balanced feature set and build

automatic classifiers by learning the model through the training data set. Finally, we evaluate the trained model through the test data set.

We construct the feature set from chat contents, chatters, communication methods and communication location perspectives. We extract features that can measure the diversity of communications and text features. Considering the fact that game bots do not usually chat much and communicate with a limited number of players, we design descriptive and diversity features for use in our model. Descriptive features are designed to measure the absolute frequency of chatting activities. Diversity features aim to measure how diverse chatting activities are in terms of content, chatter, communication method and communication location perspectives. The details of descriptive and diversity features are shown in **Table 2**. We count the number of messages to measure the absolute chatting volume. We then specify the volume in terms of chatters, communication methods and locations.

To measure the diversity, we adopt an information-theoretic measure, entropy [21]. It measures the uncertainty or impurity of data samples. Entropy-based measures have been proven to be good measures that numerate security vulnerabilities [22]. We measure the entropy value of the content, chatter, communication method and communication location. If the entropy is low, the likelihood of the player being a bot is high since the behavior patterns of bots are less diverse than those of human players.

We normalize the frequency to reduce the bias caused by large volume and find out the hidden patterns in the volume. For that, we design RFT indices, which are computationally inexpensive and intuitive. We measure diversity in terms of RFT. R represents the number of chat messages divided by the number of occurences of the same content. F represents the number of received messages divided by the number of receivers. T represents the number of sent messages divided by the number of senders. The entropy values and RFT are adopted to measure the variance and the normalized mean respectively.

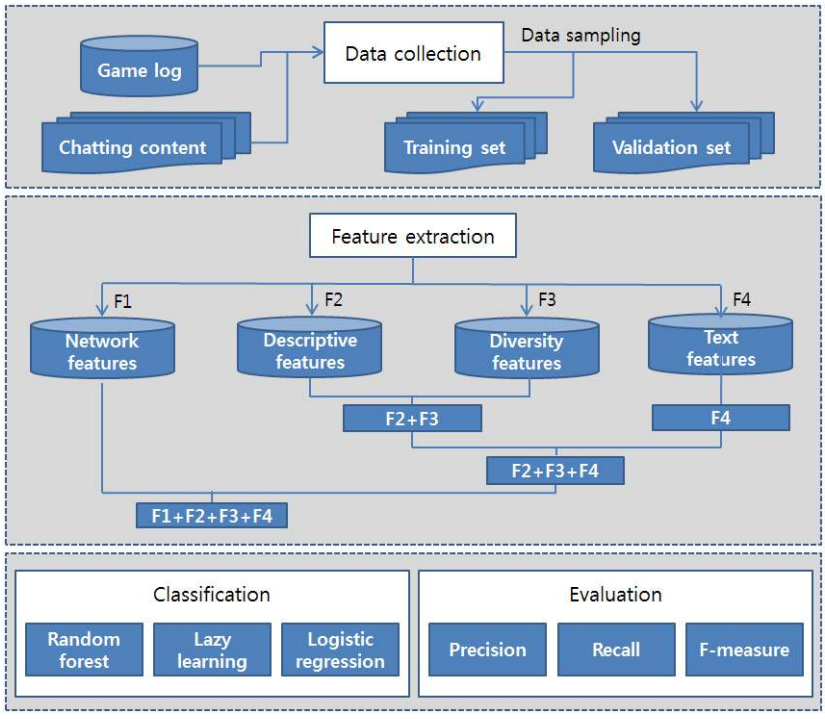

Fig. 1. Game bot detection framework

Table 2. Descriptive and diversity features

Category	Features
Content	Measures diversity of sending messages/receiving messages in terms of volume, RFT, entropy
Chatter	Measures diversity of receivers/senders in terms of volume, entropy
Communication method	Normal chat, Group chat, Alliance chat, Legion chat, Shout chat, Whisper chat, Alert chat, Channel chat
Communication location	Measures diversity of communication locations in terms of entropy

In addition, game bots' chatting contents differ from normal users' chatting contents. Game bots generate content that is repeated and difficult for humans to understand. We adopt text features to identify the difference in chatting content between human players and game bots. **Table 3** lists the text features. We define different types of lexical features in order to identify useful complex features for chatting pattern analysis. Game bots tend to repeat the same character in a word or repeat the same words in a sentence. Thus, lexical features such as the average word length and the total number of characters are used to measure lexical variations of the chatting contents of the game bot at both character and word levels [23].

We perform syntactic analysis of sentences for subjectivity detection [24]. Syntactic features, for example, the usage frequency of a function word, punctuation marks, can capture a user's chatting style at the sentence level. We can observe that game bots are more likely to speak syntactically incorrect words compared to humans. In addition, we employ content-specific features. Content-specific features are used to represent specific topics. Content-specific features are extracted based on the frequency of content-specific keywords, namely n-grams [25]. Since game bots use their own languages, they would repeat certain words that are not said frequently by humans.

Table 3. Text features

Category	Features
Lexical features	Character-based features: - Total number of characters - Total number of alphabetic characters - Total number of white-space characters - Frequency of letters - Frequency of special characters
	Word-based features: - Total number of words - Total number of characters in words - Average word length - Average sentence length in terms of word - Average sentence length in terms of character - Total number of different words
Syntactic features	- Frequency of punctuations - Frequency of function words
Content-specific features	Word-level n-grams: - Unigram, bigrams

When we construct linkages between users who communicate with each other, a chatting network is constructed. The position of users in the chatting network may also be a good identifier of game bots. To enrich the feature set we consider users' position in the network

and adopt network features. Even game bots form a group, and they usually communicate with each other. However, they only exchange messages between themselves. Therefore, it is difficult for game bots to become nodes at the center of influential neighborhoods in the chatting network that are composed of actual players. To measure this quantitatively, we analyze the difference of centrality between human players and game bots. For the centrality measure, we use betweenness centrality, closeness centrality and eigenvector centrality, which have been widely adopted in previous research. The relative importance of a node within the network can be determined by using the degree centrality, betweenness centrality, closeness centrality and eigenvector centrality [26]. The degree centrality is the number of links of a node [27]. Degree centrality is the number of other players whom a player sends or receives chat messages to or from. Betweenness is a measure for quantifying the control of a node in connecting between other nodes [28]. In a chatting network we can measure the distance between players through the shortest path between them [29]. A node is more central when its total distance is lower than those of other nodes. Closeness is a measure of how long it will take to reach from a player to all other players sequentially [30]. Eigenvector centrality is a measure of the influence of a node in the network. It assigns relative scores to all nodes. A player in the chatting network has a high value of eigenvector centrality when adjacent players are central [31].

Finally, we select several combinations of features and use them in classifiers to figure out the optimal combination of features. Each set of features is tested separately and different types of features are combined and tested in the same classifier. Classification is performed to build optimal classifiers for game bot detection. To find the model with the best performance, we compare the results of lazy learning, logistic regression and random forest. Lazy learning is a learning method where the generation of rules is delayed and performed in an adaptive way [32]. The number of bots in the entire samples is much lower than that of normal users. In this case, a lazy learning model is suitable, because the lazy learning model generates the final rule considering similar cases when the query is made. The lazy learning model revises the existing rule when a new case occurs, so the model generates a robust and adaptive rule in the case where the number of game bots is lower than that of normal users and the characteristics of the bots are diverse. Logistic regression is used for predicting the outcome of a binary dependent variable using a linear function of predictors [33]. We reviewed logistic regression to identify whether logistic regression generates a linear function that classifies users into bots and normal users through the linear combination of variables. In the random forest, a decision tree repeats on each derived subset in recursive partitioning. The decision tree considers part of variables, such as variables with high entropy, and generates the tree using them. On the other hand, random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently. Thus, we employed the ensemble decision tree that applies a decision tree repeatedly to all features.

The framework is evaluated by the users who have bot detection code recorded by internal monitoring rules of the company and the banned account lists provided by the company.

4. Experiment Results and Discussion

4.1 Experimental Configuration

We evaluate our proposed method using AION, operated by NCsoft that provides world-wide game services with well-known online games such as Lineage, Lineage II, Guild wars, AION and so on. This company maintains forty-three servers to host nearly 240,000 concurrent users

for AION. Our data from AION was collected between January 5th and 11th in 2011. During this period, there were 11,551,380 chat logs from 14,228 characters. As the ground truth, a bot user list was provided by the game company. This list was mainly built based on human observations.

The players' chatting network is shown in **Fig. 2**. The Realm vs. Realm design of the chatting network shows that there are two large separated components. By game design, the diabolic tribe cannot communicate with the heavenly tribe and vice versa. So two similar sized chatting networks are formed [34]. The nodes with large circles represent the bots detected by the internal monitoring rules or the users banned by the company. The light-colored nodes represent the players with high values of closeness centrality. The dark nodes in the middle of the network represent the players with low values of closeness centrality. As shown in **Fig. 2**, many bot nodes located around the edges rather than at the center.

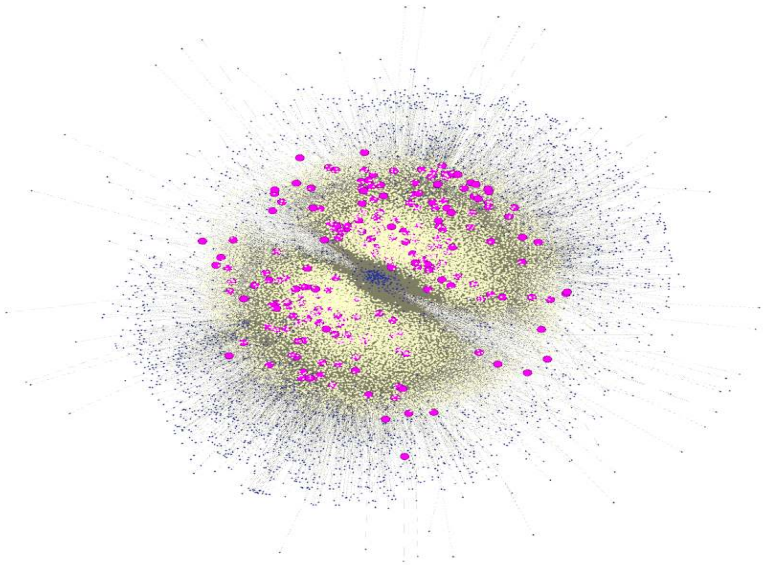

Fig. 2. Player chatting network

4.2 Evaluation Metrics

We evaluate the performance of the proposed model in terms of precision, recall and F-measure. Recall determines the percentage of declared positive cases from actual positive class. FN, TP, FP and TN represent false negatives, true positives, false positives and true negatives, respectively.

$$Preision = \frac{|TP|}{|TP| + |FP|}$$

Precision measures the percentage of positive cases accurately predicted by the classifier.

$$Recall = \frac{|TP|}{|FN| + |TP|}$$

In a classifier where the value of precision is high, the number of positive cases incorrectly classified as positive is low. As the value of recall becomes higher, the false negative error detected by the classifier decreases.

F-measure is the harmonic mean of both precision and recall, this is used to consider these two measures simultaneously.

$$F-Measure(\alpha)=\frac{|TP|}{(1-\alpha)*|FN|+|TP|+\alpha*|FP|}=\frac{Precision*Recall}{(1-\alpha)*Precision+\alpha*Recall}$$

If we want to put an emphasis on the one of the precision or recall values, we can allow different relative importance by attaching $1-\alpha$ to precision and α to recall.

In particular, when α is 0, then, no importance is attached to recall; and when α is 1, no importance is attached to precision [35]. When precision and recall are considered equally important, the α value is set at 0.5. For game bot detection, we have the following combined measure with the α set at 0.4:

$$F-Measure(0.4)=\frac{Precision*Recall}{0.6*Precision+0.4*Recall}$$

, this implies that precision is more important than recall in game bot detection. According to the game security manager, if a normal player is accused of being a game bot, he/she would become dissatisfied and some of them would demand compensation for damages. In this company, the total number of concurrent users per day can be up to 250,000. Even a small false positive ratio would generate a huge number of misjudged users.

4.3 Results Analysis

For the game bot detection, we take a discriminative approach to learn the distinction between the normal and abnormal cases and to build up automatic classifiers that automatically recognize the distinction. We split the data set into the training and test set, then build classifiers through the training data and evaluate the trained classifiers through the test data set. We used 8,228 characters as the test data and the remaining part as the training data among 14,228 characters. In addition, to avoid classifiers being over-fitted to the test data, we performed 10-fold cross validation. The cross-validation generalizes the classifier trained by the test data to the validation data. The 10-fold cross validation splits the data set into 10 groups, trains the learning model with randomly selected 9 groups and verifies the classifiers from the model with 1 group. These training and validation processes are repeated 10 times.

To evaluate the proposed framework, we compared the bot detection results from our model with internal monitoring rules and the banned account lists provided by the game company. The results of the communication pattern analysis for game bot detection are shown in **Table 4**. Three classifiers as training algorithm: random forest, lazy learning and logistic regression, are tested on four combination feature sets: (Network, Descriptive, Diversity, Text), (Descriptive, Diversity, Text), (Diversity, Text), (Text). The performances are listed in terms of the precision, recall, F-measure and running time. For the details of classification results, we presented the confusion matrix as well. In the confusion matrix, the column represents the number of instances in the predicted class while the row indicates the number of instances in the actual class. The random forest using descriptive and diversity features as well as text features outperformed other models. We identified 1,192 bots and 7,036 normal users. 1,065 bots among the bots detected by random forest turned out to be real bots. Its precision value

was 0.893, the recall value was 0.965 and the F-measure, with the emphasis on the precision (α=0.4), was 0.92.

Table 4. Evaluation results

Classifier	Features	Precision	Recall	F-Measure (α=0.4)	Time (seconds)	Human	Bot	Predicted / Actual
Random forest	Network, Descriptive, Diversity, Text	0.893	0.964	0.920	1.07	6996	128	Human
						40	1064	Bot
	Descriptive, Diversity, Text	0.893	0.965	0.920	1.62	6997	127	Human
						39	1065	Bot
	Descriptive, Diversity	0.860	0.871	0.864	0.77	6967	157	Human
						142	962	Bot
	Text	0.836	0.953	0.879	1.43	6917	207	Human
						52	1052	Bot
Lazy learning	Network, Descriptive, Diversity, Text	0.580	0.987	0.695	0.01	6336	788	Human
						14	1090	Bot
	Descriptive, Diversity, Text	0.572	0.987	0.688	0	6307	817	Human
						14	1090	Bot
	Descriptive, Diversity	0.561	0.893	0.659	0	6351	773	Human
						118	986	Bot
	Text	0.563	0.970	0.677	0	6294	830	Human
						33	1071	Bot
Logistic regression	Network, Descriptive, Diversity, Text	0.687	0.213	0.363	1.9	7017	107	Human
						869	235	Bot
	Descriptive, Diversity, Text	0.685	0.209	0.358	1.83	7018	106	Human
						873	231	Bot
	Descriptive, Diversity	0.731	0.185	0.335	0.17	7049	75	Human
						900	204	Bot
	Text	0.579	0.030	0.070	1.26	7100	24	Human
						1071	33	Bot

As shown in **Table 4**, the random forest outperforms lazy learning and logistic regression. According to the result of logistic regression, we noticed that it may be difficult to generate the linear function that classifies bots and normal users with selected features. In particular, it is almost impossible to generate a linear function using text features. Random forest and lazy

learning generated rules using all variables, so they showed better performance than logistic regression considering part of the variables. This result demonstrates that classifiers utilizing the full feature set have better performances than the classifier using just a part of the variables in the feature set. To sum up, the random forest is more suitable for chatting pattern classification than lazy learning and logistic regression for bot detection. Our experimental evaluation shows good performance that is effective in detecting game bots.

We compared the experiments results of the proposed method with those of our previous works. We analyzed the data from different periods since we had trouble in collecting data for the same period. The bot detection method using the trade log presents a precision value of 0.39 with data between April 10th and May 9th in 2010 [17]. This works focused on detection of gold farmer's workshop. The workshop consists of gold producers, brokers and buyers. The gold producers usually use game bots, but some of them use cheap laborers. Since the workshop does not consist solely of game bots, the game bot detection through this method naturally has a low precision. The party log based detection method shows a precision value of 0.9592 and a recall value of 0.113 with data between April 10th and 17th in 2010 [18]. The low recall value is due to the fact that not all the game bots perform party play. In our previous work, decision tree shows a precision value of 0.6699 and a recall value of 0.4696, and lazy learning shows a precision value of 0.523 and a recall value of 0.89 with data between January 5th and 11th in 2011 [19]. In this paper, we derived better results that the chatting log based detection method with random forest presents a precision value of 0.893 and a recall value of 0.965 with data between January 5th and 11th in 2011.

In addition, we compared the experiments results of the proposed method with those of representative methods in previous works. Most studies use the small sized and simulated data. They generated the simulated data by operating the game bot in personal. Even studies that use real data employed the small sized data or randomly selected data. The previous works have limitations that their data are lack to reflect the real situation. Our studies are based on the large-scale real data provided by a major company. When we compare the performances of previous works that adopt the real data, our method outperformed Thawonmas et al's work [10] in terms of the recall and finally f-measure. Our method also outperformed Chen and Hong's work [11] in terms of the accuracy. Our method outputs the highest accuracy of 0.98 with the random forest.

Table 5. Performances of representative previous works

Data source	Data type	# of data	Precision	Recall	Accuracy	TN	TP	Game name
Action frequency [10]	Real data	7 bots, 7 users	0.94~0.95	0.24 ~0.43	-	-	-	Cabal Online
Idle time [11]	Real data	287 players	-	-	Higher than 0.95	-	-	Angel's Love
XY coordinates, angle of the movement [12]	Simulated data	25 bots, 25 users	-	-	-	-	-	Ragnarok Online
Movement repetition [13]	Simulated data	2 bots, 10 users	-	-	-	-	-	World of Warcraft
Traffic patterns [14]	Simulated data	11 bots, 8 users	-	-	0.9~0.95	-	-	Ragnarok Online
Keyboard and mouse input patterns [15]	Simulated data	10 bots, 30 users	-	-	0.99	0.976 ~1	0.624 ~1	World of Warcraft

5. Conclusion

In this paper, we proposed a communication pattern analysis framework for game bot detection. We used chatting data that reflects gamers' communication patterns. In MMORPGs, game bots use chatting functions in a different way to normal users. We derived four types of feature: a network feature, a descriptive variable, a diversity measure and a text feature. To build the learning model for game bot detection, we tested and compared three classification models: lazy learning, logistic regression and random forest. We applied the proposed detection model to AION, the second most popular game in the world. As a result of our experiment, random forest is more suitable for chatting pattern classification than lazy learning and logistic regression.

Our study is the first research to adopt chatting patterns for game bot detection to the best of our knowledge. We employed chat content specific features and diversity measure features. We used text mining techniques to derive the content specific features. We introduced lightly summarized indices to deal with large scale data in real time. We expect that the proposed model will perform better when we add our observations of the game bot list since the baseline for game bot detection is mainly built based on human observation of game play through the game masters in the company. In addition, there exists much room for improvement in game bot detection based on user communication patterns. Other factors such as game play patterns should be incorporated in the proposed model to obtain higher performance.

References

[1] J. Bengel, S. Gauch, E. Mittur and R. Vijayaraghavan, "Chattrack: Chat Room Topic Detection using Classification," *Computer Science Intelligence and Security Informatics*, vol. 3073, pp. 266-277, 2004. Article (CrossRef Link)

[2] V. H. Tuulos and H. Tirri, "Combining Topic Models and Social Networks for Chat Data Mining," *in Proc. of Int. Conf. on on Web Intelligence IEEE Computer Society*, pp. 206-213, 2004. Article (CrossRef Link)

[3] U. Maroof, "Analysis and Detection of SPIM using Message Statistics," in *Proc. of 6th Int. Conf. on Emerging Technologies (ICET)*, pp. 246-249, 2010. Article (CrossRef Link)

[4] S. Hariharan, "Gender Prediction in Chat based Medium's Using Text Mining," *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, vol. 1, no. 1, pp. 18-22, 2011. Article (CrossRef Link)

[5] S. Gianvecchio, M. Xie, Z. Wu and H. Wang, "Humans and Bots in Internet Chat: Measurement, Analysis, and Automated Classification," *IEEE/ACM Transactions on Networking*, vol. 19, no. 5, pp. 1557-1571, 2011. Article (CrossRef Link)

[6] J. McIntire, P. Havig, K. Farris and L. McIntire, "Graphical and Statistical Communication Patterns of Automated Conversational Agents in Collaborative Computer-Mediated Communication Systems," in *Proc. of IEEE Natl Conf. on Aerospace and Electronics Conference (NAECON)*, pp. 34-40, 2010. Article (CrossRef Link)

[7] J. P. McIntire, L. K. McIntire and P. R. Havig, "Methods for Chatbot Detection in Distributed Text-based Communications," in *Proc. of Int. Symposium on Collaborative Technologies and Systems (CTS), IEEE*, pp. 463-472, 2010. Article (CrossRef Link)

[8] E. Elnahrawy, "Log-based Chat Room Monitoring using Text Categorization: A Comparative Study," in *Proc. of Int. Conf. on Information and Knowledge Sharing (IKS)*, 2002. Article (CrossRef Link)

[9] A. Khoo and R. Zubek, "Applying Inexpensive AI Techniques to Computer Games," *Intelligent Systems, IEEE*, vol. 17, no. 4, pp. 48-53, 2002. Article (CrossRef Link)

[10] R. Thawonmas, Y. Kashifuji and K. T. Chen, "Detection of MMORPG Bots based on Behavior Analysis," in *Proc. of Int. Conf. on Advances in Computer Entertainment Technology*, pp. 91-94, 2008. Article (CrossRef Link)
[11] K. T. Chen and L. W. Hong, "User Identification based on Game-Play Activity Patterns," in *Proc. of 11th Int. Workshop on Digital Signal Processing*, pp. 246-248, 2008. Article (CrossRef Link)
[12] M. van Kesteren, J. Langevoort and F. Grootjen, "A Step in the Right Direction: Bot Detection in MMORPG using Movement Analysis," in *Proc. of 21th Belgian-Dutch Conf. on Artificial Intelligence*, 2009. Article (CrossRef Link)
[13] S. Mitterhofer, C. Kruegel, E. Kirda and C. Platzer, "Server-Side Bot Detection in Massively Multiplayer Online Games," *IEEE Security and Privacy*, vol. 7, no. 3, pp. 29-36, 2009. Article (CrossRef Link)
[14] K. T. Chen, J. W. Jiang, P. Huang, H. H. Chu, C. L. Lei and W. C. Chen, "Identifying MMORPG Bots: A Traffic Analysis Approach," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, 2009. Article (CrossRef Link)
[15] S. Gianvecchio, Z. Wu, M. Xie and H. Wang, "Battle of Botcraft: Fighting Bots in Online Games with Human Observational Proofs," in *Proc. of 16th ACM Conf. on Computer and Communications Security*, pp. 256-268, 2009. Article (CrossRef Link)
[16] R. V. Yampolskiy and V. Govindaraju, "Embedded Noninteractive Continuous Bot Detection," *ACM Computers in Entertainment*, vol. 5, no. 4, pp. 1-11, 2008. Article (CrossRef Link)
[17] K. Woo, H. Kwon, H. Kim, C. Kim and H. K. Kim, Chong-kwon Kim, Huy Kang Kim, "What can Free Money Tell Us on the Virtual Black Market?," *ACM SIGCOMM Computer Communication Review - SIGCOMM '11*, vol. 41, no. 4, 2011. Article (CrossRef Link)
[18] A. R. Kang, J. Woo, J. Park and H. K. Kim, "Online Game Bot Detection based on Party-Play Log Analysis," *Computers and Mathematics with Applications*, *Article in press*, 2012. Article (CrossRef Link)
[19] A. R. Kang, J. Woo and H. K. Kim, "Data and Text Mining of Communication Patterns for Game Bot Detection," in *Proc. of 3th Int. Conf. on Internet 2011*, pp. 495-500, 2011.
[20] J. N. Kelly, "Play Time: An Overview of the MMORPG Genre," 2004. http://www.anthemion.org
[21] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, no.3, pp. 379-423, 1948. Article (CrossRef Link)
[22] E. M. Airoldi, X. Bai and B. A. Malin, "An Entropy Approach to Disclosure Risk Assessment: Lessons from Real Applications and Simulated Domains," *Decision Support Systems*, vol. 51, no.1, pp. 10-20, 2011. Article (CrossRef Link)
[23] R. Zheng, J. Li, H. Chen and Z. Huang, "A Framework for Authorship Identification of Online Messages: Writing-Style Features and Classification Techniques," *Journal of the American Society for Information Science and Technology*, vol. 57, no. 3, pp. 378–393, 2006. Article (CrossRef Link)
[24] M. Koppel, J. Schler and S. Argamon, "Computational Methods in Authorship Attribution," *Journal of the American Society for Information Science and Technology*, vol. 60, pp. 9–26, 2009. Article (CrossRef Link)
[25] F. Peng, D. Schuurmans, S. Wang and V. Keselj, **"**Language Independent Authorship Attribution using Character Level Language Models**,**" *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, vol. 1, pp. 267-274, 2003. Article (CrossRef Link)
[26] T. Opsahl, F. Agneessens and J. Skvoretz, "Node Centrality in Weighted Networks: Generalizing Degree and Shortest Paths," *Social Networks*, vol. 32, no. 3, pp. 245-251, 2010. Article (CrossRef Link)
[27] R. Diestel, "Graph Theory (3rd ed.)," *Springer-Verlag Heidelberg*, 2005.
[28] L. C. Freeman, "A Set of Measures of Centrality based upon Betweenness," *American Sociological Association*, vol. 40, no. 1, pp. 35–41, 1977. Article (CrossRef Link)
[29] G. Sabidussi, "The Centrality Index of a Graph," *Psychometrika,* vol. 31, no. 4, pp. 581-603, 1966. Article (CrossRef Link)

[30] M. E. J. Newman, "A Measure of Betweenness Centrality based on Random Walks," *Social Networks,* vol. 27, no. 1, pp. 39-54, 2003. Article (CrossRef Link)
[31] S. P. Borgatti, "Centrality and Network Flow," *Social Networks,* vol. 27, no. 1, pp. 55-71, 2005. Article (CrossRef Link)
[32] I. Hendrickx and A. Van Den Bosch, "Hybrid Algorithms with Instance-Based Classification," *Machine Learning: ECML2005,* vol. 3720, pp. 158-169, 2005. Article (CrossRef Link)
[33] P. Mehra and S. Chabbra, "Machine Learning based Anomaly Detection Techniques for Network Intrusion Detection Systems," *International Journal of Advances in Computer Networks and its Security*, vol. 1, no. 1, pp. 358-361, 2011. Article (CrossRef Link)
[34] S. Son, A. R. Kang, H. Kim, T. Kwon, J. Park and H. K. Kim, "Analysis of Context Dependence in Social Interaction Networks of a Massively Multiplayer Online Role-Playing Game," *PLoS ONE* , vol. 7, no. 4, 2012. Article (CrossRef Link)
[35] H. H. Do, S. Melnik and E. Rahm, "Comparison of Schema Matching Evaluations," *Lecture Notes in Computer Science*, vol. 2593/2003, pp. 221-237, 2003. Article (CrossRef Link)

Ah Reum Kang is taking Ph. D. degree course in Graduate School of Information Security, Center for Information Security Technologies (CIST) in Korea University. Her research interests include Online Game Security and Social Network. Contact her at armk@korea.ac.kr.

Huy Kang Kim received his Ph. D. degree in Industrial Engineering from Korean Advanced Institute of Science and Technology in 2009. Currently he is an assistant professor in Graduate School of Information Security, Center for Information Security Technologies (CIST) in Korea University. His research interests include Botnet Detection, Intrusion Detection System, Network Forensics and Online Game Security. Contact him at cenda@korea.ac.kr.

Jiyoung Woo received her Ph. D. degree in Industrial Engineering from Korean Advanced Institute of Science and Technology in 2006. Currently she is a research professor in Graduate School of Information Security, Center for Information Security Technologies (CIST) in Korea University. Her research interests include Social Media Analytics and Online Contents Security. Contact her at jywoo@korea.ac.kr.