Detecting Real Money Traders in MMORPG by Using Trading Network

Atsushi Fujita Hiroshi Itsuki Hitoshi Matsubara
Future University Hakodate
116-2, Kameda-nakano-cho, Hakodate, 041-8655, Japan
{fujita,g3106001,matsubar}@fun.ac.jp

Abstract

We have developed a method for detecting real money traders (RMTers) to support the operators of massively multiplayer online role-playing games (MMORPGs). RMTers, who earn currency in the real world by selling properties in the virtual world, tend to form alliances and frequently exchange a huge volume of virtual currency within such a community. The proposed method exploits (i) the trading network, to identify the communities of characters, and (ii) the volume of trades, to estimate the likelihood of communities and characters becoming engaged in real money trading. The results of an experiment using actual log data from a commercial MMORPG showed that using the trading network is more effective in detecting RMTers than conventional machine learning methods that assess individual character without referring to the trading network.

Introduction

These days, a variety of online services, including online games, are operated on high-speed computer networks. Online games include not only traditional games that share data via the network but also those facilitating interaction between numerous players. A representative genre of the latter is massively multiplayer online role-playing games (MMORPGs), in which thousands of players share both time and space. The motivation for playing online games is not limited to having fun. Players often pursue wealth in the virtual world, which can be in the form of a character's status, special items, and virtual currency.

Online games have been rapidly gaining popularity and their business models have been improved so as to better satisfy players. However, as online games evolve and social function and economy in the virtual world comes to resemble that of the real world, various problems have emerged, such as conflict and harassment between players and abusive access to the game system servers. Most of these problems are not specific to online games. They are also observed in other online services including social networking services and auction networks (Castronova 2005; Lehdonvirta 2009).

One of the most controversial actions a user of online services, including online games, can perform is real money trading (RMT). RMT is an economic activity in which virtual properties, such as currency, items, and even characters, are exchanged for currency in the real world, i.e., real money. Game operators have different attitudes toward RMT depending on the type, design, and business model of their game. There are two major opposing attitudes. One regards RMT as a harmless act and seeks ways to bring out its merit, such as by accelerating personal trades between players and cutting down costs for setting up a physical store. One famous example of this attitude in action is Second Life[1]. In contrast, the other sees RMT as the cause of problems and prohibits it in their games.

Most of the MMORPGs in Japan prohibit RMT because of the considerable problems it causes. To keep the virtual world sound and peaceful, operators of such MMORPGs have been taking strong actions against RMT, for example by banning the accounts of real money traders (RMTers). The inspection process, however, requires an incredible amount of human labor, and the online gaming community is in urgent need of methodologies and tools to assist them.

To address this need, we developed a method of detecting RMTers by exploiting actual log data of MMORPGs, in particular the trading network and the volume of trades.

Real Money Trading in MMORPGs

RMT in MMORPG causes serious problems for the operators, as exemplified below.

Imbalance of the virtual economy: The most critical influence is on economic balance. Dealing with a huge amount of virtual currency for RMT causes inflation in the virtual world and hampers the ability of general players to perform ordinary economic activities.

Direct harm of general players: RMTers often directly harm other general players by, for example, occupying specific locations for obtaining currency and items and attacking other players to rob their properties.

Encouragement of unauthorized deeds: RMTers also perform dishonest actions, such as cheating, using bots, and even taking over characters by fraud.

Discouragement of honest players: The unfair advantages of RMTers discourage honest players and often

[1] http://secondlife.com/

prompts them to quit the game. It also prevents new players from joining the game.

To keep the virtual world sound and peaceful, operators of many MMORPGs have been taking strong actions against RMT. The procedure of inspecting characters in an MMORPG can be done in the following steps.

Step 1. Identify suspects: Operators identify characters suspected of involvement in RMT on the basis of, for example, tip-offs from general players or self-advertisements of RMTers.

Step 2. Verify each suspect: Operators verify whether each suspect is an RMTer or not by referring to his/her previous actions and utterances in log data that are accumulated in the game server.

Step 3. Ban the account: Once a player is determined to be an RMTer, operators ban the account with/without exhortations. Players are allowed to use multiple characters in many MMORPGs, so all the characters operated by the same account are disabled.

The ultimate goal of MMORPG operators is to eliminate all RMTers from their game. However, to avoid kicking out honest players, an incredible amount of human labor is required for the manual verification in step 2. The amount of available human resources for steps 2 and 3 depends on the number of players/characters, the seriousness of RMT in the given MMORPG, and the budget for the operation.

There has been a recent trend of RMTers forming alliances for the division of RMT labor: each RMTer plays a specific role to maximize the effectiveness of earning real money. A rough classification of roles is shown below.

Sellers: Sell virtual property to general players and earn currency in the real world.

Earners: Acquire virtual property, such as currency and items, from the virtual world, non-player characters (NPCs), and general players by repeating specific actions in the virtual world.

Collectors: Convey virtual property from earners to sellers.

Virtual property trades can only be carried out in the virtual world by using the MMORPG infrastructure, so in general, the facts and figures of trading are recorded on log data in the game server, as are other actions. The history of a user's actions is supposed to be a strong clue in RMTer detection.

Related Work

To date, a few researchers have exploited log data to classify players and characters in MMORPGs. Matsumoto and Thawonmas (2004) proposed a method for classifying players on the basis of sequential patterns of action. However, the classes in their definition are specific to the game title, thus raising doubts as to the method's applicability. Soeda and Matsubara (2008) clustered characters in an MMORPG on the basis of the distribution of 428 types of action recorded in a certain period. However, each cluster does not necessarily correspond to some pre-determined type of character. Thus, their method cannot be directly applied to identification of specific type of characters, such as RMTers and bots.

There have also been several studies that attempt to detect title-independent types of characters. The most fundamental issue in such studies is to identify information that is useful for detecting the specific character type. Thawonmas, Kashifuji, and Chen (2008) and Chen et al. (2009), who targeted bots, reported that the frequency of actions and the deviation of traffic to the server are useful indicators.

RMTers in MMORPGs have also been investigated. Ahmad et al. (2009) regarded the task of detecting RMTers as a binary classification of individual character into either RMTer or not and applied various supervised machine-learning algorithms. In contrast, Itsuki et al. (2010) attempted to generate an RMTer ranking, assuming the succeeding manual inspection. However, both of them could not achieve reasonable performance. Keegan et al. (2010) reported several statistical tendencies observed through the trading network in an MMORPG, and made several findings: e.g., RMTers tended to repeat trading with each other. However, they did not investigate the behavior of RMTers in the entire network to find distinctive features; nor did they examine how their findings are helpful for detecting RMTers in practice. Even in their recent work, which augmented the character network with other elements (Ahmad et al. 2011), they did not address the task of detecting RMTers.

RMTer Detection Based on Communities

Aiming at reducing the amount of human labor in the succeeding manual inspection by MMORPG operators, we regard the task of detecting RMTers as ranking all the characters through estimating their likelihood of being an RMTer. This approach enables the operators to flexibly control the amount of human labor by the number of top-ranked suspects to be investigated. If we could give true RMTers a higher rank than honest characters, human labor would be vastly reduced.

Unlike previous work, we exploit the trading network for RMTer detection. We regard characters as nodes and trades between characters as edges because most MMORPGs only allow trades between two characters. Figure 1 depicts a trading sub-network surrounding manually identified RMTers, which is extracted from the actual log data used in our experiment. The figure demonstrates that RMTers tend to form dense relationships with each other. A large portion (61.5 – 83.3% in our data) of the RMTers' counterparties were also RMTers. In contrast, general players who traded with RMTers were significantly few ($<$1%). It is thus supposed that RMTers can be detected one after another once one of them is detected.

On the basis of this observation, we developed a method that exploits the trading network to detect RMTers. The entire procedure is comprised of the following two steps.

Step 1. Extracting communities: Identify the groups of characters who traded closely with each other.

Step 2. Ranking communities and characters: Rank the extracted communities in descending order of the total volume of in-house trades and then rank characters in each community in descending order of the total volume of trades.

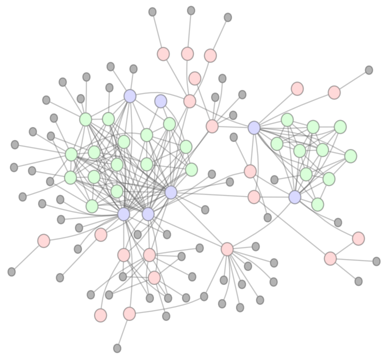


Figure 1: Trading sub-network surrounding manually identified RMTers (red: seller, green: earner, blue: collector) and non-RMTers (small nodes) in period B of our data.

Step 1. Extracting Communities

"Community" is defined as a set of nodes in a network that are tightly linked to each other. Methods of extracting communities have recently been applied to various networks, such as those of Web sites, citation networks, and users of social networking services.

The task of identifying communities in a given network is tied to determining the best division of the network. The notion of *modularity*, given by the following equation, was proposed by Newman and Girvan (2004) to quantify the quality of a division of a given network.

$$Q = \sum_i (e_{ii} - a_i^2),$$

where e_{ii} and a_i are calculated for each community c_i as

$$e_{ii} = \frac{\text{\# of edges between nodes in } c_i}{\text{\# of all edges in the network}},$$

$$a_i = \frac{\text{\# of edges connecting to a node in } c_i}{\text{\# of all edges in the network}}.$$

The more edges each community includes and the fewer number of edges that link different communities, the larger the modularity Q is. Note that the expected value of link ratio a_i is subtracted to avoid a trivial solution, i.e., regarding the whole network as one community.

The problem of finding a division that maximizes modularity is NP-hard. Thus, bottom-up algorithms for finding reasonable solutions at lower complexity have been studied (Newman 2004; Clauset, Newman, and Moore 2004). We extract communities from a given trading network by using the algorithm proposed in (Clauset, Newman, and Moore 2004). Let ΔQ_{ij} be the gain of Q if two communities c_i and c_j are merged. The algorithm is briefly explained below.

Step 1-1. Regard each node as a community and calculate ΔQ_{ij} for each connected pair of communities c_i and c_j.

Step 1-2. Merge c_i and c_j whose ΔQ_{ij} is largest and positive into $c_{i'}$. Increment Q by ΔQ_{ij}.

Step 1-3. Update $\Delta Q_{i'k}$ $(= \Delta Q_{ki'})$ for each community c_k that is connected to at least one of c_i or c_j.

Step 1-4. Repeat steps 2 and 3 while Q gains.

Table 1: Definition of trading networks.

Option	Edge	Weight of edge
tb	All trades	Existence of trades (uniform)
tt		Number of times of trades
cb	Currency trades	Existence of trades (uniform)
ct		Number of times of trades
cv		Volume of exchanged currency

This algorithm can effectively solve the problem as in (Newman and Girvan 2004; Newman 2004) despite its naiveness.

As summarized in Table 1, the effectiveness of focusing on currency trades and incorporating the weight of edges were examined by taking sum of the edge weights instead of counting the number of edges for calculating e_{ii} and a_i.

Step 2. Ranking Communities and Characters

The main focus of previous studies on community extraction was to discuss the properties of the extracted communities and the original network, such as scale-free and small world, and the appropriateness of the extracted communities. However, community extraction alone cannot function as either a classification nor ranking method for detecting RMTers.

We therefore give ranks to communities and characters. Following the finding in (Itsuki et al. 2010) that RMTers tend to deal with a huge volume of virtual currency, we determine the rank of characters by the following two steps.

Step 2-1. Communities are ranked in descending order of the total amount of trades in the community. We examine those of "tt," "ct," and "cv" as alternative options.

Step 2-2. Characters in each community are then ranked in descending order of the number of times they traded, the number of times they traded currency, and the volume of exchanged currency, alternatively. For convenience, they are also labeled "tt," "ct," and "cv," respectively.

Experiment with Actual Log Data

We performed an experiment to evaluate the performance of the proposed method, using the commercial MMORPG "Uncharted Waters Online" (Japanese version)[2] as the sample game. The data provided by the game operator (TECMO KOEI GAMES Co., Ltd.) included actual log data (actions of individual character and system alerts, but no personal information) and IDs of RMTers that are manually identified by the operators. In our experiment, we refer to only certain periods of log data for which manual inspection of RMTers was carried out. Table 2 shows the volume of corresponding records in the log data and the distribution of manually identified RMTers.

Extracted Communities

First, we extracted action records describing trades between two characters from the log data. Table 3 summarizes the statistics of the extracts. By targeting only traders, the number of suspects was reduced to half (all trades) and one-third (currency trades), but only one character in period D was

[2] http://www.gamecity.ne.jp/dol/

Table 2: Statistics of the provided data.
†Neither characters who did not play in each period nor NPCs are included.

Period			# of action records	# of chars. played in the period†	# of manually identified RMTers			
					Total	Seller	Earner	Collector
Period A	August 30 - September 13, 2009	(15 days)	308,921,785	15,249	29	10	15	4
Period B	November 18 - December 8, 2009	(21 days)	417,516,270	16,471	52	20	25	7
Period C	February 23 - March 17, 2010	(23 days)	479,468,978	18,745	106	29	54	23
Period D	May 10 - May 24, 2010	(15 days)	300,809,905	17,114	130	19	91	20

Table 3: Statistics of the identified trading networks.
tt: the total number of trading transactions, ct: the total number of currency trade transactions, cv: the total volume of exchanged currency, Cov.: the number of RMTers included in the extracts among all of the manually identified ones.

Period	All trades				Currency trades				
	# of nodes	# of edges	tt	Cov.	# of nodes	# of edges	ct	cv	Cov.
Period A	8,152	13,452	193,395	29	4,624	4,590	15,164	313,591,306,074	29
Period B	9,440	17,152	278,728	52	5,423	5,718	18,633	392,631,400,843	52
Period C	10,265	19,140	316,849	106	6,317	7,272	28,950	912,677,938,945	106
Period D	9,358	15,785	211,041	130	5,174	5,413	18,682	683,310,026,086	129

Table 4: Statistics of the extracted communities.
Q: modularity of the partitioned network, C: set of the extracted communities,
c_m: the maximum size of the communities ($\max_{c \in C} |c|$), C_{T}: set of communities containing at least one RMTer.

Option	Period A				Period B				Period C				Period D			
	Q	$\|C\|$	c_m	$\|C_{\mathrm{T}}\|$	Q	$\|C\|$	c_m	$\|C_{\mathrm{T}}\|$	Q	$\|C\|$	c_m	$\|C_{\mathrm{T}}\|$	Q	$\|C\|$	c_m	$\|C_{\mathrm{T}}\|$
tb	0.931	690	397	4	0.917	646	553	3	0.903	729	709	6	0.920	720	411	5
tt	0.986	818	280	1	0.986	812	199	3	0.985	902	397	3	0.984	885	264	6
cb	0.965	895	127	5	0.956	929	178	4	0.930	945	255	4	0.951	896	194	7
ct	0.965	909	96	2	0.965	947	153	2	0.952	986	326	3	0.961	911	140	6
cv	0.886	976	104	3	0.882	1,036	175	4	0.885	1,089	183	5	0.923	982	153	8

missed. This demonstrates that trading is a strong clue in detecting RMTers. Another notable feature of the trading network is the significant sparsity of edges. Only 0.04% of pairs of traders traded with each other. This highlights the computational efficiency of the method's use of community extraction compared to clustering algorithms that compute distance/similarity between two arbitrary nodes.

Next, we extracted communities from each of five types of trading network (see Table 1), individually. Table 4 shows some of the results. RMTers were condensed into a few communities: $|C_{\mathrm{T}}|$ was at most eight. This demonstrates the usefulness of the trading network and the community extraction method. By taking the weight of edges into account (tt, ct, cv), we could extract larger numbers of relatively smaller communities. In the "cv" network, the Q values were lower and $|C|$ values were higher than the others. We speculate that the pairs of characters who traded less currency were not preferably merged into the same community due to the large deviation of the volume of exchanged currency.

Newman and Girvan (2004) predicted that the Q value for general networks would fall between 0.3 and 0.7. However, all of the values for our networks were higher than that range. This implies that trades tend to be carried out between specific pairs of characters.

Evaluation Setting: Models and Measures

We combined the following three parameters, and obtained 45 types of ranking: five types of trading network for community extraction (tb, tt, cb, ct, cv), three measures for ranking communities (tt, ct, cv), and three measures for ranking characters (tt, ct, cv).

We employed three baselines[3]. The simplest baseline, "cv only," ranks characters in the descending order of the volume of exchanged currency[4]. Note that it can only rank currency traders. The remaining two baselines are well-known machine learning methods: naïve Bayes with multinomial distribution (MNB) (McCallum and Nigam 1998) and support vector machines (SVMs) (Vapnik 1999). While these methods are normally used to classify individual character into either RMTer or non-RMTer, here we use their classification score to rank the characters, i.e., a score is calculated for each character and then all the characters are ordered on the basis of the score. The score of MNB was calculated based on the posterior probability of each type of action, because the prior $p(\text{RMTer})/p(\text{non-RMTer})$ does not affect the ranking. We estimated the posterior probability by maximum likelihood estimation with Laplace smoothing. As the score of SVMs, we used the distance between the given char-

[3]We also applied agglomerative hierarchical clustering and k-means to the same data, varying the number of clusters from 2 to (the number of characters) -1. However, they produced only poor results despite their significant computation time, so we excluded them from comparison.

[4]This is not equivalent to the handled currency proposed in (Itsuki et al. 2010). They also counted currency flow between each character and the game system, such as shops and treasures.

acter and the learned hyperplane multiplied by the estimated label $l \in \{-1, +1\}$. For a fair comparison to the proposed method, we applied MNB and SVMs to three sets of characters: all characters, traders, and currency traders. The models were trained[5] on data from three periods other than the target period, referring to all of the 338 types of action recorded in the log data and their frequency that individual character had taken.

As mentioned earlier, the amount of human labor available for the verification of suspects depends on the situation. Thus, recall and precision were used to evaluate the performance at an arbitrary amount of human labor. Let N be the number of RMTers. They can be calculated for arbitrary k top-ranked characters to be investigated.

$$Recall = \frac{\text{\# of correctly identified RMTers}}{\text{\# of RMTers } (= N)},$$
$$Precision = \frac{\text{\# of correctly identified RMTers}}{\text{\# of players identified as RMTer } (= k)}.$$

The total performance of a ranking measure was also evaluated by average precision given by the following equation.

$$AveragePrecision(N) = \frac{1}{N}\sum_{n=1}^{N} P(n),$$

where $P(n)$ is the precision when n RMTers are identified. We used Wilcoxon signed rank-sum test to compare two arbitrary ranking measures having the N corresponding precision rates, at the significance level $p < 0.01$ (one-sided).

Results of RMTer Detection

We first evaluated each parameter of the proposed method by fixing the other two parameters; for example, five types of trading networks were compared for each of the 3×3 pairs of the other two parameters. We observed the following, each of which was common to all four periods.

Extracting the communities: Weighting the edges of the network (tt, ct, cv) did not necessarily yield a better performance than models that only took the existence of trades (tb, cb) into account. Moreover, different measures achieved the best result depending on the period.

Ranking the communities: Rankings based on the volume of exchanged currency (cv) were superior to those based on the number of trades (tt, ct). This is natural because RMTers tend to exchange a large volume of currency at once to effectively earn real money.

Ranking the characters: Currency trades (ct, cv) produced better results than relying on all trades (tt) in ranking the characters. This implies that virtual currency is more popular in RMT than other tradable elements.

Ten combinations of the above three parameters (5×1×2) remained as comparable versions of the proposed method. Henceforth, a combination of the three parameters is denoted by concatenating them with a single dot, e.g., "tt.cv.ct."

[5]While we implemented MNB, as an implementation of SVMs, we used SVM^{light} V6.02 (http://svmlight.joachims.org/) on its default setting, such as linear kernel.

Table 5: Average precision of selected models.
$^*P(130)$ is given assuming the worst case (130 / 17,114)

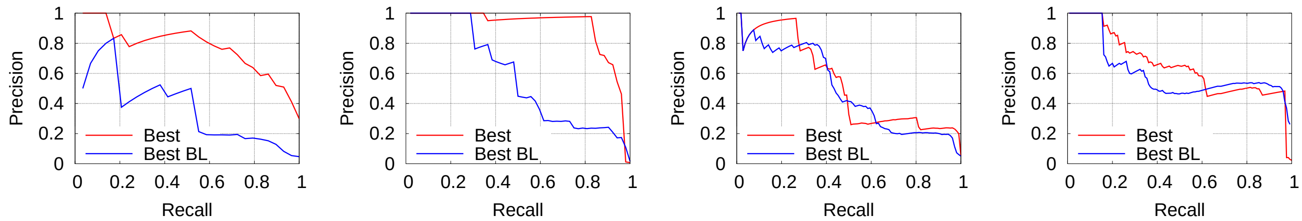
Model / Char. set	Period A $N = 29$	Period B $N = 52$	Period C $N = 106$	Period D $N = 130$
cv only	0.320	0.440	0.484	*0.466
MNB				
All chars	0.239	0.305	0.342	0.357
Traders	0.273	0.367	0.381	0.416
Cur. traders	0.336	0.391	0.420	*0.469
SVMs				
All chars	0.340	0.198	0.438	0.517
Traders	0.310	0.567	0.408	0.553
Cur. traders	0.356	0.554	0.421	*0.599
Proposed				
tb.cv.ct	0.385	**0.900**	0.499	0.404
tb.cv.cv	0.393	**0.860**	0.503	0.388
tt.cv.ct	0.328	**0.882**	0.459	**0.648**
tt.cv.cv	0.362	**0.837**	0.448	**0.624**
cb.cv.ct	0.167	**0.883**	**0.524**	*0.570
cb.cv.cv	0.179	**0.832**	**0.510**	*0.554
ct.cv.ct	**0.764**	0.626	**0.515**	*0.557
ct.cv.cv	**0.756**	0.606	0.498	*0.540
cv.cv.ct	**0.522**	0.573	**0.513**	*0.547
cv.cv.cv	**0.547**	0.564	0.498	*0.529

Table 5 summarizes the average precision of various models. The bold-faced scores indicate those significantly better than all of the baselines, while the best score for each period is underlined. For all of the four periods, one of our models achieved the best result, thus demonstrating the effectiveness of using the trading network. However, none of them always beat all of the baselines. Interestingly, MNB and SVMs tended to give a large weight to the actions associated with trading, which corroborates the usefulness of the trading information. In the scoring phase, however, these models also regarded the trades of honest characters as "bad behavior" and consequently pushed them up in ranking.

As shown in Table 3, RMTers are inseparable from trades. We therefore expected that precision could be higher when we targeted only traders. However, only MNB showed monotonic improvements. Surprisingly, SVMs were not stable at all: the average precision in period C deteriorated when it dealt with only traders. The rankings of the proposed method based on currency trades were not necessarily better than those based on all trades.

Figure 2 displays the recall-precision curves of the best version of the proposed method and the best baseline for each period. Similar to these curves, most of the models demonstrated the trade-off between recall and precision. In periods A and B, some versions of the proposed method had significantly higher precision than the baselines at a wide range of recall. In contrast, however, the advantage of the proposed method over the baselines was limited to a narrow range of recall and small in periods C and D.

By definition, communities themselves do not guarantee to contain only RMTers. Thus, the strict order of ranking strategy of the proposed method, i.e., communities first and characters second, was too naive to achieve a reasonable performance. Even if RMTers in a community were exhaus-



(a) Period A (b) Period B (c) Period C (d) Period D

Figure 2: Recall-Precision curves of the best version of the proposed method and the best baseline.

tively identified, the remaining honest characters in the community were still prioritized over RMTers in the succeeding communities. In reality, the operators can conduct the inspection more intelligently, exploiting the ranked communities. For example, the operators should look ahead characters in the succeeding communities if a certain number of honest characters in a community under investigation are incorrectly identified as RMTer. With such a dynamic ranking mechanism, the performance can be improved further.

Conclusion and Future Work

In this paper, we addressed the task of detecting RMTers in MMORPGs to support game operators and proposed a method that exploits the trading network and the volume of trades. The results of an experiment using actual log data demonstrated the effectiveness of using the trading data.

The trading network and the volume of trades are substantial clues in detecting RMTers, because the exchange rate tends to be extremely low and all the trades are recorded at the game server. Thus, further investigation into trading networks is promising to boost the performance. For example, several aspects of trading, such as currency and items, can be distinguished by introducing mixture models (Newman and Leicht 2007). Augmentation of the network with other components than characters (Ahmad et al. 2011) is also worth investigating. We also intend to evaluate the proposed method from two aspects. The one is to apply it to a shorter period of log data to determine the robustness of the method: whether it can be used to prevent the disposable use of characters. The other is its application to other MMORPG titles to empirically justify its generality.

Acknowledgments

The authors are deeply grateful to TECMO KOEI GAMES Co., Ltd. who provided actual log data of MMORPG and the anonymous reviewers for their valuable comments. This work was supported by JST Core Research of Evolutional Science & Technology (CREST) Project entitled "Development Support and Evaluation of On-line Games."

References

Ahmad, M. A.; Keegan, B.; Srivastava, J.; Williams, D.; and Contractor, N. 2009. Mining for gold farmers: Automatic detection of deviant players in MMOGS. In *Proceedings of the 2009 IEEE Social Computing Conference, Symposium on Social Intelligence and Networking*.

Ahmad, M. A.; Keegan, B.; Williams, D.; Srivastava, J.; and Contractor, N. 2011. Trust amongst rogues? a hypergraph approach for comparing clandestine trust networks in MMOGs. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media*, 10–17.

Castronova, E. 2005. *Synthetic Worlds: The Business and Culture of Online Games*. The University of Chicago Press.

Chen, K.-T.; Jiang, J.-W.; Huang, P.; Chu, H.-H.; Lei, C.-L.; and Chen, W.-C. 2009. Identifying MMORPG bots: A traffic analysis approach. *EURASIP Journal on Advances in Signal Processing* 2009:1–22.

Clauset, A.; Newman, M. E. J.; and Moore, C. 2004. Finding community structure in very large networks. *Physical Review E* 70(066111).

Itsuki, H.; Takeuchi, A.; Fujita, A.; and Matsubara, H. 2010. Exploiting MMORPG log data toward efficient RMT player detection. In *Proceedings of the 7th International Conference on Advances in Computer Entertainment Technology*.

Keegan, B.; Ahmad, M. A.; Williams, D.; Srivastava, J.; and Contractor, N. 2010. Dark gold: Statistical properties of clandestine networks in massively-muliplayer online games. In *Proceedings of the 2010 IEEE Social Computing Conference*.

Lehdonvirta, V. 2009. Virtual item sales as a revenue model: Identifying attributes that drive purchase decisions. *Electronic Commerce Research* 9(1-2):97–113.

Matsumoto, Y., and Thawonmas, R. 2004. MMOG player classification using hidden Markov models. In *Proceedings of the 3rd International Conference on Entertainment Computing*, 429–434.

McCallum, A., and Nigam, K. 1998. A comparison of event models for Naïve Bayes text classification. In *Proceedings of AAAI/ICML-98 Workshop on Learning for Text Categorization*, 41–48.

Newman, M. E. J., and Girvan, M. 2004. Finding and evaluating community structure in networks. *Physical Review E* 69(026113).

Newman, M. E. J. 2004. Fast algorithm for detecting community structure in networks. *Physical Review E* 69(066133).

Newman, M. E. J., and Leicht, E. A. 2007. Mixture models and exploratory analysis in networks. *National Academy of Sciences of the USA* 104(23):9564–9569.

Soeda, S., and Matsubara, H. 2008. Tools to support management of online games. *Journal of the Japanese Society for Artificial Intelligence* 23(1):68–73. (in Japanese).

Thawonmas, R.; Kashifuji, Y.; and Chen, K.-T. 2008. Detection of MMORPG bots based on behavior analysis. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology*, 91–94.

Vapnik, V. N. 1999. *The Nature of Statistical Learning Theory: 2nd Ed.* Springer-Verlag.