Proc. of the 5th Game-On International Conference, CGAIDE 2004, Reading, UK, pp. 311-316, Nov. 2004.

EVALUATING REPUTATION OF ONLINE-GAME PLAYERS BASED ON INCOMING CHAT MESSAGES

Ruck Thawonmas, Yao Zhai, and Yuki Konno
Intelligent Computer Entertainment Laboratory
Department of Human and Computer Intellgence, Ritsumeikan University
Kusatsu, Shiga 525-8577, Japan
URL: www.ice.ci.ritsumei.ac.jp
E-mail: ruck@ci.ritsumei.ac.jp

KEYWORDS

Reputation Systems, Chat Messages, Massively Multiplayer Online Games, Online Communities

ABSTRACT

In this paper, we propose an online-game player reputation system that is based on incoming chat messages to each player. The key concepts for implementation of the proposed system are that good players are those who received many chat messages and that good players are those who received chat messages from other good players. The proposed system is tested using computer simulations, in which issues on balance, reliability, and characteristics of different implementation recipes are examined.

INTRODUCTION

The online game industry is one of the fastest growing industries. Among various types of online games, MMOGs (Massively Multiplayer Online Games) have the most important role. According to The Themis Group (Alexander et al. 2004), estimated worldwide revenues of MMOGs will rise from 1.30 Billion USD in 2004 to 4.10 Billion USD in 2008, and to 9 Billion USD in 2014.

In MMOGs, social connections among players are naturally formed. Being in a good community usually makes a player addicted to the game. For game companies, this means *more revenues*. To maintain a good community, many types of social systems have been developed. Typical existing social systems are as follows (Pizer 2003):

Chat systems for conveying messages among players,

Guild systems for letting players form their own communities,

Reputation systems for helping players determine whom they can trust.

Among these social systems, reputation systems are arguably most sophisticated, and are gaining a lot of interests among developers and researchers on online communities (see for example http://depts.washington.edu/ccce/digitalMedia/reps.html). For MMOGs, existing reputation systems (Brockington 2003), such as those in *Ultima Online, EverQuest, Neverwinter Nights*, can automatically detect nasty players by checking, for example, whether or not a player kills other player characters. The reputation systems then reduce the reputation values of those nasty players accordingly. Detection of good players is technically more challenging. For this task, some MMOGs, such as *Lineage II* (http://www.lineage2.com/), have a reputation system based directly on user feedback. User feedback, unfortunately, is not always reliable.

In this paper, we propose a new reputation system that evaluates the reputation of each player based on incoming chat messages. Our system is partly inspired by PageRank (Brin and Page 1998) used in Google (http://www.google.com) as wells as Activity Rating proposed by Kobayashi in his M.Eng. Thesis (Kobayashi 2002). In PageRank, the importance of a page relies on whether or not that page is cited by many other pages and/or by important pages. The basic concepts of the proposed system, their implementations, and experimental results are given in the following sections.

PLAYER REPUTATION

As in Activity Rating, four basic concepts that we employ for evaluating the reputation of each player in the game are as follows:

Concept I Players who receive many chat messages are good.

Concept II Players who receive chat messages from good players are good.

Concept III Players who receive chat messages from many different players are good.

Concept IV Players who have recently received chat messages are good.

Concepts I and II are similar with the key concepts of PageRank. Following the recipe of PageRank, the reputation of the sender of a chat message is evenly distributed to that of the receivers. However, PageRank iteratively computes the importance of a given page by summing the evenly distributed importance of the citing pages until the resulting value converges. It thus requires high computational costs. In our reputation system, upon receiving a chat message, the evenly distributed reputation from the sender is added to the current reputation of the receiver of the chat message.

Concept III indicates that a player who received, say, one hundred chat messages, each from a different sender, has higher reputation than a player who received one hundred chat messages from the same sender. For implementing this concept, we introduce a sender list of player i that keeps track of the senders of the chat messages received by player i in FIFO (first-in, first-out) discipline. Our use of the sender list is straightforward. Namely, a chat message from a sender who is not in the sender list of player i has the highest weight in calculation of the reputation of player i. For a chat message from a sender who is in the sender list, the nearer to the front of the list the sender is, the higher weight has the chat message in calculation of the reputation of player i.

Concept IV is for better reflecting the current reputation of a player. Here, it is simply implemented by regularly decreasing the reputation of each player. This kind of implementation increases the gap between the reputation of the players who have recently received chat messages and that of the players who have not.

Now we are ready to give an implementation recipe of the proposed reputation system. Let $rep(x)$ denote the reputation of player x. We assume player S sends a message to M other players, $R_1, R_2, \ldots, R_M$. Upon receiving the message, the reputation of R_i, where $i \in \{1, 2, \ldots M\}$, is given as follows:

$$rep(R_i) = \min(rep(R_i) + \alpha\frac{rep(S)}{M} + \beta(R_i, S), rep_{MAX}). \tag{1}$$

In (1), the maximum reputation of each player is limited to the parameter rep_{MAX}. The term $\alpha\frac{rep(S)}{M}$ implements Concepts I and II and uses α, a small positive constant, for controlling the evenly distributed (among M receivers of the message) reputation from player S. The term $\beta(R_i, S)$ implements Concept III and its value is evaluated before S is added into the sender list of R_i; this term is defined as follows:

$$\beta(R_i, S) = \begin{cases} \gamma_0, & \text{if } send(R_i, S) = 0 \\ \frac{\gamma_1}{2^{send(R_i,S)-1}}, & \text{otherwise,} \end{cases} \tag{2}$$

where γ_0 and γ_1 are small positive constants, $\gamma_0 \geq \gamma_1$, and $send(x, y)$ is the function that returns the position of player y from the front of the sender list of player x. For the sender list of size L, if player y is present in the list, $send(x, y) \in \{1, 2, \ldots, L\}$, where the value of 1 indicates the front and L the back of the list; otherwise, $send(x, y) = 0$.

A typical implementation of Concept IV is given as follows: for every N chat messages sent by player i,

$$rep(i) = \max((1 - \tau)rep(i), rep_{MIN}), \tag{3}$$

where τ is a small positive constant, and rep_{MIN} is the parameter defining the minimum reputation of each player.

EXPERIMENTS

To examine the characteristics of the proposed reputation system, we conducted computer simulations in which, unless stated otherwise, $\alpha, \gamma_0, \gamma_1, L, \tau, rep_{MAX}, rep_{MIN}$, and N were set to 0.0001, 0.002, 0.0016, 5, 0.035, 1.0, 0.1, and 20, respectively. For each player, their initial reputation was set to 0.1.

Balance of Reputation

Here, we report our results where there are two types of players in the game, i.e., socializers and standard players. Following the definition in (Bartle 2004), socializers are players whose main emphasis is to interact with other players. In addition, socializers tend to give useful information to other players. In this respect, they are good players.

In our simulations, 50 players out of 500 players are socializers who send/and receive chat messages twice more than the other 450 players, simply called standard players. In particular, the following experimental procedure was performed.

Step 1 Randomly divide 500 players into 100 groups, each having at most 10 players.

Step 2 Have all players in each group randomly send chat messages to other players in the same group.

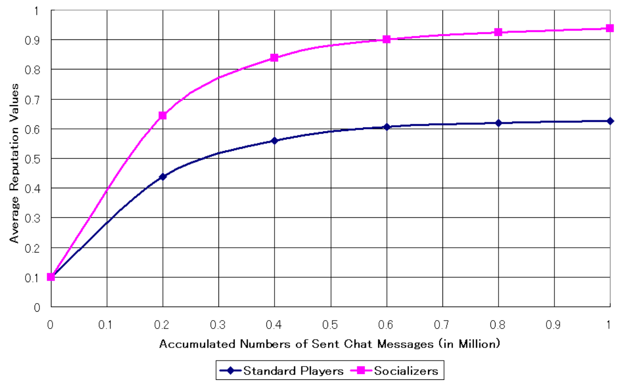


Figure 2: Average Reputation Values over Accumulated Numbers of Sent Chat Messages for the Standard Players and the Socializers

Step 3 Repeat Step 2 until the average number of messages sent by each player reaches 20.

Step 4 Have all 50 socializers randomly send chat messages to other socializers.

Step 5 Repeat Step 4 until the average number of chat messages sent by each socializer reaches 20.

Step 6 Repeat the procedure from Step 1 until the total number of sent chat messages reaches one million.

Figure 1 shows the histograms of the player reputation at different accumulated numbers of sent chat messages. At 0.2 million messages, almost all reputation values are clustered around the middle. These values are shifted rightward as the number of sent messages increases. At 1.0 million messages, the reputation values form two clusters, the bigger one around the range [0.6, 0.7] and the smaller one around [0.9, 1.0]. From Fig. 2 discussed below, the former represents the cluster of the standard players, and the latter that of the socializers.

Figure 2 shows the average reputation values of the standard players and those of the socializers over different accumulated numbers of sent chat messages. As can be seen, both series sharply rise in the beginning and then saturate, to 0.63 for the standard players and 0.94 for the socializers.

From the above results, reputation is well balanced for each player type. In addition, the socializers eventually have higher reputation than the standard players, as we expect they should.

System Reliability

Here we want to know whether or not our reputation system can cope with cheating by players. For example, a situation may arise where some players intentionally continue sending chat messages to particular friends just for the only purpose of increasing their friends' reputation. We tested the reliability of the proposed system by simulating the following two scenarios.

Scenario 1 Player A continues receiving chat messages from a friend whose reputation value is 0.1.

Scenario 2 Player B continues receiving chat messages from a friend whose reputation value is 0.9.

Figure 3 shows the reputation values of player A and those of player B over different accumulated numbers of received chat messages. From this figure, though the reputation values of both players increase as the accumulated number of received chat messages increases, a relatively high number of chat messages are needed to double the reputation value of players A and B, i.e, more than 1,000 chat messages and 600 chat messages, respectively.

Comparisons among Different Recipes

Here, we compare three different recipes for implementing Concepts I-IV. Each recipe consists of two parts, one for Concepts I-III and the other for Concept IV. By assuming that player S sends a message to M other players, $R_1, R_2, \ldots, R_M$, each recipe is given in the following.

Recipe 1 This is the one we have discussed so far, i.e., (1) for Concepts I-III and (3) for Concept IV.

Recipe 2 For Concepts I-III, the reputation of R_i is given as follows:

$$rep(R_i) = \min(rep(R_i) + \beta(R_i, S)\frac{rep(S)}{M}, rep_{MAX}), \quad (4)$$

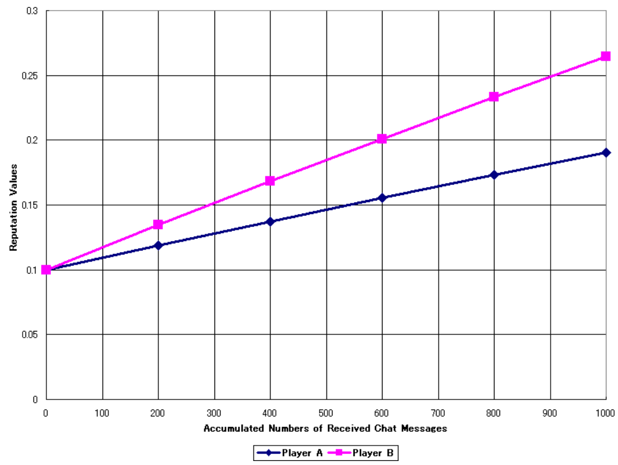


Figure 3: Average Reputation Values over Accumulated Numbers of Received Chat Messages for Player A and Player B

where γ_0 and γ_1 were set to 0.004 and 0.003 in the experiments, respectively. For Concept IV, (3) is used.

Recipe 3 For Concepts I-III, the reputation of R_i is given as follows:

$$rep(R_i) = \min(rep(R_i) + \beta(R_i, S), rep_{MAX}), \quad (5)$$

where γ_0 and γ_1 were set to 0.004 and 0.003 in the experiments, respectively. Concept IV is implemented as follows: for every N chat messages sent by player i,

$$rep(i) = \max(rep(i) - \delta, rep_{MIN}), \quad (6)$$

where δ is a small positive constant; in the experiments, it was set to 0.035.

Figure 4 shows for each recipe the average reputation values of the standard players and those of the socializers over different accumulated numbers of sent chat messages. As it is clear from this figure, each recipe has its own characteristic curve. In practice, they should be properly selected by game designers. For example, Recipe 1 seems to be a good candidate if the game designer wants the player reputation to rise more easily in the beginning and then gradually make it more difficult.

DISSCUSION ON PRACTICAL ISSUES

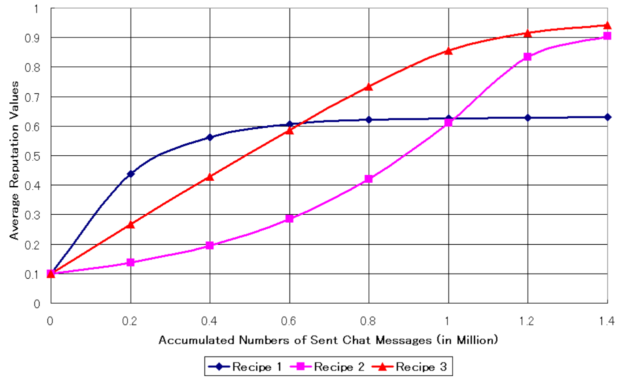
In most MMOGs the concept of a reputation system is informed in advance to all players. So, for successful use in practice, the proposed reputation system definitely should be operated together with other related systems such as a system that detects cheating by a guild of dedicated players or bots. Such cheating can be easily noticed by, for example, an unnatural rise in the reputation value. Players involved in detected cheating (of course this has to be confirmed manually by Game Masters) might be penalized by increasing the length of their sender lists or decreasing the values of α, γ_0, and γ_1. This would make such players more difficult to raise their reputation values.

We also expect that once a player is aware of the proposed reputation system, they would refrain from sending negative comments to bad players, such as killers as defined in (Bartle 2004). They don't want to unnecessarily raise the reputation of those bad players. Instead, players would send such complaints to Game Masters.

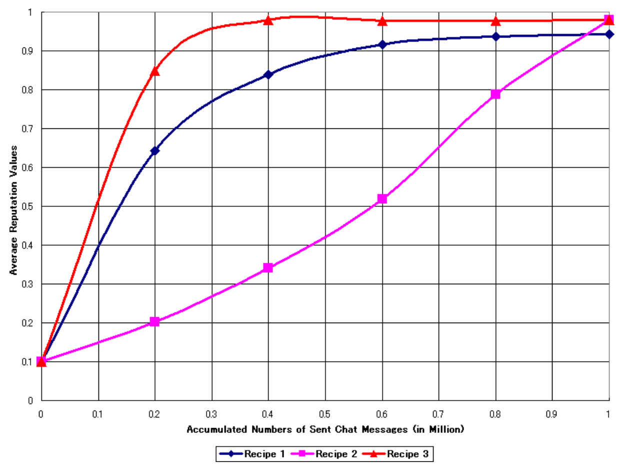
CONCLUSIONS AND FUTURE WORK

In this paper we have presented how a reputation system for online-game players can be constructed based on incoming chat messages to each player. The system has a good balance for players of the same type and well reflects players' chatting characteristics. It can also gracefully tolerate cheating by players. We have discussed three different recipes for implementation of the proposed system, each having its own characteristics that should fit different online-game design concepts.

As our future work, we plan to study how to take into account also the content of chat messages in calculation of the player reputation. We also plan to test the reputation system in an edutainment multiplayer online game called The ICE (Thawonmas and Yagome 2004), under development at our research laboratory.



(a) Standard Players



(b) Socializers

Figure 4: Average Reputation Values over Accumulated Numbers of Sent Chat Messages for Recipes 1, 2, and 3

ACKNOWLEDGEMENTS

This work has been supported in part by the Ritsumeikan University's **Kyoto Art and Entertainment Innovation Research**, a project of the 21^{st} Century Center of Excellence Program funded by the Japanese Ministry of Education, Culture, Science and Technology; and by Grant-in-Aid for Scientific Research (C), Number 16500091, the Japan Society for Promotion of Science.

REFERENCES

Alexander, K.; Bartle, R.; Castronova, E.; Costikyan, G.; Hayter, J.; Kurz, T.; Manachi, D.; Smith, J. 2004. "The Themis Report on Online Gaming 2004." www.themis-group.com/reports.phtml.

Bartle, R. 2004. *Designing Virtual Worlds* News Riders Publishing, Boston·Indianapolis·London·Munich·New York· San Francisco.

Brin, S. and L. Page. 1998. "The anatomy of a large-scale hypertextual Web search engine." *Computer Networks and ISDN Systems*, vol. 30, no. 1–7, pp. 107–117.

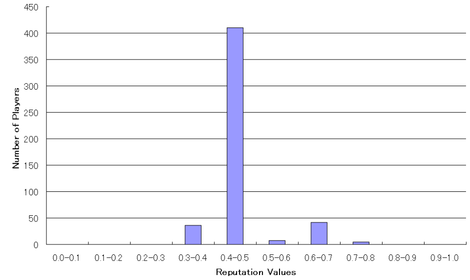
Brockington, M. 2003. "Building a Reputation System: Hatred, Forgiveness, and Surrender in Neverwinter Nights." In *Massively Multiplayer Game Development* Alexander, T., ed., Charles River Media, Inc., Massachusetts, pp. 454–463.

Kobayashi, T. 2002. "Users' Activity Rating Algorithm for Online Communication" M.Eng. Thesis, Department of Communications and Computer Engineering. http://forest.kuee.kyoto-u.ac.jp/research/master/2001/pdf/toshi-mt.pdf (in Japanese)

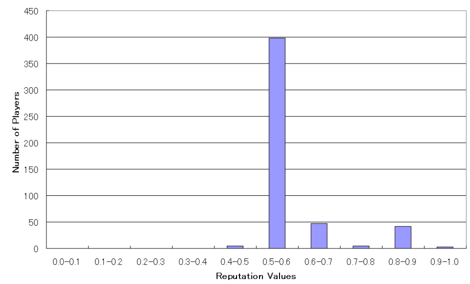
Pizer, P. 2003. "Social Game Systems: Cultivating Player Socialization and Providing Alternate Routes to Game Rewards." In *Massively Multiplayer Game Development*

Alexander, T., ed., Charles River Media, Inc., Massachusetts, pp. 427–441.

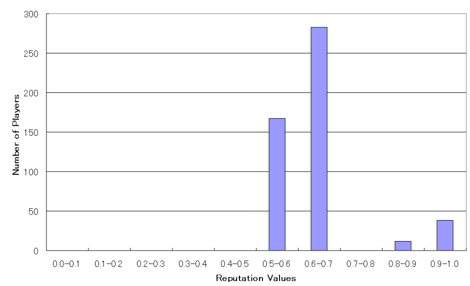
Thawonmas, R. and T. Yagome. 2004. "Application of the Artificial Society Approach to Multiplayer Online Games: A Case Study on Effects of a Robot Rental Mechanism." In *Proc. of the 3rd International Conference on Application and Development of Computer Games (ADCOG 2004)* (HKSAR, Apr.), pp. 12–17.



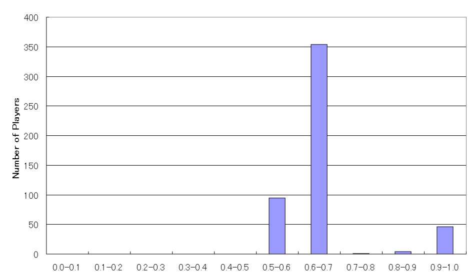
(a) 0.2 million messages



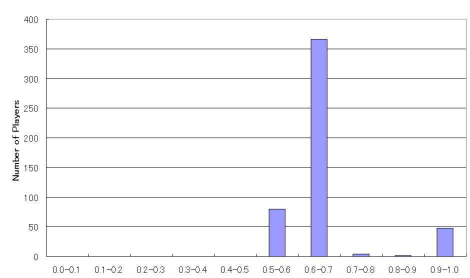
(b) 0.4 million messages



(c) 0.6 million messages



(d) 0.8 million messages



(e) 1.0 million messages

Figure 1: Histograms of the Player Reputation