Abstract—To successfully operate online games, gaming companies are introducing systematic customer relationship management model. Particularly, churn analysis is one of the most important issues, because preventing a customer from churning is often more cost-efficient than acquiring a new customer. Churn prediction models should thus consider maximizing not only accuracy but also the expected profit derived from the churn prevention. We, thus, propose a churn prediction method for optimizing profit consisting of two main steps: 1) selecting prediction target, 2) tuning threshold of the model. In online games, the distribution of a user’s customer lifetime value is very biased that a few users contribute to most of the sales, and most of the churners are no-paying users. Consequently, it is cost-effective to focus on churn prediction to loyal customers who have sufficient benefits. Furthermore, it is more profitable to adjust the threshold of the prediction model so that the expected profit is maximized rather than maximizing the accuracy. We applied the proposed method to real-world online game service, Aion, one of the most popular online games in South Korea, and then show that our method has more cost-effectiveness than the prediction model for total users when the campaign cost and the conversion rate are considered.

Index Terms—churn prediction, cost-benefit analysis, customer lifetime value, game analytics, data mining.

I. INTRODUCTION

Online games have become one of the most successful online services with cumulative user base reaching around 40% of the global online population [1] and a global online gaming market valued at around USD 20 billion [2]. To successfully operate an online game, it is crucial not only to create a fun game but also to provide proper user management. Therefore, gaming companies are introducing various customer relationship management techniques to monitor and analyze the behavior of game users [3].

In general, preventing a customer from churning is more cost-efficient than acquiring a new customer [4]. For high retention rates, companies implement various strategies, such as launching promotional campaign incentives to remain loyal. Considering the cost associated with such incentivizing policy, correctly identifying and targeting potential churners has become increasingly crucial, and with continuously advancing data mining techniques, many researchers have studied various classification methods [5].

Churn prediction’s main goal is pursuing high accuracy to find possible churners, but a churn prediction model should consider maximizing the profit derived from churn prevention as well. There are various studies for a churn prediction technique and tools that account for the associated costs and derived earnings [4], [6]–[8]. While these studies propose general frameworks or methods, several issues should be considered upon applying them to an online game.

First, churn should be appropriately defined unlike traditional subscription-based services, such as telecommunications or banking services, online game services do not require formal withdrawal procedures, such as deleting accounts. In fact, many online game users stop playing the game while maintaining their accounts, as there is no cost associated with maintaining accounts. In the case of Aion, extensively analyzed throughout this paper, only 0.8% of users who have not resumed playing over a year deleted their accounts. Many users also resume playing the game after a long-term hiatus – about 50% of the Aion users resume playing after a one-month break. Consequently, to minimize the cost from false-positive and undetected cases, correctly defining churn is crucial.

Second, a churn prediction target should be appropriately selected. Because the ultimate goal of churn prediction is to prevent loss derived from churning, users who add profits when retained should be the primary target of churn prediction models. Considering additional marketing and operating costs to switch churners, it is not cost effective to target all users in predicting churn. Also, increasing the survival rate of users who access a game temporarily due to curiosity may be a valuable strategy to pursue a newly released game title, but for a mature game title, it is more useful to focus on preventing the churn of loyal users. Furthermore, some users affect a game service negatively, that is, gold farming group (GFG) using game bots [9] or cyber bullies who disturb others online [10]. If these users are not filtered out correctly, a churn prevention strategy can have an adverse effect to the game service. Consequently, it is necessary to select an appropriate target before churn prediction modeling.

Third, it is necessary to quantitatively estimate the profit generated when a potential user is retained. Accurate expected benefit estimates are essential information for determining churn prevention strategies since the cost of them may exceed the expected profit.

Therefore, we propose a churn prediction process and a detailed methodology considering the factors mentioned above.
The characteristics of the proposed method are as follows:

1) We extract “long-term loyal customers” and analyze them for churn prediction. To do this, users are assigned to loyalty grades by using their in-game activities and payment patterns, and the sequence of their loyalty levels are analyzed for approximately six months.

2) We estimate the expected profit of preventing user churn using cost-benefit analysis. The churn prediction model then is optimized for maximizing the profit.

We have applied the proposed method to the live data of Aion, one of the most popular massively multiplayer online role-playing games (MMORPGs) in South Korea that has been serviced for nine years from 2009.

The key contributions of this paper are as follows:

1) We propose a churn prediction framework considering the expected profit in online games. The framework includes predictive modeling as well as the related prerequisite works such as churn definition and selection of prediction targets and performs a cost-benefit analysis of the prediction model. To the best of our knowledge, this is the first study that has considered overall procedures concerning expected profit in the online game field.

2) The cost-benefit analysis techniques in other fields have been modified for the application of the online game domain. The modified method was used to optimize the prediction model via simulations, using real data and comparing it with the results by only considering prediction accuracy. We expect our result to serve as an excellent benchmarking reference for other churn prediction models regarding online games.

3) We analyze the game activities for one and a half years of approximately half a million users and find that the social events in the game are highly related to user churn. Additionally, trends and volatility of in-game activities can also be an essential factor for churn prediction. Our results can constitute a reference for churn analysis in other online games.

The remainder of this paper is organized as follows. In section II, we review related studies. Section III presents the dataset and criteria for churn analysis. Section IV outlines the proposed methodology for churn prediction. Section V describes the characteristics of churners via exploratory data analysis. In section VI, we evaluate the performance of the prediction model and show the result of profit optimization. Section VII discusses limitations of our study and issues that need to be improved via future works. Finally, we summarize our findings and conclude with section VIII.

II. RELATED LITERATURE

Churn analysis using data mining is a subject actively studied in various fields [11].

Wei et al. [12] and Mozer et al. [13] are pioneering papers regarding churn analysis in the mobile communications industry. Particularly, Mozer et al. analyzes churn from the viewpoint of service satisfaction, credit risk, and service provider profitability of subscribers. On the other hand, Baumann et al. [14] proposes a prediction method, focusing on profit growth through churn prevention from an actual marketing viewpoint, and estimates the per user profit that can be obtained through experiments using five public communication data. Moreover, Dasgupta et al. [15] proposes a technique for predicting churners through the call network analysis of mobile phone users. This network analysis technique was also used in several online games [16]–[18].

Glady et al. [19] and Mavri et al. [20] target churn analysis in the financial industry. Mavri et al. investigates customer churn using Greek bank data and analyzes the determinants of the churn rate increase by using the hazard proportional model and a survival analysis methods. Alternatively, Glady et al. proposes a method for churn prediction by defining customer loyalty via customer lifetime value and analyzing customer behavior, which is expected to decrease future profits. It is similar to Mozer et al. [13] and Baumann et al. [14]; in terms of studying customers’ churn from the viewpoint of revenue obtained from them.

Churn analysis has also been actively conducted in the field of online games. Runge et al. [21] is a study on the technique of churn prediction of users who have a high profit in casual games, and Hadiji et al. [22] proposes prediction framework on free-to-play game. Milošević et al. [23] shows an impressive result in predicting early churners and preventing their departure using A/B testing. Tamassia et al. [24] proposes a Hidden Markov Model to utilize the features of time series data. Periáñez et al. [25] and Viljanen et al. [26] propose applying survival analysis methods for churn prediction. Borbora et al. [27] proposes a method that clusters game users based on motivation theory and applies a model suitable for each group. Although clustering is similar to our study in that it divides the type of users, we analyze the churn only for the groups with the highest expected profit, whereas Borbora et al. proposes the model for each type. We think that their study will be a good reference when proceeding the churn prediction for more diverse types in the future.

Furthermore, there are various studies on the effect of social relations on user churn. Kawale et al. [28] is a study on the social relationship between users who perform a quest together, and it analyzes the effect of one user’s churn on another. Park et al. [29] reveals that in-game achievements are primary factors for users with characters in a growth period to continue playing the game whereas communications and social interactions among users are important for users who already completed growth period. Moreover, Shores et al. [10] analyzes the effect of malicious users bullying other users by using user reputation data for League of Legends. Our study also shows that the factors related to social activities have a significant influence on the user churn, which supports the results of the above studies.

Feng et al. [30] and Debeauvais et al. [31] show that users have a change in play time or frequency just before leaving via study of popular MMORPGs. According to our study, Aion also shows that trend and volatility of gaming activity are highly correlated with churn.

Finally, there are studies of cost-benefit analysis for churn prediction model: Hand et al. [8], Bahnsen et al. [7], and Verbraken et al. [4], [6]. Theses describe how to measure
performance and compensate the limit of commonly used ROC curves when measuring the performance of classification model—such as churn prediction—from the viewpoint of expected profit and cost. Specifically, Hand et al. proposes H-measure in terms of minimizing costs, and Verbraken et al. proposes a indicator designed from the perspective of maximizing profits by expressing an expected profit that takes the operating cost and conversion rate of a campaign execution into account. Bahnsen et al. [7] and Verbraken et al. [4] refer to [6], and propose a method for predictive modeling aiming to maximize profit rather than accuracy. Our formula used for profit evaluation is created by referring to [6] and [7]. Additionally, while the previous studies mainly focus on the prediction modeling stage, we propose an overall procedure consisting of churn definition, predicting target selection, prediction modeling, and cost-benefit analysis.

III. Dataset

We use the dataset of the MMORPG Aion, which launched in November 2008. Specifically, we used the game activity logs of one million users from November 2014 to July 2017.

A. Time unit for data aggregation

Many Aion users have a weekly cycle of play pattern. While there are users who access the game daily, most users access it only on weekends. Moreover, Aion has a few hours of downtime every Wednesday morning for game updates or server maintenance. Consequently, all statistics are based on data weekly aggregated from a Wednesday morning to the subsequent Wednesday before downtime.

B. Churn definition

For online games, unlike telecommunications or financial services, it is inappropriate to use withdrawal of membership as a definition of churn. In Aion, only 0.8% of users who have not accessed the game for more than a year have explicitly unsubscribed. If we define membership withdrawal as churn, most users who have effectively churned will be classified as non-churners. Therefore we should identify a user as a churner using inactivity period. If we set the period to define a churner to be too long, the expected benefits of churn prevention will be reduced because a user who is actually churning will not be identified until it is too late. On the other hand, if the period is too short, most users who return after a short break will be misclassified as churners, and unnecessary costs will be incurred. Consequently, for optimizing profit via churn prediction, it is necessary to set the appropriate period to define a churner.

Aion usually goes through a significant contents update every six months. Therefore, the developing team of Aion defines 26 weeks as the maximum possible dormant period for a user because a user who does not respond to the major update should be regarded as having lost interest in the game entirely. However, it is necessary to define a shorter period because a half-year is too long when considering the expected profit related to user churn.

To find an appropriate period, we calculated $P_n(churn)$, the fraction of users who have not accessed the game for 26 weeks among users who have not accessed the game at least n weeks (see Eq. 1).

$$P_n(churn) = \frac{F(26)}{F(n)}, 1 \leq n \leq 26$$

where $F(n)$ represents the number of users who haven’t logged in for at least n weeks.

To avoid bias associated with specific dates, we used the median of $P_n(churn)$ measured using multiple 26-week-sequences from November 5, 2014, to December 23, 2015 (see Figure 1). Although the ‘elbow method’ is commonly used to determine an appropriate parameter, the outcome as shown in Figure 1 did not present any outstanding point. Consequently, we use $P_n(churn) = 0.75$ as an ‘elbow’ and define a churner as a user who does not access the game for more than 13 weeks. In other words, we compromise between certainty and profit; the compromise is that among the players we define as churners, 25% are not going to churn according to Aion definition.

IV. Methodology

This section describes the detailed methodology of the proposed churn prediction process. The overall process consists of 1) selection of prediction targets, 2) feature engineering, 3) expected profit evaluation.

A. Prediction target selection

To identify long-term loyal customers who are expected to generate profit if retained, we first assign loyalty grades based on the user’s weekly gaming activities and payment amounts, then track the grade change for thirty weeks to select a user group that maintains a high grade for a sufficiently long period. The details are as follows.

1) Loyalty grade assignment: First, user’s in-game activities described in Table I are aggregated weekly and clustered into nine clusters using the k-means clustering algorithm. We then analyzed the characteristics of each clusters using the average value of the in-game activity features and extracted the following additional indicators to understand the characteristics of the clusters better.
### TABLE I
FEATURES FOR IN-GAME ACTIVITIES.

<table>
<thead>
<tr>
<th>Name</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>DICE</td>
<td>The count of rolling dice</td>
</tr>
<tr>
<td>DT</td>
<td>The number of days a user plays</td>
</tr>
<tr>
<td>EXP</td>
<td>The amount of experience a user obtains</td>
</tr>
<tr>
<td>FORTRESS</td>
<td>The number of combats in a fortress a user participates</td>
</tr>
<tr>
<td>AP</td>
<td>The achievement points a user obtains</td>
</tr>
<tr>
<td>GP</td>
<td>The glory points a user obtains</td>
</tr>
<tr>
<td>GLIDE</td>
<td>The number of glide activities a user play</td>
</tr>
<tr>
<td>DUNGEON</td>
<td>The number of dungeons a user plays</td>
</tr>
<tr>
<td>PLAYTIME</td>
<td>Play time (unit: hour)</td>
</tr>
<tr>
<td>PVP</td>
<td>The number of combats against another player</td>
</tr>
<tr>
<td>PVE</td>
<td>The number of combats against NPC</td>
</tr>
<tr>
<td>HARVEST</td>
<td>The number of materials a user obtains in the field</td>
</tr>
<tr>
<td>QUEST</td>
<td>The number of quests a user completes</td>
</tr>
<tr>
<td>TELEPORT</td>
<td>The number of teleports a user casts</td>
</tr>
<tr>
<td>SCROLL</td>
<td>The count of scrolls which a user uses</td>
</tr>
<tr>
<td>GETMONEY</td>
<td>The amount of game money a user obtains from other users (unit: millions)</td>
</tr>
<tr>
<td>GIVEMONEY</td>
<td>The amount of game money a user gives to other users (unit: millions)</td>
</tr>
<tr>
<td>PARTY</td>
<td>The number of parties a user joins</td>
</tr>
<tr>
<td>PARTYMEMBER</td>
<td>The number of users who are in the same party</td>
</tr>
</tbody>
</table>

**ARPU.** Average Revenue Per User. ARPU is expressed as a relative value so that the average ARPU of all users is 1 instead of the actual one to mask the actual ARPU.

**BOT.** The ratio of users detected as game bots via the detection system [9] among users in each clusters.

**IP.** The number of other users accessed to IP address which a user accesses the most frequently.

Table II summarizes the characteristics of each cluster type. Cluster 1 through 5 show positive correlations between in-game activities and ARPU. On the other hand, cluster 6 and 7 have very low ARPU despite the extremely long PLAYTIME. The activities related to the production of goods such as PVE, HARVEST and GETMONEY are very high, while activities related to the consumption of goods or interaction between users such as PVP, DICE, GLIDE, SCROLL, and GIVEMONEY have relatively low values. Besides, these types have a very high ratio of game bots compared to other types. Consequently, we defined these types as game bots.

Cluster 9 has a low level of overall activities, but it has a large number of users accessing the game using the same IP address. Although this type has a low ratio of detected game bots, we identified these users to be malicious accounts of ‘GFGs’ which operate numerous game bots or cheap laborers to obtain virtual assets and monetize. According to our previous study [32], some GFGs manage multiple accounts with little playtime to avoid bot detection. Finally, cluster 8 is an outlier type with a deficient proportion of actual users and extreme values for several game activities.

After that, nine clusters are reassigned into six grades in descending order based on their profitability to simplify user types — grade 1 represents those with the highest profitability. Cluster 1 through 5 are assigned to grade 1 through 4 ordered by the number of in-game activities and payment amount.

Game bots (Cluster 6, 7 and 9) are integrated as grade 5 because they have negative impacts on other game users. Cluster 8 is assigned as grade 6 — the lowest grade — to be excluded from further analysis. Table III shows the result of reassigning for loyalty grades.

2) **Long-term loyal customer extraction:** In addition to the six grades, a user who did not access the game for a week or just recently subscribed was given a grade 9 (dormant state) and grade 0 (unsubscribed state) for that respective period. With the additional two grades as well as the six grades mentioned above, all users have a 30-weeks-long data sequence consisting of 8 possible weekly grades.

![Fig. 2. Categorizing customers via sequence clustering. Users of red box(left top) are identified as long-term loyal customers because they are given a grade 1 or 2 for most periods.](image)
### TABLE II
SUMMARY OF THE USER CLUSTERING RESULT

<table>
<thead>
<tr>
<th>Cluster No.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate (%)</td>
<td>0.97</td>
<td>4.08</td>
<td>8.54</td>
<td>6.33</td>
<td>20.93</td>
<td>15.51</td>
<td>8.09</td>
<td>0.04</td>
<td>35.52</td>
</tr>
<tr>
<td>PLAYTIME</td>
<td>79.82</td>
<td>66.90</td>
<td>37.33</td>
<td>15.57</td>
<td>2.94</td>
<td>152.06</td>
<td>138.24</td>
<td>56.42</td>
<td>1.73</td>
</tr>
<tr>
<td>PVP</td>
<td>442.98</td>
<td>261.27</td>
<td>88.70</td>
<td>8.70</td>
<td>3.54</td>
<td>0.88</td>
<td>0.71</td>
<td>4.32</td>
<td>0.02</td>
</tr>
<tr>
<td>PVE</td>
<td>2,515.14</td>
<td>2,278.43</td>
<td>1,204.67</td>
<td>187.28</td>
<td>94.57</td>
<td>31,215.15</td>
<td>7,598.14</td>
<td>141.02</td>
<td>14.75</td>
</tr>
<tr>
<td>HARVEST</td>
<td>8.95</td>
<td>16.87</td>
<td>17.68</td>
<td>34.50</td>
<td>2.84</td>
<td>0</td>
<td>557.12</td>
<td>10,999.47</td>
<td>0.76</td>
</tr>
<tr>
<td>SCROLL</td>
<td>1,942.62</td>
<td>1,412.52</td>
<td>623.37</td>
<td>77.59</td>
<td>38.87</td>
<td>47.80</td>
<td>14.81</td>
<td>41.40</td>
<td>0.54</td>
</tr>
<tr>
<td>GETMONEY</td>
<td>113.48</td>
<td>93.69</td>
<td>42.51</td>
<td>5.71</td>
<td>4.29</td>
<td>171.50</td>
<td>320.09</td>
<td>60.88</td>
<td>1.70</td>
</tr>
<tr>
<td>GIVEMONEY</td>
<td>392.31</td>
<td>284.38</td>
<td>110.71</td>
<td>46.59</td>
<td>19.93</td>
<td>59.38</td>
<td>74.91</td>
<td>154.41</td>
<td>7.73</td>
</tr>
<tr>
<td>DICEMONEY</td>
<td>667.53</td>
<td>464.80</td>
<td>180.04</td>
<td>18.85</td>
<td>10.06</td>
<td>0.80</td>
<td>2.04</td>
<td>13.43</td>
<td>0.09</td>
</tr>
<tr>
<td>GLIDE</td>
<td>538.50</td>
<td>415.40</td>
<td>195.98</td>
<td>27.13</td>
<td>15.78</td>
<td>2.12</td>
<td>5.33</td>
<td>14.65</td>
<td>0.19</td>
</tr>
<tr>
<td>SCROLL</td>
<td>1,942.62</td>
<td>1,412.52</td>
<td>623.37</td>
<td>77.59</td>
<td>38.87</td>
<td>47.80</td>
<td>14.81</td>
<td>41.40</td>
<td>0.54</td>
</tr>
<tr>
<td>ARPU</td>
<td>7.93</td>
<td>6.00</td>
<td>4.31</td>
<td>2.50</td>
<td>0.44</td>
<td>0.09</td>
<td>0.31</td>
<td>0.94</td>
<td>0.06</td>
</tr>
<tr>
<td>BOTT (%)</td>
<td>0.52</td>
<td>1.48</td>
<td>1.20</td>
<td>0.10</td>
<td>0.00</td>
<td>15.74</td>
<td>60.12</td>
<td>0.00</td>
<td>0.05</td>
</tr>
<tr>
<td>IP</td>
<td>2.36</td>
<td>2.60</td>
<td>2.35</td>
<td>1.94</td>
<td>2.66</td>
<td>17.97</td>
<td>18.56</td>
<td>10.35</td>
<td>641.57</td>
</tr>
</tbody>
</table>

### TABLE III
CUSTOMER LOYALTY GRADE.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Cluster No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>Users who had many in-game activities and spent significant amounts</td>
</tr>
<tr>
<td>2</td>
<td>2 and 3</td>
<td>Users who had some in-game activities and spent moderate amounts</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>Users who had some in-game activities and spent small amounts</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>Users who had a few in-game activities and spent almost no money</td>
</tr>
<tr>
<td>5</td>
<td>6, 7 and 9</td>
<td>Users who are suspected game bots</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>Outliers</td>
</tr>
</tbody>
</table>

Features used in the churn prediction model are as follows.

1) **Social relationship – friend and legion:** Aion provides various social activities in the virtual world. Game users can make friends and join a guild called ‘legion,’ which consists of dozens to hundreds of members that cooperate and compete against other legions. A negative social experience such as a friend quitting a game, reduction or disbandment of the legion due to legion members retiring affects a user’s churn rate [28]. Table IV shows the features accounting of social ties among friends and legion members.

### TABLE IV
FEATURES OF THE SOCIAL RELATION INDICATOR.

<table>
<thead>
<tr>
<th>No.</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td># of friends who played the game over the last 4 weeks</td>
</tr>
<tr>
<td>2</td>
<td># of friends who were in the same party over the last 4 weeks</td>
</tr>
<tr>
<td>3</td>
<td># of legion members who played the game over the last 4 weeks</td>
</tr>
<tr>
<td>4</td>
<td># of legion members who were in the same party over the last 4 weeks</td>
</tr>
<tr>
<td>5</td>
<td># of times a user moved legion over the last 12 weeks</td>
</tr>
</tbody>
</table>

2) **Social relationship – party:** Users can also temporarily organize a party of two to six members to complete a quest or explore a dungeon. Given most activities in Aion require collaboration among party members, party activities also influence user churn [28]. We use social network indicators to extract party-related features. The feature engineering process for the party network is as follows:

1) Party network is constructed. Nodes represent users and edges represent party relationship. An edge is generated only when the party is performed more than or equal to twice between the same users over a week to prevent the excessive network construction.
2) Party network is divided into multiple communities so that the edges between the nodes in the same community is denser than the edges between nodes in other communities. The more detailed process is explained in the next paragraph.

B. Feature engineering

Algorithm 1 Prediction target selection – only choose players that have never been grade 5 or 6, and have been most often one of grade 1 to 3, and have played for at least 10 weeks.

```python
if C(13, 1) + C(13, 2) + C(13, 3) + C(13, 4) ≥ 10 AND mode_grade(13) ≤ 3 AND max_grade(13) ≤ 4 then
    extract the user
else
    filter out the user
end if
```
3) The indicators described in Table V are calculated for each community network.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clustering coefficient</td>
<td>The likelihood that three nodes will become fully connected forming a clique</td>
</tr>
<tr>
<td>Graph density</td>
<td>( \frac{2E}{(E - N + 1)} ), ((E, N)): the number of edges and nodes</td>
</tr>
<tr>
<td>Community size</td>
<td>The number of nodes in a community</td>
</tr>
</tbody>
</table>

The algorithm proposed in Clauset et al. [34] is used to divide communities on the party network. To do this, an indicator called ‘modularity’ is calculated as per Eq. 3

\[
Q = \sum_{i} (e_{ii} - a_{ii}^2)
\]  

where \( e_{ii} \) is the fraction of edges between nodes in \( i \)th community to all edges in the network and \( a_{ii} \) is the fraction of edges connecting to a node in \( i \)th community to all edges in the network.

The modularity \( Q \) is larger as each community includes the more edges and the edges between communities is the fewer. The communities are extracted as the below.

1) At first, each node is constructed a community.
2) The modularity \( Q \) is calculated as per Eq. 3
3) For each community pair, \( \Delta Q \), the difference in \( Q \) when two communities are combined with the \( Q \) of the previous step, is calculated.
4) Combine the pairs where \( \Delta Q \) is the maximum
5) Repeat step 2, 3 and 4 until \( \Delta Q \) is no more positive.

3) Trend and deviation for in-game activities: Changes in play time or frequency are highly correlated with a user’s churn [31]. Notably, long-term loyal customers tend to access and play the game steadily and consistently. Therefore, we assume that these users gradually lose interest in the game rather than suddenly leaving the game, so, the number of their activities or the number of connections in the game is likely to decrease as well gradually. Various statistics related to the amount of in-game activity for each user are collected on a weekly basis for the 12 weeks to measure trends and volatilities. The specific in-game actions are described in Table I.

The methods for measuring trend and deviation for the statistics are described as follows:

1) Moving Average Convergence Divergence (MACD), which is calculated by the difference between the moving average for four weeks and 12 weeks. If this value is a positive number, it indicates an increasing trend and a negative number indicates a decreasing trend.
2) Coefficient of variation, which is the variance divided by the average statistics for 12 weeks. A high value means that a user’s in-game activity is irregular.

All features are scaled via the standardization as per Eq. 4.

\[
x'_i = \frac{x_i - \mu}{\sigma}
\]

where \( \mu \) and \( \sigma \) is a mean and a standard deviation for each feature, respectively.

C. Profit evaluation

When the retention campaign is applied, the expected profit per user can be described as Figure 3.

In Figure 3, \( CLV \) is the expected customer lifetime value that the company gains if a user continues to play the game, and \( C \) is the campaign cost. \( TP \) and \( TN \) denote the proportion of users who are accurately classified as churners and non-churners among all users, and \( FP \) and \( FN \) represents the proportion of users misclassified as churners or non-churners among all users.

Campaign cost \( (C) \) represents the price of discount coupons or reward items that are provided to change the mind of potential churners and retain them. If a game company gives reward items to users who are classified as churners via the prediction model, the total cost of such campaign is proportional to the number of targeted users. Consequently, a campaign cost is expressed as \( TP \) and \( FP \) multiplied by \( C \). \( \gamma \) is the rate of users who are retained via the retention campaign among potential churners. If a churner is correctly identified, the expected profit per detected churner would be \( \gamma \) of \( CLV \) (i.e., \( \gamma \times CLV \)).

In the classification model, the rate of true positives or false positives depends on the threshold used to determine whether a user has churned. Consequently, \( TP \) and \( FP \) can all be expressed as a function of threshold \( t \), and then the expected profit can be shown as per Eq. 5:

\[
Profit(t) = CLV\{\gamma TP(t)\} - C\{TP(t) + FP(t)\}
\]  

The \( CLV \) can be calculated as the product of the user’s service life expectancy and the average revenue generated by the user per time unit. Since it is difficult to estimate the service life expectancy of a user accurately, it is generally calculated using the infinite series of a retention rate as follows [35]:

\[
CLV = \sum_{i=0}^{\infty} (ARPU \times r^i) = \frac{ARPU}{1 - r}
\]

In Eq. 6, \( ARPU \) stands for the average revenue per user, and \( r \) for the retention rate. The original equation includes the interest rate and operating cost per time unit, but these two parameters are omitted for simplicity in this research because for online games, the interest rate and operating cost are far less than revenue generated from a user.
The above equation assumes that the retention rate and ARPU per time unit are constant. However, weekly ARPU and retention rate vary depending on the time of aggregation. Therefore, we used the following method to determine the ARPU and the retention rate with change over time:

1) The ARPU and the retention rate for \( n \) weeks are calculated. \( n \) ranges from 1 to 15 with one increments. These values are called \( ARPU_n \) and \( r_n \), respectively.

2) The variances of \( ARPU_n \) and \( r_n \) for the various time points are calculated. These are called \( \text{var}(ARPU_n) \) and \( \text{var}(r_n) \) respectively.

3) Determine \( ARPU_n \) and \( r_n \) at the point where \( \text{var}(ARPU_n) \) and \( \text{var}(r_n) \) are smallest.

Figure 4 shows the variance of \( ARPU_n \) and \( r_n \) for long-term loyal customers. The \( \text{var}(ARPU_n) \) continues to decline as the aggregate period increases, but after the point where \( n \) is 9, there is an insignificant difference. The variance of the retention rate, on the other hand, is in the form of a cubic line with the least variance point where \( n \) is 13. Based on these data, we choose 13 weeks as a unit period for ARPU and \( r \) to calculate CLV.

Furthermore, we calculate the individual CLV of each user instead of just calculating the average CLV for the entire user base to estimate the expected profit precisely when the predicted model is applied. For this, we use the payment amount for the last 13 weeks for each user instead of ARPU. After dividing the whole user base into long-term customer group and the rest, the retention rate for each group is used because the retention rate is not available for individual measurement.

We use the function in Eq. 5 to measure the expected profit of a churn prediction model and to find the threshold that maximizes the expected profit. The detailed process is described in section VI.

\[
\text{Expected Profit} = \sum_{n=1}^{15} \text{CLV}_n \times \text{ retention rate}_n 
\]

\( \text{CLV}_n \) is the coefficient of variation of playtime for churners. The MACD for playtime also shows a clear difference between the churners and non-churners, and such quantitative difference in playtime between churners and non-churners are depicted in Figure 6. Consequently, the trends and variations of users’ in-game activities are measured by the various methods described in section IV-B3 and are used as features of the prediction model.

\[
\text{MACD} = \text{Playtime}_n - \text{Playtime}_{n-1} 
\]

\[
\text{Playtime CV} = \frac{\text{standard deviation}}{\text{mean}} \times 100 
\]

\( \text{MACD} \) for churners became irregular as churning point approached while it remained constant for non-churners; hence higher coefficient of variation of playtime for churners. The MACD for playtime also shows a clear difference between the churners and non-churners.

Fig. 5. Comparison of play trend between churners and non-churners. The playtime of churners gradually decreases from about ten weeks before churning.

B. Party Activity

As mentioned in section IV-B2, party activities are critical social activities that users need to play the game. Many of the contents provided by the game require collaboration with other users through party activities.

We constructed a social network for cases in which users formed a party with the same members for at least two times and extracted party communities using the method described in section IV-B2. We then calculated network indicators, such as community density, clustering coefficient, and community size (see Figure 7).

The clustering coefficient of communities for churners appears to be much lower than that of non-churners, while
the size of the party community for churners is larger than non-churners. This characteristic is related to the sociality of churners. The clustering coefficient is the rate at which any three nodes within the community form a clique and are completely connected to each other. Therefore, the low clustering coefficient of communities for churners means that party members who churners formed a party with more than once are not mutual party members. On the other hand, party members of non-churners tend to be shared party members since non-churners repeatedly perform party activities with members from previous parties. Consequently, we can assume that the non-churners mainly perform party activities with friends while churners tend to perform party activities with random users.

For example, in the case of Aion, a party usually consists of six players. Therefore, the size of the closed community that only performs parties with the same members will not exceed 6, and the clustering coefficient is 1. On the other hand, in the case of the community that consists of members who tend to form party no more than once with the same member, the community size is a maximum of 6\(^n\), and the clustering coefficient is closed to 0 (see Figure 8).

Previous studies have shown that social networks, such as Twitter and Facebook, have higher clustering coefficients than random networks [36]. This study supports our assumptions.

**C. Legion**

Legions also affect the users’ experience. Members of good legions will be more attached to the game due to the additional benefits and sense of belonging they attain from their legions. The exact opposite is the case for members of small or inactive legions as they feel isolated instead of feeling attached; hence more likely to churn.

Table VI shows that churners tend not to join a legion compared to non-churners. Moreover, churners’ legions have lower activity values—members’ play time, loyalty to the legion, and legion points compared to non-churners’ legions (see Figure 9).

**TABLE VI**

<table>
<thead>
<tr>
<th>Legion</th>
<th>No legion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Churner</td>
<td>33%</td>
</tr>
<tr>
<td>Non-churner</td>
<td>42%</td>
</tr>
</tbody>
</table>

Another interesting finding is that churners tend to drift from a legion to another more frequently than non-churners (see Figure 10). Such frequent transfer of legions indirectly indicates a low sense of belonging, and previous studies have shown that game users tend to leave legions if they have little benefit or minimal communication with legion members [18]. The desire for such personal accomplishments and interaction with other users is a crucial factor of player retention [29].
VI. EXPERIMENTS

A. Total churner prediction vs. loyal churner prediction

We aggregated users who played the game at least once from March to June 2017 for the experiments. We then tagged a long-term loyal customer and a churner who left a game within four weeks. Customer lifetime value (CLV) was calculated separately for long-term loyal customers and the rest. All features, calculations, and tags were generated using the process described in section IV. For confidentiality, CLV has been normalized so that the value for the rest set to one.

Table VII shows that the long-term loyal customers are 2.4% of the total users, and only 11% of them leave a game within four weeks. However, CLV per user of long-term loyal customers is about 300 times higher than that of non-loyal customers. This means that preventing one loyal customer from churning will yield an expected profit comparable to that of preventing three hundred non-loyal customers from churning. Consequently, the total CLV for long-term loyal churners is larger than the value for the rest, while the number of them is only 0.37% of total churners.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Long-term loyal</td>
<td>Others</td>
</tr>
<tr>
<td># of users</td>
<td>15,797</td>
<td>637,527</td>
</tr>
<tr>
<td># of churners</td>
<td>1,801</td>
<td>480,064</td>
</tr>
<tr>
<td>ratio of churners</td>
<td>11.40%</td>
<td>75.40%</td>
</tr>
<tr>
<td>CLV (total)</td>
<td>4,750,029</td>
<td>637,527</td>
</tr>
<tr>
<td>CLV (per user)</td>
<td>300.69</td>
<td>1</td>
</tr>
</tbody>
</table>

We constructed two training sets. One is the training set extracted from total users, and another is obtained from only long-term loyal customers. Both training sets were sampled as the ratio of churners to non-churners becomes one-to-one. We then constructed test sets for both the entire users and the long-term loyal customers to compare the expected profit of the prediction models. In the test set I, the ratio of the long-term loyal customers and the ratio of the churners were adjusted to be the same as those in the table VII to accurately estimate the expected profit. All datasets were randomly sampled from the dataset described in table VII. The test set II consisted of only long-term loyal customers who extracted from the test set I. We then created three prediction models for the training set I and the training set II, respectively, using random forest, XG boost, and generalized boosting regression as training algorithms. Table VIII describes the size of training sets and test sets constructed for our experiments. Maximum expected profit is the value which can be obtained when all churners are switched from churners to non-churners.

To calculate the expected profit, we should do a campaign which provides reward items to the predicted churners and then measure the conversion rate. Unfortunately, we were not able to apply the churn prevention service to the live game service. Consequently, we compared the differences between models by carefully selecting γ and C.

Table IX shows the comparison of the results of six prediction models. Columns of test set I are the result of prediction models using the test set I and columns of test set II are the result of prediction models using the test set II. RF, XGB, and GBM denote random forest, XG boost, and generalized boosting regression, respectively. The expected profits were calculated assuming γ and C are set to the values written in table IX. Each value for γ and C was assigned as realistic as possible with the help of domain experts. C and the expected profit value were normalized in the same manner as CLV in table VII.

Table IX shows that the prediction accuracy results of the test set II are lower than the accuracies of the test set I, whereas the expected benefits of test set II are significantly higher than the test set I, even though maximum expected profit of the test set I is higher than the test set II. The reason is that CLV for true positive users of the test set II intend to be much higher than those of the test set I (see Figure 11). If a retention campaign is progressed using the results of the test set I, the game company will take a monetary loss except for the case in which the campaign cost is close to zero. When a churn prediction model was constructed for all users, any long-term loyal churners were not detected while only churners with very low CLV intended to be detected. The reason is that the long-term loyal churners are only about 0.3% of the total users, so a training algorithm treated them as an outlier in the training process. As a result, while the expected profit obtained from true positive users of the test set I is meager, the cost to be spent is enormous because the total number of churners is large. Consequently, the total expected profit is very small or even harmful. On the other hand, in the test set II, even though the number of true positive users is tiny, the CLV per target user of retention campaign is very high, and the total cost for them is low. Consequently, The total expected profit is very high compared to the test set I.

B. Threshold optimization

As described in the previous section, it is crucial to detect as many as possible of the users who have high CLV. Therefore, it is advantageous to reduce the number of false negative users as many as possible even if the proportion of false positives is increased in the churn predictive model for long-term loyal customers with high expected profit. Consequently, it is a better decision to keep the threshold for classifying churners lower. Table X shows how much the predicted profit increases when the threshold is adjusted to maximize the expected profit using the three models of the test set II used in the previous
TABLE IX
COMPARISON OF PREDICTION PERFORMANCE AND EXPECTED PROFIT BETWEEN THE TEST SET I (FOR TOTAL CUSTOMERS) AND THE TEST SET II (FOR ONLY LONG-TERM LOYAL CUSTOMERS).

<table>
<thead>
<tr>
<th></th>
<th>Test set I</th>
<th>Test set II</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RF</td>
<td>XGB</td>
</tr>
<tr>
<td># of true positive users</td>
<td>63,778</td>
<td>63,249</td>
</tr>
<tr>
<td># of false positive users</td>
<td>3,917</td>
<td>4,307</td>
</tr>
<tr>
<td># of true negatives users</td>
<td>22,253</td>
<td>21,843</td>
</tr>
<tr>
<td># of false negatives users</td>
<td>10,072</td>
<td>10,601</td>
</tr>
<tr>
<td>acc</td>
<td>0.8601</td>
<td>0.8509</td>
</tr>
<tr>
<td>precision</td>
<td>0.9421</td>
<td>0.9362</td>
</tr>
<tr>
<td>recall</td>
<td>0.8636</td>
<td>0.8565</td>
</tr>
<tr>
<td>F1 score</td>
<td>0.9012</td>
<td>0.8946</td>
</tr>
<tr>
<td>AUC</td>
<td>0.9358</td>
<td>0.9264</td>
</tr>
</tbody>
</table>

TABLE X
COMPARISON OF EXPECTED PROFIT BETWEEN THRESHOLD OPTIMIZING MODEL AND BASE MODEL (THRESHOLD SET TO 0.5) IN THE TEST SET II.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0.01</th>
<th>0.1</th>
<th>1</th>
<th>0</th>
<th>0.01</th>
<th>0.1</th>
<th>1</th>
<th>0</th>
<th>0.01</th>
<th>0.1</th>
<th>1</th>
<th>0</th>
<th>0.01</th>
<th>0.1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RF</td>
<td>XGB</td>
<td>GBM</td>
<td>RF</td>
<td>XGB</td>
<td>GBM</td>
<td>RF</td>
<td>XGB</td>
<td>GBM</td>
<td>RF</td>
<td>XGB</td>
<td>GBM</td>
<td>RF</td>
<td>XGB</td>
<td>GBM</td>
<td>RF</td>
</tr>
<tr>
<td>base profit</td>
<td>5,192</td>
<td>5,184</td>
<td>5,114</td>
<td>4,412</td>
<td>2,596</td>
<td>2,588</td>
<td>2,518</td>
<td>1,816</td>
<td>519</td>
<td>511</td>
<td>441</td>
<td>-261</td>
<td>519</td>
<td>511</td>
<td>441</td>
<td>-261</td>
</tr>
<tr>
<td>adjusted threshold</td>
<td>0</td>
<td>0.11</td>
<td>0.11</td>
<td>0.25</td>
<td>0</td>
<td>0.11</td>
<td>0.16</td>
<td>0.5</td>
<td>0</td>
<td>0.11</td>
<td>0.25</td>
<td>0.89</td>
<td>0</td>
<td>0.11</td>
<td>0.25</td>
<td>0.89</td>
</tr>
<tr>
<td>optimized profit</td>
<td>6,738</td>
<td>6,717</td>
<td>6,523</td>
<td>4,989</td>
<td>3,369</td>
<td>3,348</td>
<td>3,163</td>
<td>1,816</td>
<td>674</td>
<td>652</td>
<td>499</td>
<td>9</td>
<td>674</td>
<td>652</td>
<td>499</td>
<td>9</td>
</tr>
<tr>
<td>increasing rate</td>
<td>0.2978</td>
<td>0.2956</td>
<td>0.2754</td>
<td>0.1309</td>
<td>0.2978</td>
<td>0.2935</td>
<td>0.2563</td>
<td>0</td>
<td>0.2983</td>
<td>0.2764</td>
<td>0.1314</td>
<td>-</td>
<td>0.2983</td>
<td>0.2764</td>
<td>0.1314</td>
<td>-</td>
</tr>
<tr>
<td>XGB base profit</td>
<td>5,169</td>
<td>5,161</td>
<td>5,086</td>
<td>4,341</td>
<td>2,585</td>
<td>2,576</td>
<td>2,502</td>
<td>1,757</td>
<td>517</td>
<td>509</td>
<td>434</td>
<td>412</td>
<td>517</td>
<td>509</td>
<td>434</td>
<td>412</td>
</tr>
<tr>
<td>adjusted threshold</td>
<td>0</td>
<td>0.01</td>
<td>0.01</td>
<td>0.29</td>
<td>0</td>
<td>0.01</td>
<td>0.03</td>
<td>0.41</td>
<td>0</td>
<td>0.03</td>
<td>0.29</td>
<td>0.94</td>
<td>0</td>
<td>0.03</td>
<td>0.29</td>
<td>0.94</td>
</tr>
<tr>
<td>optimized profit</td>
<td>6,710</td>
<td>6,689</td>
<td>6,504</td>
<td>4,903</td>
<td>3,355</td>
<td>3,334</td>
<td>3,151</td>
<td>1,858</td>
<td>671</td>
<td>650</td>
<td>499</td>
<td>40</td>
<td>671</td>
<td>650</td>
<td>499</td>
<td>40</td>
</tr>
<tr>
<td>increasing rate</td>
<td>0.2981</td>
<td>0.2961</td>
<td>0.2787</td>
<td>0.1295</td>
<td>0.2978</td>
<td>0.2943</td>
<td>0.2949</td>
<td>0.0575</td>
<td>0.2978</td>
<td>0.2777</td>
<td>0.1298</td>
<td>-</td>
<td>0.2978</td>
<td>0.2777</td>
<td>0.1298</td>
<td>-</td>
</tr>
<tr>
<td>GBM base profit</td>
<td>4,908</td>
<td>4,900</td>
<td>4,830</td>
<td>4,121</td>
<td>2,454</td>
<td>2,446</td>
<td>2,375</td>
<td>1,667</td>
<td>491</td>
<td>483</td>
<td>412</td>
<td>-296</td>
<td>491</td>
<td>483</td>
<td>412</td>
<td>-296</td>
</tr>
<tr>
<td>adjusted threshold</td>
<td>0</td>
<td>0.26</td>
<td>0.26</td>
<td>0.52</td>
<td>0</td>
<td>0.26</td>
<td>0.29</td>
<td>0.39</td>
<td>0</td>
<td>0.26</td>
<td>0.32</td>
<td>0.73</td>
<td>0</td>
<td>0.26</td>
<td>0.32</td>
<td>0.73</td>
</tr>
<tr>
<td>optimized profit</td>
<td>6,738</td>
<td>6,716</td>
<td>6,517</td>
<td>4,957</td>
<td>3,369</td>
<td>3,347</td>
<td>3,155</td>
<td>1,860</td>
<td>674</td>
<td>652</td>
<td>496</td>
<td>0</td>
<td>674</td>
<td>652</td>
<td>496</td>
<td>0</td>
</tr>
<tr>
<td>increasing rate</td>
<td>0.3729</td>
<td>0.3706</td>
<td>0.3494</td>
<td>0.2029</td>
<td>0.3729</td>
<td>0.3684</td>
<td>0.3285</td>
<td>0.1160</td>
<td>0.3723</td>
<td>0.3494</td>
<td>0.2032</td>
<td>-</td>
<td>0.3723</td>
<td>0.3494</td>
<td>0.2032</td>
<td>-</td>
</tr>
</tbody>
</table>

experiment. In table X, the base profit is the predicted profit when the threshold set to 0.5 in the model. This value is equal to the expected profit in the test set II in table IX. The adjusted threshold is the threshold adjusted to maximize expected profit. When the campaign cost is zero, the adjusted threshold is zero because there is no cost to the user, so it is the best strategy to campaign for all users. In most cases, the adjusted threshold is lower than 0.5. The optimized profit is the predicted profit when the threshold is adjusted. The increasing rate is a variable by the gap of the optimized profit and the base profit. When we optimized the threshold, we could check that the profit increased by approximately 10% to 30%.

VII. DISCUSSIONS

In this section, we introduce a limitation of our method and an idea which we can further improve the profit optimization model in the future works.

A. Limitation of a binary classifier

The prediction model for long-term loyal customers had low precision in our experiments. To analyze the reason, we compared the characteristics of false positive users and true negative users. In Figure 12, left box plots show the changes of the playtime before and after predicted churn point. False positive users tend to decrease their play time later than before the predicted time. On the other hand, there is no significant difference in
Comparison of change in user status after prediction point. The time. (Viljanen et al. [26] and Periáñez et al. [25] are good references of applying survival analysis to churn prediction for online game. We also have the plan to apply survival analysis to our method in future works.

In the right of Figure 12, around 26% of false positive users churned within the subsequent four weeks, while only 8% of true negative users churned. Additionally, approximately 50% of false positive users decreased in their loyalty grade after the time of the expected churning, while loyalty grades of only 20% of true negative users decreased. These findings indicate that misclassified users are likely to leave the game soon, or at least to lose their game loyalty, even if they do not leave precisely within four weeks after prediction point.

Fig. 11. Comparison of CLV per user between true positive users of the test set I and the test set II. CLVs for most true positive users of the test set II have than 10, while most CLVs for the test set I are close to 0.

Fig. 12. (left) Comparison of change in playtime before and after the expected churn time. False positive users show a significant decrease in playtime after the time. (right) Comparison of change in user status after prediction point.

Fig. 13. The estimation of conversion rates by cost

B. Cost optimization

In section VI, we treated campaign costs and conversion rates as mutual independent variables. However, The conversion rate $\gamma$ is positively correlated with the incentive (i.e., campaign cost) for the user who would be a churner. In other words, the bigger the incentive is given to a user planning to leave, the more likely the user is to be retained. Consequently, conversion rate can be expressed as a function of the cost $C$. The function can be a logistic function considering the following assumptions:

1) As the cost of campaign increases, the conversion rate of users will increase.
2) At a certain point, no matter how much the cost is, the conversion rate will follow the law of diminishing marginal utility.
3) Until the campaign cost exceeds a certain level, the conversion rate of users will be small; however, at a certain point, the efficiency will be maximized. However, this efficiency will decrease again and converge to zero following 2).

We can define $f(C)$ which is a function of conversion rate for a cost as per Eq. 7 considering these assumptions.

$$ f(C) = \frac{\gamma_{\text{max}}}{1 + e^{-\epsilon x(\gamma_{\text{max}} - \rho)}} \quad C_{\text{max}} = \gamma_{\text{max}} \times CLV \quad (7) $$

where $\gamma_{\text{max}}$ is the maximum conversion rate that can be obtained through the campaign for churn prevention, and $C_{\text{max}}$ is the maximum campaign cost. $\rho$ and $\epsilon$ are hyper-parameters for adjusting the position and width of the interval, respectively, in which the efficiency is maximized.

$\rho$ is a value between 0 and 1. If $\rho$ is less than 0.5, it is efficient at low cost; otherwise it is efficient at a high cost. A large value of $\epsilon$ means that the conversion rate rises sharply in a narrow interval. For example, Figure 13 is a graph when $\gamma_{\text{max}}, \rho$ and $\epsilon$ are 0.2, 0.5 and 10, respectively. If it is possible to track changes of the conversion rate of churners with various reward conditions in a live game via A/B test, these parameters can be estimated approximately via least square method. We then can determine the optimal campaign cost by applying $f(C)$ to Eq. 5 (see Figure 14).

VIII. CONCLUSION

The purpose of churn analysis is to prevent losses caused by user churn. Consequently, churn prediction is required to not...
only improve prediction accuracy but also maximize expected benefits. To the best of our knowledge, there is no study for churn prediction considering the expected profit in the online game, while there are numerous studies in other fields [4], [6], [7].

We propose a churn prediction process considering the expected profit of the online game by referring to the existing research methods and apply it to the live game that has been in service for over nine years to verify its effectiveness. There are three main features of our proposed method. First, we define churn via analyzing the access patterns of users. Second, long-term loyal customers with a high benefit are identified and used for churn prediction. Finally, we calculate the expected profit per user via cost-benefit analysis and optimize the prediction model.

According to our experiments, only the users with little benefit are most likely to be detected when the churn prediction model is applied to the entire user. Therefore, considering the campaign cost, it is possible to incur a revenue loss. On the other hand, if the prediction model is applied only to loyal customers, high profit can be expected in most scenarios. Furthermore, optimizing the threshold of the predictive model can obtain an additional benefit of approximately 10% to 30% over the optimized model for accuracy.

The churn prediction process we propose partially applied to the game data mining competition at CIG 2017 [37]. The training data used in the competition was constructed using the process described in section IV-A and VI.

Furthermore, we discovered some useful characteristics of churners in an online game. According to our analysis, social factors have a significant influence on loyal customers’ churn in online games. The result supports the claims made by previous studies [18], [28], [29]. Additionally, we found that churners may show signals of a descending trend or increasing volatility of in-game activities from up to 10 weeks before leaving. We believe that this result will be a useful reference for predicting user churn in other online game services.

Finally, we consider that the profit estimation method we used will be necessary for other researchers to analyze user churn in online game services. However, there is still the limitation that sufficient verification has not been achieved in practice. We plan to verify and improve the proposed method in subsequent studies rigorously.

REFERENCES

REFERENCES


Yoonjae Jang received B.S. degree in Statistics from Carnegie Mellon University, and has worked as a member of Data Analysis and Modeling team at NCSOFT, South Korea since 2017. His research interest includes predictive modeling of game user behavior based on behavioral and survival analysis of in-game activities.

Huy Kang Kim received the B.S. degree in industrial management and the M.S. and Ph.D. degrees in industrial and systems engineering from KAIST in 1998, 2000, and 2009, respectively. He founded A3 Security Consulting, South Korea, the first information security consulting company, in 1999. He was the Technical Director and the Head of the Information Security Department, NCSOFT, from 2004 to 2010. He is currently an Associate Professor with the Graduate School of Information Security, Korea University. His research interests include solving security problems in online games based on the user behavior analysis and data mining.

Sungwook Kang received the B.S. degree in software engineering from Kumoh National Institute of Technology in 2014, and the M.S. degree in information security engineering from Korea University in 2016. From 2016 to 2018, He worked in Data Analysis and Modeling Team of NCSOFT. He is currently working as a member of Data Analysis Team in NEXON, Korea. His research interest includes data mining and machine learning.

Byungsoo Kang received the B.S. degree in Computer Science from Dankook University in 2014. He is currently working as a member of Data Analysis and Modeling Team of NCSOFT, South Korea. His research interest includes the game fraud detection using deep learning algorithm.

Boram Kim received the B.S. degree in computer science from Hankuk University of Foreign Studies in 2006, and the M.S. degree in statistics from Seoul National University in 2009. From 2009 to 2017, She worked in Data Analysis and Modeling Team of NCSOFT. She is currently working as a member of Data Analysis Team in NEXON, Korea. Her research interest includes data mining and artificial intelligence.

Eunjo Lee received the B.S. and M.S. degree in computer science from Soongsil University in 2003 and 2007, respectively, and is in Ph.D. candidate in information security from Korea University. He is currently the Leader of Data Analysis and Modeling Team in NCSOFT. His research interests include game fraud detection, user behavioral pattern modeling and time-series analysis in on-line games.