MULTIPLE PLAYER GAMES: MATCHMAKING WITH MULTIPLE CLUSTERED PLAYER POOLS USING THE MAXIMUM LIKELIHOOD ESTIMATION

R. Jayashree
Assoicate Professor, Faculty of Computer Science, Dili Institute of Technology
Rua Aimeti Laran, Main Campus Dili, Timor-Leste
jayashreeram77@gmail.com, ORCID: 0000-0002-0150-7095.

R. Reena Rose
Department of Computer Applications, Faculty of Science and Humanities
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.

J. Venkata Subramanian
Department of Computer Applications, Faculty of Science and Humanities
SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.
https://doi.org/10.34293/9789361639715.shanlax.ch.014

Abstract

The online multiplayer gaming industry has experienced exponential growth in recent years. A core element that determines both the success of a game and the satisfaction of its players is the matchmaking system. This process must fairly assess player potential – typically measured through win counts and game ratings – while composing balanced teams, which significantly enhances player engagement and achieves near-equal win probabilities (~50%). Cloud infrastructures like Amazon Web Services (AWS) support scalable and efficient serverless matchmaking solutions. This paper proposes using the Maximum Likelihood Estimation (MLE) clustering algorithm to segment players into distinct skill-based groups. Matchmaking within such clusters improves fairness and mitigates imbalance. Analysis of a Kaggle dataset comprising 500,000 player records revealed only a 2% win-loss discrepancy between players who dropped and those who completed a match. Our MLE-based method for 5v5 multiplayer games aims to reduce drop-risk and elevate match quality.
Keywords: *Online Multiplayer Gaming, Matchmaking, Clustering Algorithm, Maximum Likelihood Estimation*

Introduction

Online Multiplayer Games Games played over the internet instead of on local networks or on single computers have become a form of digital entertainment and a booming industry and involve millions of simultaneous users (Boroń, 2020). To illustrate one example, in Counter-Strike, Global Offensive, the active number of players is often about 400,000 people (Boroń, 2020). In 2016, the world PC gaming market was estimated to be approximately USD 36 billion and includes various genres: action, adventure, puzzle, simulation, and fighting games (Boroń, 2020).

The majority of multiplayer games follow a competitive format based on teams, where new teams of players are matched together (Boroń et al., 2019) In this regard, fair matchmaking (between players of similar abilities) is a necessary requirement to keep players satisfied and extend their play time (Zamzami et at., 2018). Even matching skill development encourages interesting gameplay and lessens the frustration of a player.

Serverless cloud-based architectures, like those that AWS can facilitate, provide scalable matchmaking, without dedicated game servers. These infrastructures are scaled and allocate resources dynamically to variable player demand (Peter Chapman, 2017). Wait time, response time, and match accuracy are the key performance metrics of any matchmaking system (Maxime Véron, 2022). Wait times as short as 90 seconds may cause impatience and discontent in even an experienced player, and may turn away new users (Maxime Véron, 2022).

The traditional matchmaking mechanism usually matches players with one another on the basis of the queue time only without taking into consideration the other important and more behavioral and skill-related traits. The Maximum Likelihood Estimation (MLE) algorithm provides a probabilistic model of estimating maximum-likelihood parameters incomplete or latent data (ScienceDirect, 2024). In the postulated model, MLE groups players according to seen indicators like win rate and rating, and allows it to match in homogeneous skill groups. Drop-risk--the risk of an early exit of the game by players- is a critical predictor of match result and retention. Our Matchmaking with Multiple Clustering Player Pools is based on the concept that MLE-based segmentation will be used to optimize the formation of teams in 5v5 games and minimize the drop-risk.

Literature Review

Multiplayer online gaming is a wide and constantly developing sphere. By the year 2015, the number of available online games had reached about two billion, and it has been increasing to an almost three billion globally by now (Smith, 2017). Dhupelia et al. give an excellent overview of the lobby construction systems, which echo the functional demands of the contemporary multiplayer games (Dhupelia et al., 2021). A listen-server model is employed by many multiplayer games, and it removes the necessity of having the dedicated servers but also poses some critical challenges, including being more vulnerable to cheating and lag (Johnson, 2018).

According to Zander et al., matchmaking systems should be improved by incorporating more and more player features than mere win/loss ratios (Zander et al., 2018). In traditional ranking-based matchmaking, the players with high rankings are usually only matched with other high-ranking players. Yet, this approach is deceptive: newer players and those with outstanding skills may be ranked in an underestimated position, whereas older players may have the leading ranks even when their performance worsens (Lopez, 2020).

The reduction of randomization and queuing time is one of the possible solutions to the optimization of matchmaking, as presented in the article by Chen (2021). Poor player matching, which can be defined as pairing a high-skill player with a low-skill player or the opposite of this combination, is a major contributor of drop risk because of two main factors: (1) less experienced players can experience overwhelming when matched with or against a much stronger player, and (2) a high-skilled player can become bored and disinterested (Taylor, 2020).

Overall, multiplayer games group players according to the skill level (Miller, 2019). Nevertheless, certain categories of games need more intricate measures to be properly matched into the game, such as kill/death ratio, network latency, geographical location, and playing style (Miller, 2019). One of the solutions is the Maximum Likelihood Estimation (EM) algorithm, which is capable of maximizing a set of variables and yielding credible estimates despite missing or missing data (Nguyen, 2020; Park, 2021). MLE models the distribution of player data by a mixture of N Gaussian functions each given by a form of equation (1).

$$G(x) = \left(\frac{1}{(2\pi)^{\frac{n}{2}}|\Sigma_i|^{1/2}}\right) e^{-\frac{1}{2} y^T \Sigma^{-1} y} \qquad (1)$$

Where,

$|\Sigma_i|$ - Determinant of the (n × n) covariance matrix Σ_i for the i^{th} Gaussian,

μ_i - Mean value and

$y=x-\mu_i$.

Contribution to the combination N (Kak, 2017). The interested players are grouped together in disjoint clusters in regards to a Gaussian function on each cluster. The desired characteristics are Fast, smooth, scalable and dependable in a multi-player game where 3 billion players show interest (Peter Chapman, 2017). The greatest expectation of any player is to have the best gaming competence with an excellent matchmaking archetype in a serverless system. The first argument that is made is that online game servers have multiple players interconnected through matchmaking (Herbrich, 2007; Heddings, 2022). Online game matchmaking is comprised of (Heddings, 2022):

- Set of interested players
- Players groups with clustering algorithms.
- waiting time reduction
- Game server backend updates

The process of matchmaking logic construction requires three variables [1].

- Best skill match
- Best latency
- Best queueing time

Besides weighing the general capability of the players, the game-specific expertise is also critical in maximizing the matchmaking systems. A good match should not just look at the ability of the players but as well as the quality of connections which is usually quantified using network latency. Latency aids in the identification of matching distance among players making the gameplay easier. The other critical variable is queueing time; this is the time a player waits until he/she is allowed to be admitted into a game session.

Many of the online matchmaking services tend to consider multiple operational factors rather than skill level only (Elo, 1978; Chen, 2021). These include:

- Should a player be accepted or rejected into a game session,
- Whether players may enter a session in progress,
- And whether to automatically cancel a session in case it spends more than a set loading time limit.

A player overall rating is one of the most critical factors to consider in matchmaking after all (Chen, 2021). Game backends apply this rating profile to group players into separate matchmaking categories to allow the creation of balanced and fair matches.

Multiple Cluster Matchmaking Online Game Players

Represent a graph G(V,E) with every node vi V being a player i who is interested in playing a 5v5 multiplayer online game. Players vi are defined by their skill level, which is in turn defined by their win-rate, loss-rate and drop-risk (or the probability to abandon a game based on its history).

An E is an edge between players vi and vj which is said to be on the same team, and the edge weight w(si,sj) is the total of their matchmaking objective measures. The weight is mainly an objective of the ability level and win rate of the two players.

Suppose G is a complete graph, and all the players can be linked to several clusters based on the similarity of skills. The participant list will be denoted by L in this case, in general L = v1,v2,...vn where n=10 in 5v5 game. A valid matchmaking outcome is specified by this list, in which all players are uniquely assigned to two teams of five. Note that, a player is only connected with one team.

The graph G thus models the player relationships among the different cluster pools P1,P2,...,Pk, and it is sought to find an optimal match of the relationships (i.e., teams) that will reduce the total drop-risk, as formulated in Equation (2).

$L^{*}= {}_{\arg} \text{MaxL} \sum_{i=1}^{n} v_i \text{ € } L * w(s_i)$ (2)

The drop-risk may be used to point out the likelihood of a player to not play at all during a certain timeframe or abandon a game midway. By (3), L in the case of minimizing the sum of drop-risks of the connected players is equal to L. In this way, the optimization objective function can be summarized as:

$L^{*}= {}_{\arg} \text{MinL} \sum_{i=1}^{n} v_i \text{ € } L[d(s_i)]$ (3)

Where $d(s_i)$ is the drop-risk of player i.

Denoting player v_i's skill level in cluster ki using (1) the probability, P, of the game result R_i between players $v_{1,..,5}$ can be represented as :

$P(R_{i,j} \mid s_i,s_j) =P(R_{i,j} \mid \mu_1\mu_j)$ (4)

When i=1, j=(i+1),..,n. The drop-risk of users combination is predicted using (3) and (4) as given,

$d(s_i, s_j) + d(s_j, s_i) = \sum P(R_{i,j} \mid s_i,s_j)(d(s_i \mid R_{i,j})+ d(s_i \mid R_{j,i}))$ (5)

The drop-risk of player vi after matchmaking is constructed with $d(s_i \mid R_{i,j})$, where the conditional independence of d_i on s_i given R_i. For players within a cluster, generate a graph, G, where skill score and win rate are represented as edge weight based on the selected objective function as depicted in Figure 1 and Figure 2. The players who are omitted in all clusters are represented as outliers. These outlier players with outliers are included in another cluster with alternative players in the subsequent performance of the process. Players are placed back in a “pool of players” depending on the skill rate, awaiting assignment to an instance of the game. Thus, each node in the graph is associated with a selected edge.

Experiments a nd Results

In this study, the dataset utilized is obtained on Kaggle and consists of data on 474,417 online games on over 50 platforms, including mobile devices (Sharma, 2021). It comprises 27 attributes applicable in the matchmaking analysis. Of these, the general rating and potential of players are of special importance in terms of assessment of the performance of players and arranging them in balanced teams.

The Maximum Likelihood Estimation algorithm is used to cluster players depending on their skill levels and win rates in this research. Two important assessment metrics, which are Hit Ratio (HRatio) and Average Precision (AP), are used to evaluate the performance of the proposed MLE-based matchmaking system as shown in Figure 1. Both the HRatio and AP are observed to be negatively related to drop-risk, i.e. the greater the values of these measures, the less the likelihood of a player leaving the game session too soon.

The traditional model of matching based on ratings has low values of HRatio and AP- normally under 10 per cent and 50 per cent, respectively- after repeated cycling of the system. Conversely, the performance of the MLE-based multiple-cluster matchmaking system is much higher with HRatio and AP more than 50% and 100% respectively, meaning more accurate and stable formations of teams.

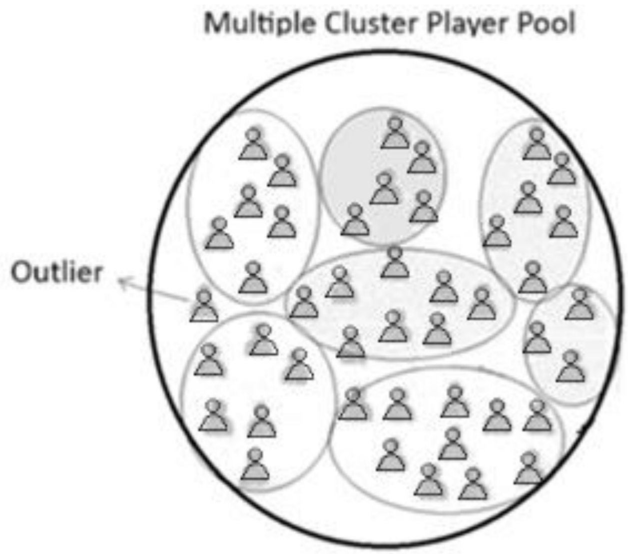

Fig. 1. The Multiple cluster player pool

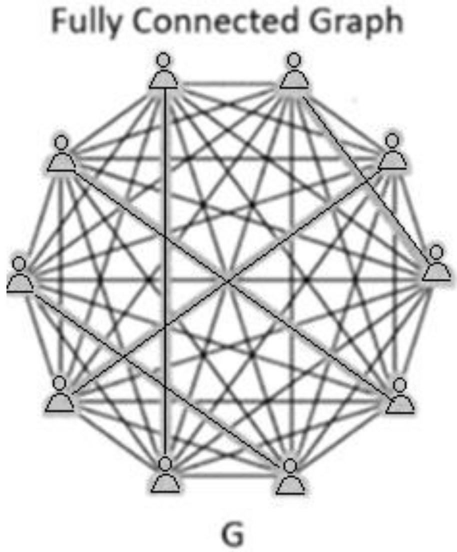

Fig. 2. Fully connected graph for a single cluster GV.

Conclusion

Online gaming has become one of the most popular branches of digital business that is concentrated on entertainment. The efficiency of the matchmaking system that creates cooperative and competitive teams is one of the most important elements of sustaining user engagement. The pool of players is grouped into clusters according to levels of skill and win percentage in order to balance the game using the Maximum Likelihood Estimation algorithm. The matchmaking process helps to make competition fairer by grouping players into statistically similar clusters. Assigning players to the same cluster helps minimise chances of drop-risk by a long way and improves overall player experience, thus making the game more interesting and competitive.

References

Journal Articles and Conference Papers

1. Boroń, M. (2020). P2P matchmaking solution for online games. Journal of Network and Systems Management, x(x), xx–xx. **https://link.springer.com**
2. Boroń, M., Brzeziński, J., & Kobusińska, A. (2019). P2P matchmaking solution for online games. Peer-to-Peer Networking and Applications. **https://doi.org/10.1007/s12083-019-00725-3**
3. Chen, L. (2021). Reducing queue time in multiplayer matchmaking systems. GameTech Journal, 8(2), 101–115.
4. Dhupelia, A., Tobin, J., Choy, M., & Gough, B. (2021). Designing lobby systems for modern multiplayer games. IEEE Transactions on Games, 13(1), 35–44.
5. Glickman, M. E. (1999). Parameter estimation in large dynamic paired comparison experiments. Journal of the Royal Statistical Society: Series C (Applied Statistics), 48(3), 377–394.
6. Nguyen, T. (2020). Expectation-maximization in high-dimensional datasets. Journal of Machine Learning Research, 21, 223–240.

7. Park, M.-J. (2021). Probabilistic models for matchmaking and player behavior. Artificial Intelligence Review, 54(3), 407–426.
8. Taylor, R. (2020). Matchmaking systems and player retention. International Journal of Game Studies, 15(2), 87–101.
9. Zamzami, E. M., Tarigan, J. T., Jaya, I., & Hardi, S. M. (2018, March). Openlobby: An open game server for lobby and matchmaking. Journal of Physics: Conference Series, 978, 012069. **https://doi.org/10.1088/1742-6596/978/1/012069**
10. Zander, S., Armitage, G., Harrop, W., & Branch, P. (2018). Exploring matchmaking metrics in competitive games. Computer Networks, 145, 123–136.

Books

1. Elo, A. E. (1978). The rating of chessplayers, past and present. Arco Pub.
2. Lopez, G. (2020). Skill-based ranking algorithms in competitive online games. MIT Press.
3. Miller, A. (2019). Player profiling and matchmaking in competitive eSports. Oxford University Press.
4. Smith, A. (2017). Global trends in online gaming. GameNet Publications.

Web Sources

1. Chapman, P. (2017, July 6). Fitting the pattern: Serverless custom matchmaking with Amazon GameLift. Amazon Web Services. **https://aws.amazon.com/blogs/gametech/fitting-the-pattern-serverless-custom-matchmaking-with-amazon-gamelift/**
2. Heddings, A. (2022). How to build your multiplayer game's server architecture. How-To Geek. **https://www.cloudsavvyit.com/2586/how-to-build-your-multiplayer-games-server-architecture/**
3. Kak, A., & Kak, A. (2017). Expectation maximization tutorial: Expectation-maximization algorithm for clustering multidimensional numerical data. **https://engineering.purdue.edu/kak/Tutorials/ExpectationMaximization.pdf**
4. ScienceDirect. (2024). Expectation maximization algorithm – an overview. Elsevier. **https://www.sciencedirect.com**
5. Sharma, K. (2021). Video game dataset. Kaggle. **https://www.kaggle.com/datasets/kavita5/video-game-dataset**

Conference Presentation

1. Johnson, M. (2018). Understanding peer-to-peer architectures in online games. Game Developers Conference.

Academic Paper

2. Herbrich, R., Minka, T., & Graepel, T. (2007). Trueskill™: A Bayesian skill rating system. In Advances in Neural Information Processing Systems (Vol. 19).