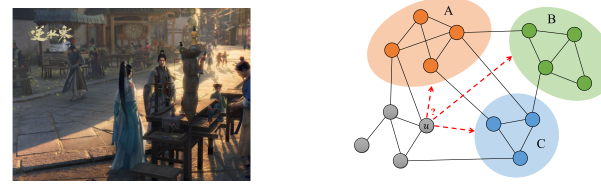
Find Your Organization in MMORPGs

Qilin Deng, Minghao Zhao, Kai Wang, Runze Wu, Xudong Shen, Jianrong Tao, and Changjie Fan

***Abstract*—Social relationships are the basis for communication and collaboration between players in many online games. In this article, we propose a machine-learning-based approach to model the relationship between players and guilds in online games. Our approach combines deep learning techniques with useful prior expert knowledge, where the core component is a graph convolutional network that is designed to utilize both social relationships and behavior preferences of players. For each player in the game, the model is trained to estimate the likelihood of whether the player matches the guild, which enables rapid matching of players and guilds via recommendation. The proposed approach is evaluated on an industrial dataset collected from a popular online game and also deployed in the game as a basic component of the social system. Experimental results show that our approach is not only intuitive but also very superior to other baseline methods.**

***Index Terms*—Deep learning, graph convolutional networks, guild recommendation, online games, recommender system, social modeling.**



(a) (b)

Fig. 1. (a) Justice—a breathing virtual world. (b) Illustration of the social structure of *Justice Online* in terms of friendship between players and membership between players and guilds. Nodes, colored areas, and black lines represent players, guilds, and friendships, respectively.

I. Introduction

AS AN increasingly prevalent modern entertainment, online games play an important role in people's daily lives [1], especially for young people. Among them, massively multiplayer online role-playing games (MMORPGs), such as *World of Warcraft*[1] (*WoW*) and *AION*,[2] are one of the most popular game types in which a very large number of players interact with each other within a virtual world. As in all role-playing games or RPGs, the player assumes a virtual character and controls the character's activities and behaviors in the virtual world of the game. MMORPGs are distinguished from other online RPGs by the number of players able to interact together, and by the game's persistent world, which continues to exist and evolve while the player is offline and away from the game.

The persistent world and massive player interaction characteristics of MMORPGs make the existence of stable social relationships a natural result. Sometimes the relationships between players within the game even spill over into friendships in the real world. Fig. 1 illustrates the social structure in MMORPGs in terms of friendship between players and membership between players and guilds. Although there are usually several other social relationships between players in online games, such as teammates, chatting and trading partners, we only consider the friendship relation in this work, and leaving the extension of other relationships as future work. As illustrated in Fig. 1(b), the guilds (e.g., A, B, and C, represented as colored areas) consists of the affiliated players, while some players (like u) may not belong to any guild currently. In addition, there is an in-game friendship between players (represented as black lines in the figure).

As one of the core social elements of MMORPGs, a guild is an organized group of like-minded players, which provides a scene to facilitate communication between players. These characteristics make the guild system an effective tool to promote collaboration and social networking. For example, the cooperation between players to accomplish difficult tasks is often an integral mechanic of gameplay in online games, and it is usually easier and more efficient to convene guild members to form a team to complete some kind of teamwork such as difficult quests. In addition, skilled players can help freshmen (new players) to better understand the game mechanics and quickly master the skills of the character. To a certain extent, such social interactions in the form of in-game communication can provide participating players with a sense of belonging and accomplishment. In addition, it has been revealed that stable guild relationships have a great impact on maintaining player retention and improving player activeness [2], [3].

From the perspective of sociology, human behavior is personalized and greatly influenced by social factors, both in real life and on social media platforms (e.g., Facebook, Twitter, and YouTube) [4]–[6]. In online games (especially MMORPGs), the social preferences of players are also personalized and diverse, which could be reflected in several social aspects [3], including making new in-game friends, working with specific teammates,

Manuscript received 25 September 2020; revised 3 March 2021; accepted 20 July 2021. Date of publication 12 August 2021; date of current version 15 September 2022. *(Qilin Deng and Minghao Zhao are co-first authors.) (Corresponding authors: Runze Wu; Xudong Shen.)*

The authors are with the Fuxi AI Lab, NetEase Games, Hangzhou 310052, China (e-mail: qlqldeng@163.com; zhaominghao@corp.netease.com; wangkai02@corp.netease.com; wurunze1@corp.netease.com; hzshenxudong@corp.netease.com; hztaojianrong@corp.netease.com; fanchangjie@corp.netease.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TG.2021.3104319.

Digital Object Identifier 10.1109/TG.2021.3104319

[1][Online]. Available: https://worldofwarcraft.com/

[2][Online]. Available: https://www.aiononline.com/

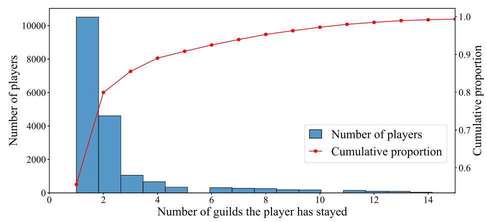


Fig. 2. Distribution of the number of guilds the player has stayed.

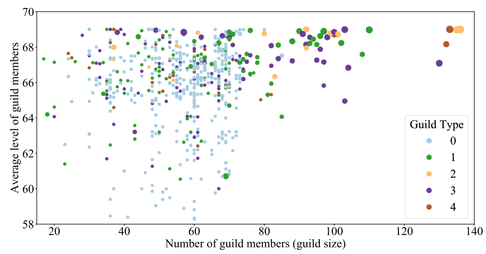


Fig. 3. Distribution of guilds with different attributes. The size of the mark represents the level of the guild itself, i.e., the larger the mark, the higher the level. The *guild type* means the overall impression of the guild, such as fighting and social.

and becoming members of a guild. In terms of the guild affiliation, some players prefer guilds with a good atmosphere, where there may be many daily chats and activities within the guild. Some players like aggressive guilds because they can participate in hunting stronger monsters to get more rewards. On the contrary, other players may be "lazy" players, meaning that they do not care much about internal events in the game, and often do not have a strong desire to participate in the game tasks. Fig. 2 shows the distribution of the number of guilds the player has stayed for a single game server, where the time interval is from the creation of the account to the reporting time, and the statistics show that more than 20% of players have stayed in more than three different guilds. Considering that rejoining a guild usually means additional social costs, such as being familiar with guild activity patterns and establishing contact with other members of the guild, finding a guild suitable for a specific player through trial-and-error will significantly increase the player's social cost and harm the gaming experience. On the other hand, the guild itself also has its own tenet, social atmosphere and member structure, showing diversified characteristics. Fig. 3 shows the distribution of guilds with different attributes for a single game server, where the *guild type* means the overall impression of the guild, such as fighting, social, casual, etc. For these reasons, it is necessary to recommend a best-matching guild to new players or regular players who do not currently belong to any guilds, since it could reduce the trial-and-error social cost for them to join a guild blindly.

In this article, we aim to model the affiliation relationship between players and guilds in MMORPGs. Taking Fig. 1(b) as an example, we want to know which of the three guilds A, B, C matches player u best. Technically, we consider the matching relationship between players and guilds as a recommendation problem, and further formalize it as a link prediction problem on the social graph. Our approach combines deep learning techniques with useful expert knowledge from experienced game developers. The central component is a graph convolutional network that is trained to estimate the likelihood, for each player in the game, of whether the player matches the guild. The neural network is designed and trained to utilize both social relationships and behavior preferences of players, which enables effective matching of players and guilds through recommendation and improves the social efficiency of players' game experiences.

The presented approach is evaluated on an industrial game dataset collected from *Justice Online*,[3] a heavily story-driven *Wuxia* MMORPG developed and published by *Netease Games*,[4] where players could impersonate specific roles and their behaviors data are extensively tracked by the game server. The offline experimental results verify that the proposed approach substantially outperforms other baseline methods. To further evaluate the effectiveness of our proposed method, we also deploy the guild recommendation system in the virtual world of the game as a basic component of the social system. The online A/B testing results demonstrate that the presented approach substantially outperforms the heuristic recommendation rules, improving the quality of the guild suggestions and the social efficiency of players. The main contributions of our work are summarized as follows.

1) We highlight the necessity of guild recommendation in the online games, especially MMORPGs, so that players could experience the game more efficiently, brings a better gaming experience to players.
2) We propose a novel framework to address the guild recommendations problem, which combines effective deep learning techniques with useful expert knowledge from experienced game developers.
3) We conduct offline and online experiments on the dataset collected from a popular MMORPG. The results demonstrate that the proposed framework significantly outperforms the baseline approaches. Further, the guild recommendation system has been deployed into production for more than one year in the game, and the launch of the system yields better guild suggestions and brings a positive impact to the game.

The rest of this article is organized as follows. First, we introduce some preliminaries that are the foundation of this work in Section II. Then we introduce the game dataset that is used in this research work in Section III. In Section IV, we describe our proposed recommendation framework in details. After that, we show our experiment settings and results to verify the proposed models in Section V. Finally, Section VI concludes this article.

II. Related Work

In the following sections, we briefly review some related research works on social relationship modeling in online games and recommender systems.

[3][Online]. Available: https://n.163.com/
[4][Online]. Available: https://www.neteasegames.com/

A. Social Modeling in Online Games

There are many social elements between players in MMORPGs, such as friends, teams (groups) and guilds. For example, individual players could make in-game friends with each other, while teams usually exist only for players to complete quests or missions together. Different from the teams, the guilds are long-lived player associations, and usually have some specific internal organizational structures. The social behavior of game players is also built on and closely related to these elements, and the modeling of players' social behavior in online games has always been an interesting topic in the game and social research works [3], [7], [8]. To some extent, the results of such research works can help us better understand the social behavior of players, discover interesting behavioral patterns, and further improve game designs [9], [10].

The rationality of guild design in online games is extensively discussed in the literature [2], [7], [10], [11]. The authors of [2] use longitudinal data collected directly from the game to examine play and grouping patterns in *WoW*. Their results indicate that longer-lived player associations (the guilds) have significant impacts on play patterns. For example, guilds facilitate the formation of groups, encourage players to play more often and more regularly, and act as an important and ever-present source of support and socializing. They also point out several challenges that can affect the growth and longevity of guilds, such as the absence of good social navigation tools when managing a guild, and leveling which also has damaging impacts on the game's social fabric. In some cases, differences in levels can be enough to reduce a guild's cohesion and probably play a role in the heavy "churn rate" among the members. The work of [11] tries to explain the success or failure of game guilds in *WoW*, based on some structural properties of these guilds, such as guild size, social connectivity, and centrality. Their findings show that there might be a hard limit on the size of a viable organic group in online games as in other online social spaces. The pioneering work in [12] investigates the social roles that emerged from the players' behavior and interaction within a guild of *WoW*. From the social network analysis perspective, they identify and analyze three major social roles with distinct interaction styles and network properties. Although these efforts have been used to study the social behaviors in online games, they basically follow the paradigm of descriptive statistics, focusing on the correlation analysis from game data to justify guild mechanics, and few of them have tried to improve the gaming experience from the perspective of intervention.

B. Recommender System

As an effective technology to alleviate information overload, recommender systems have now been widely used in online services such as e-commerce, advertising, and social media platforms [13], which are ubiquitous in our daily life. The core goal of a recommendation system is to estimate how likely a user will be interested in an item, either based on the historical interactions like clicks and purchases [14], [15], or based on features such as user gender or item category [16], [17]. For example, the considerable work on collaborative filtering (CF) [14] addresses the recommendation problem by assuming that behaviorally similar users would exhibit similar preferences on items. The research area most relevant to the specific problem addressed by this article is social recommendation [18]. In fact, if we aim to leverage the social relationships established between players to predict the guild affiliation for the target player, then we can formalize it as a typical social recommendation problem by treating the guilds as *items*. Social recommendation aims to leverage the relevance between users implied by social networks to recommend potential items for specific users [18]. It is based on the assumption that people tend to share similar interests with their friends, which applies almost equally to social networks in the virtual world of online games. Early works on social recommendation are mainly based on MF with trust-aware regularization to take both ratings and trust into considerations [19], [20]. Nowadays, graph convolutional networks (GCNs) [21], which can utilize node attributes and network structures simultaneously, have been proved to be effective in node representation learning for graph-structured data. GCNs have also brought new insights into the development of social recommendation and several GCN-based methods have been proposed to leverage social relationships between users for improved recommendation performance [22], [23].

In this article, we study the modeling of social relationships in the environment of online games with graph convolutional networks. Specifically, we focus on the problem of guild recommendation, i.e., recommending the most suitable guilds for new players who have not yet joined any guild or regular (old) players who just want to switch guild membership because of mismatching with the previous guilds. Different from existing works that focus on the correlation analysis, we design a whole framework named GuildNet to provide a guild recommendation service for online games. We hope that this system serves as an additional tool that could be used to better support gaming communities in online games. Methods discussed here can be adapted to similar scenarios with rich user behaviors and social networks, such as personalized teammate recommendation in MMORPGs, group recommendation on social media, etc. To the best of our knowledge, this is the first work that proposes a deep learning method to solve the guild recommendation problem in online games, using not only the behavioral preference characteristics of players, but also the social network constructed from the virtual world in online games.

III. Dataset

We collect players and guilds data from the popular MMORPG *Justice Online*, where players' behaviors data are extensively tracked by the game server. The collected game data can be divided into feature data and relationship data.

A. Features

1) Basic Portrait: We use some basic player portraits, such as game level, role profession, equipment score, and guild portraits, such as guild type, guild size, guild level. Note that some guilds have some restrictions when recruiting members, e.g., players with specific professions, or players with high level and high

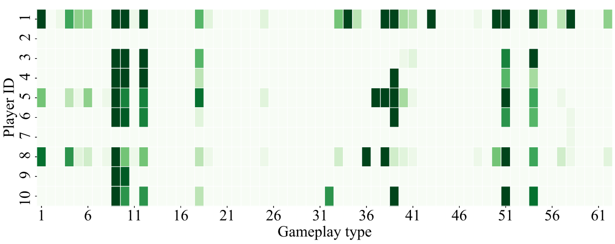


Fig. 4. Gameplay statistics of individual players. Each row (y-axis) represents a player and each column (x-axis) represents a specific gameplay in the online game. The darker the color, the higher the frequency.

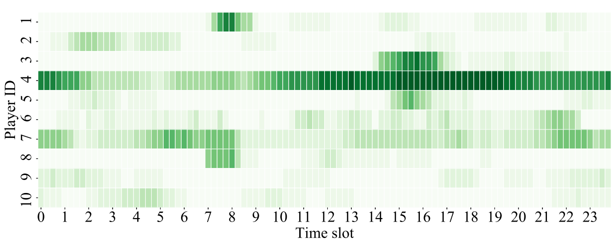


Fig. 5. Playtime statistics of individual players. Each row (y-axis) represents a player and each column (x-axis) represents specific time interval of the day (We divide the time of day into time slots by 15 minutes). The darker the color, the higher the frequency.

equipment score. These restrictions make the basic portrait features useful for recommendation. We additionally consider three different variables, namely total playtime (in hours), lifetime (in terms of days since the first login) and total expenditure or lifetime value (LTV) in real (or in-game) currency, to describe player engagement [24].

2) Gameplay Preference: Generally, there are many types of gameplay (or game activities) in online games in order to enrich the game story and attract different types of players, e.g., main and side quests, daily and nondaily quests, etc. Fig. 4 shows the different gameplay preferences of ten players in the game (we consider a total of 62 types of gameplay), which indicates the significant differences between individual players. Guilds frame the game experience of players by providing a stable social backdrop to many gameplay, and their members tend to group with others more often and play longer than non-affiliated players [9]. Therefore, it is very helpful to consider the player's gameplay preferences in detail for guild recommendation problems.

3) Playtime Preference: We divide the day into 15-minute intervals, i.e., 96 time slots, which constitutes 96 playtime features of players. We get the player's time preference vector according to the actual online time of the player. For example, for a certain time slot, if the player is online, the corresponding element in the time preference vector is 1, otherwise it is 0. Fig. 5 shows the different playtime preferences of ten players in the game. Some players like to play the game from 7 to 9 A.M., some players from 2 to 5 P.M., while some players prefer late at night.

4) Player Tags and Guild Tags: we also create some high-level tags for players and guilds in online games. These manually created tags describe basic characteristics of players and guilds, and are used to help players evaluate the guild's portrait, i.e., high combat power and enthusiastic captain for players, good

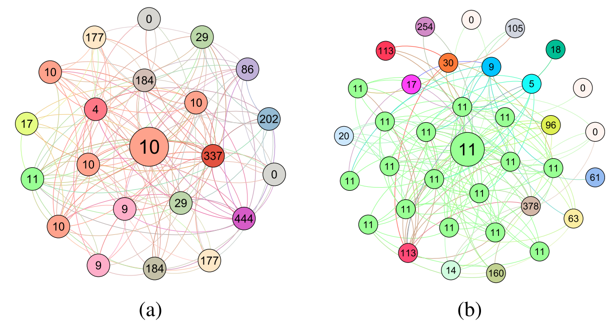


Fig. 6. Guild distribution of players' friends (Nodes represent players, numbers represent guild IDs, and lines represent friendships).

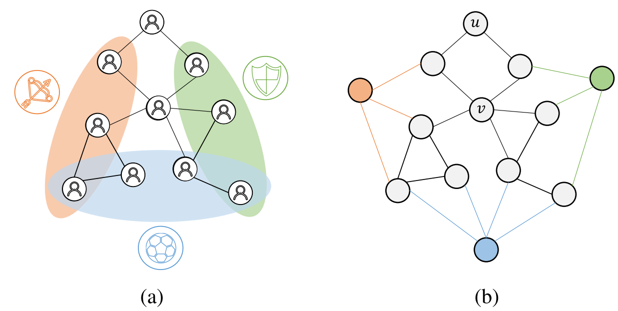


Fig. 7. Left part is the abstract social structure in terms of friendship and membership and the right part is the corresponding heterogeneous graph constructed with multiple social relationships from the left part.

atmosphere, frequent activities, and generous rewards for guilds. We give several representative tags and their corresponding meanings in Table I.

B. Relationships

The two types of social relationships (networks) considered in this study are the friendship between players and the membership between players and guilds.

1) Membership: Member relationships can be represented by a bipartite graph. Nodes in the bipartite graph represent players and guilds, respectively, and edges connecting a player and a guild represent that the player has joined the guild (is a member of the guild, once or now).

2) Friendship: Friend relationships can be represented by a *simple* graph. Nodes in the graph represent players, and edges represent game friends between players. Fig. 6 shows the *Ego Network*, i.e., the sub-network containing that node, its neighbors, and all the edges between them in its neighborhood, of two player nodes (the central large nodes) in *Justice Online*. As can be seen from Fig. 6(a), the guilds of this player's friends is distributed in several guilds, i.e., 9, 10, 29, 177, 184. This phenomenon is more obvious in Fig. 6(b), where most of them are concentrated in the guild 11. From these two examples, it can be seen that the friendship between players has a great influence on the choice of players' guilds and reflects the guild membership to some extent.

Instead of processing these two graphs separately, we combine them together to build a heterogeneous graph, which contains multiple types of nodes and edges, as illustrated in Fig. 7.

TABLE I
SOME EXAMPLES OF PLAYER TAG AND GUILD TAGS

Tag	Description	
	Name	Meaning
Player	Highly active	Participate in guild activities and leagues more than a certain number of times per week
	High combat power	The combat ability value ranks in the top positions in the whole server
	High fund contribution	Contribute more than a certain amount of funds (in-game currency assets) to the guild every week
	Enthusiastic captain	Served as the captain of daily quests more than a certain number of times per week
	Scalping master	Complete transactional tasks more than a certain number of times per week
Guild	Good atmosphere	Strong atmosphere of chat and social gameplay among guild members
	Frequent activities	More than a certain number of guild activities per week
	Generous rewards	More than a certain number of bounty tasks within the guild per week
	Have friends	More than a certain number of friends are in the guild
	Powerful guild	The league ranking of the guild is high in the whole server

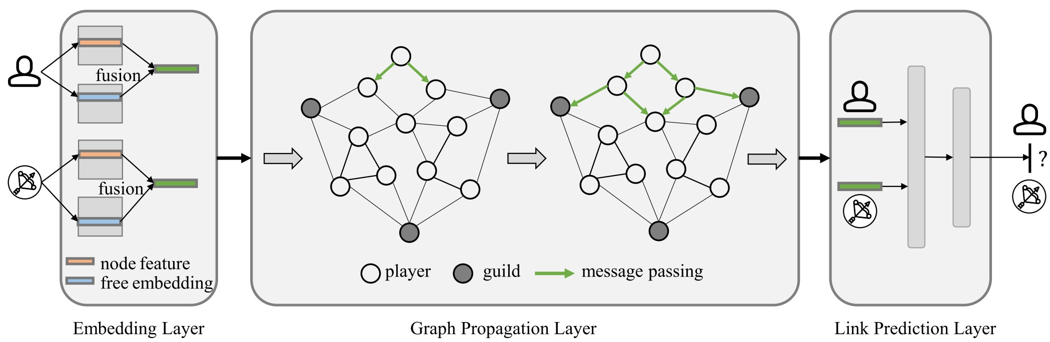


Fig. 8. Illustration of the overall model architecture of GuildNet. The embedding representations of players (light gray nodes) and guilds (dark gray nodes) are refined with multiple graph propagation layers (green arrows represent the direction of message passing), whose outputs are concatenated to make the final link (membership) prediction.

Specifically, on the basis of the original friendship network, we add a virtual node for each guild, and connect an edge between the player node and the guild node if the player recently joined the guild. Note that players may joined several guilds for a period of time. For example, as shown in Fig. 7, player node u and v have not joined any guild, while other players have joined one or more guilds.

Here, it is worth mentioning that we choose these features (or factors) and the friendship based on our own observations of the game and expert knowledge from experienced game developers. We believe that these features can well represent the player's in-game behavior patterns, and are the main factors that may influence the player's guild choice. On the one hand, the interdependence between players within the guild leads to coordinating activities towards common goals, such as completing specific game tasks, which could be reflected by gameplay preferences. On the other hand, only when players have similar online time preferences can they have more interactions with other members of the guild, which is consequently beneficial to the development of group cohesion and group identification. Moreover, research in the field of social networking has found that group members have similar behavior patterns, and the guild entity itself is a group organization. It is precisely because of the diversity of players' preferences (e.g., gameplay and playtime preferences) that they have created a personalized demand for group organizations. This makes using these features to provide guild recommendations a very meaningful attempt, and the final experimental results also validate our ideas. Besides, as noted by [7], the social aspects are the core elements of MMORPGs. The social relationships (especially friendship) among game players often develop into an affective bond, increasing player loyalty, and leading players to spend more time in the virtual world. In particular, friends can help new members form the initial affinity groups within the guild.

IV. PROPOSED MODEL

In this section, we first define the problem formally, then detail each component of the proposed model. We name the proposed model as *GuildNet*, which means a neural network model for guild recommendation, and the overall architecture is shown in Fig. 8.

A. Problem Formulation

First, we introduce the notation conventions. We use bold uppercase letters to denote matrices (e.g., $\mathbf{W}$), bold lowercase letters to denote vectors (e.g., $\mathbf{w}$), and nonbold letters to denote scalars or indices (e.g., w). The uppercase calligraphic symbols (e.g., $\mathcal{W}$) stand for sets.

Suppose we have players $\mathcal{U} = \{u_i | i = 1, 2, \ldots, M\}$ and guilds $\mathcal{V} = \{g_j | j = 1, 2, \ldots, N\}$, where the size of these two sets is $|\mathcal{U}| = M, |\mathcal{V}| = N$ respectively. We also have following two interaction graphs introduced in Section III-B.

1) Player-Guild Interaction Graph: A player can have an interaction with a guild, which corresponds to guild membership. The interaction is represented as a binary variable in the adjacency matrix of the graph, i.e., 1 means the existence of the interaction and 0 otherwise. We use $\mathbf{A}_m = [a_{ij}]_{M\times N}$ to denote the player-guild interaction matrix.

2) Player-Player Interaction Graph: The interaction graph between players represents friendship. Since it makes sense to distinguish between strong and weak ties, we use friendship degree, which represents the degree of intimacy between players in the game and is normalized to $[0, 1]$, to represent elements in the adjacency matrix of the graph. We use $\mathbf{A}_f = [a_{ij}]_{M\times M}$ to denote the player-player interaction matrix.

Based on these interaction data, we build a player-guild heterogeneous graph $\mathcal{G} = (\mathcal{U}, \mathcal{V}, \mathcal{E})$, where $\mathcal{U}$ denotes player nodes, $\mathcal{V}$ denotes guild nodes and $\mathcal{E}$ is the corresponding link edges between these nodes, including friendship edge and membership edge. The adjacency matrix of graph $\mathcal{G}$ could be constructed as

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_f & \mathbf{A}_m \\ \mathbf{A}_m^T & \mathbf{0} \end{pmatrix}. \tag{1}$$

Because players and guilds have their own features, e.g, the playtime and gameplay features, the corresponding nodes in the graph have associated features, forming a feature matrix $\mathbf{X}$. We define the recommendation problem as estimating the likelihood of the player u belongs to the guild g, based on their features and relations. Formally, given the interaction graph $\mathcal{G}$ and associated feature matrix $\mathbf{X}$, we propose a neural optimization model to learn a match function f as follows:

$$\hat{p} = f(u, g|\mathcal{G}, \mathbf{X}; \Theta). \tag{2}$$

Here, Θ is the parameters of the neural model to be learned and $\hat{p}$ is the predicted likelihood that the player u matches the guild g, which will be specified in the following subsections.

B. Embedding Layer

Following the existing research works, we define $\mathbf{e}_u \in \mathbb{R}^d$ and $\mathbf{e}_g \in \mathbb{R}^d$ as the embedding vectors of player node u and guild node g respectively, where d is the embedding size. Since node features (i.e., portraits, gameplay and playtime preferences) $\mathbf{x} \in \mathbb{R}^f$ are available in our problem, we could transform the input features into higher-level features by a learnable linear transformation (potentially applying a nonlinearity) to obtain better representational capacity. To that end, as an initial step, a type-specific shared linear transformation is applied to all of the graph nodes. Here, the node type includes two types: the player node and the guild node. After that, we could combine the embedding vectors and transformed feature vectors to serve as the final input features. For player node u, we have

$$\mathbf{h}_u^{(0)} = \text{COMBINE}(\mathbf{h}_u, \mathbf{e}_u)$$

where

$$\mathbf{h}_u = \text{ReLU}(\mathbf{W}_u\mathbf{x}_u + \mathbf{b}_u). \tag{3}$$

Here, $\mathbf{x}_u$ is the feature vector of player node, $\mathbf{W}_u \in \mathbb{R}^{d\times f}$ is the type-specific weight matrix for player nodes, which is used to map players to a common vector space $\mathbb{R}^d$, $\mathbf{b}_u \in \mathbb{R}^d$ is the bias, and $\text{ReLU}(\cdot) = \max(0, \cdot)$ is the activation function. $\mathbf{e}_u$ represents the corresponding player's free embedding vector which acts as complementary features and can be optimized by back-propagation algorithm. The COMBINE function is usually add operation, although other operations are also possible. Likewise, we can get the input representation $\mathbf{h}_g^{(0)}$ of guild nodes, which is omitted here for redundancy.

C. Graph Propagation Layer

Inspired by recent graph convolutional networks (GCNs) that operate directly on graph-structured data, we expand it to process the heterogeneous graph data. It learns hidden layer representations that encode both local graph structure and features of nodes, providing better node representations for the downstream task, such as node classification and link prediction. Its propagation rule can be formulated as $\mathbf{Z} = f(\mathbf{X}, \mathbf{A})$, where $\mathbf{X}$ denotes node feature matrix, $\mathbf{A}$ denotes adjacency matrix of the underlying graph structure, and $\mathbf{Z}$ denotes the encoded node representation. The single-layer propagation rule is

$$\mathbf{Z} = f(\mathbf{X}, \mathbf{A}) = \text{ReLU}(\hat{\mathbf{A}}\mathbf{X}\mathbf{W}). \tag{4}$$

Here, $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}$ with $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ and $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$, $\mathbf{W}$ is an input-to-hidden weight matrix and $\text{ReLU}(\cdot) = \max(0, \cdot)$ denotes an element-wise nonlinear activation function. The $\hat{\mathbf{A}}$ can be calculated in a pre-processing step in which the adjacent matrix $\mathbf{A}$ used to compute $\hat{\mathbf{A}}$ is defined as formula (1).

The standard graph convolutional layer performs local operations that only take the (direct) one-hop neighbors of nodes into account, whereby the same transformation is applied across all nodes in the graph. This type of local graph convolution can be seen as a form of message passing, where vector-valued messages are being passed and transformed across the graph structure. In our case, following the process of above single graph convolution layer, we can stack several layers to learn better hidden representations for graph nodes, with the following layer-wise propagation rule:

$$\mathbf{H}^{(l+1)} = \text{ReLU}(\hat{\mathbf{A}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}) \tag{5}$$

where $\mathbf{H}^{(l)}$ denotes input representation matrix of (player and guild) graph nodes in the lth layer and $\mathbf{H}^{(0)}$ is given by formula (3). $\mathbf{W}^{(l)}$ denotes a layer-specific trainable weight matrix of lth graph propagation layer and $\mathbf{H}^{(l+1)}$ is the output representation matrix in the $(l+1)$th layer. The intuition behind this is that at each layer, nodes aggregate information from their local neighbors, and as this process iterates, nodes incrementally gain more and more information from further reaches of the graph.

Since prior works have revealed that stacking deep GCN layers can damage the performance [25], [26], we consider a two-layer GCN to learn player and guild representations. We first calculate $\hat{\mathbf{A}}$ in a pre-processing step, and the forward model of each layer takes the following forms:

$$\mathbf{Z}^{(\mathbf{0})} = \mathrm{ReLU}\left(\hat{\mathbf{A}}\mathbf{H}^{(0)}\mathbf{W}^{(0)}\right)$$
$$\mathbf{Z}^{(\mathbf{1})} = \mathrm{ReLU}\left(\hat{\mathbf{A}}\ \mathbf{Z}^{(\mathbf{0})}\mathbf{W}^{(1)}\right). \quad (6)$$

Here, $\mathbf{W}^{(0)} \in \mathbb{R}^{d \times d_0}$ is an input-to-hidden weight matrix for a hidden layer with d_0 feature maps, and $\mathbf{W}^{(1)} \in \mathbb{R}^{d_0 \times d_0}$ is a hidden-to-output weight matrix.

D. Link Prediction Layer

After the iterative diffusion process propagated with a two-layer GCN, we obtain multiple representations for player node u—namely, $\{\mathbf{z}_u^{(0)}, \mathbf{z}_u^{(1)}\}$. The hidden representations obtained in different layers emphasize the messages passed over different connections or search depth in the graph, which makes them have different contributions in reflecting players' preferences. To this end, we concatenate them to constitute the final player representations. Likewise, we can obtain the final guild representations by concatenating the guild node representations $\{\mathbf{z}_g^{(0)}, \mathbf{z}_g^{(1)}\}$ learned by different layers. Thus, we have

$$\mathbf{z}_u = \mathbf{z}_u^{(0)}||\mathbf{z}_u^{(1)}, \ \ \mathbf{z}_g = \mathbf{z}_g^{(0)}||\mathbf{z}_g^{(1)} \quad (7)$$

where $||$ is the concatenation operation. With the final representations of players and guilds, we concatenate the latent vector representations of player u and guild g as $\mathbf{z}_u||\mathbf{z}_g$, and feed them into a two-layer fully connected network [multilayer perceptron (MLP)] to predict the preference $\hat{p}$ of player u to guild g, where the propagation rule of each layer is as follows:

$$\hat{p}_{ug} = \sigma\left(\mathbf{W_2}\ \mathrm{ReLU}\left(\mathbf{W_1}[\mathbf{z}_u||\mathbf{z}_g] + \mathbf{b_1}\right) + \mathbf{b_2}\right) \quad (8)$$

where $\mathbf{W}_1 \in \mathbb{R}^{d_1 \times 4d_0}$ and $\mathbf{W}_2 \in \mathbb{R}^{d_2 \times d_1}$ are corresponding weight matrices, $\mathbf{b}_1 \in \mathbb{R}^{d_1}$ and $\mathbf{b}_2 \in \mathbb{R}^{d_2}$ are corresponding bias, respectively. The σ is the *sigmoid* activation function, defined as $\sigma(x) = 1/(1+e^{-x})$.

E. Loss Function

To train our GuildNet model, we adopt the Bayesian personalized ranking (BPR) [27] loss function. As a pairwise learning framework, BPR is an very pervasive personalized ranking criterion used in recommender systems and information retrieval community. It is based on the triplets data $\{u, g^+, g^-\}$, and the semantics is that player u is assumed to prefer positive guild g^+ over negative guild g^-

$$L(u, g^+, g^-) = -\ln\sigma(\hat{p}_{ug^+} - \hat{p}_{ug^-}) + \lambda||\Theta||_2^2 \quad (9)$$

where σ denotes the *sigmoid* activation function, and Θ denotes model parameters. $L2$ regularization is applied to prevent overfitting and λ controls the regularization strength. For the triplet data $\{u, g^+, g^-\}$, we first sample a player u, then sample a positive guild g^+ from the guilds which u have interacted with, and a paired negative guild g^- from the rest of guilds.

V. Experiments

To demonstrate the effectiveness of the proposed framework, we have run extensive offline and online evaluations.

TABLE II
Statistics of Server Data

Server	# of players	# of guilds	# of P-G pairs[a]	# of P-P pairs[b]
1	26,895	431	27,097	488,025
2	73,639	1,010	75,099	285,049
3	84,545	1,263	85,773	328,736

[a] Number of player-guild interaction pairs.
[b] Number of player–player interaction pairs.

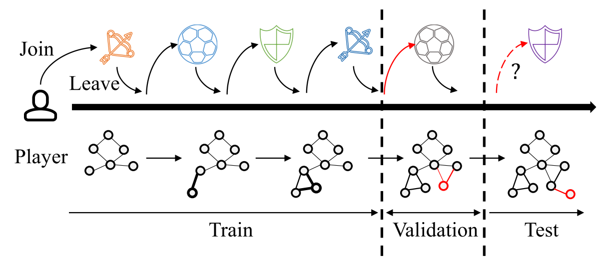


Fig. 9. Illustration of the dataset construction. The upper part indicates the history of interaction between the player and the guild in chronological order and the lower part indicates the history of friendship between the players in chronological order. The model use the features of player and guild and social relationships between them before joining a new guild as input data.

A. Offline Evaluation

1) Datasets: Considering that the game process of each game server is different, our recommendation model is server-dependent. We collected the data of three representative game servers for the whole year, from January 1, 2019 to December 31, 2019, to evaluate the effectiveness of our proposed model. The detailed data statistics of these servers are shown in Table II. Among them, the first server is a relatively old one, and the other two are newly opened and growing servers.

In the data we use for offline evaluation, the behavior history of each player can be represented as $(g_1, g_2, \ldots, g_k, \ldots, g_K)$, where g_k is the kth guild that the player interacts with in chronological order. We make the first k guild behaviors of each player as the context to predict the $k+1$th guild behavior in the training set and validation set, where $k = 1, 2, \ldots, K-2$, and we use the first $K-1$ guild behaviors to predict the last one in the test set. An illustration of dataset construction for the guild recommendation model is shown in Fig. 9. To make a reasonable offline evaluation, we would need to use the features and relationship graph for the target players before the time of joining the guilds.

2) Baselines: We compare the proposed model with several baselines in the experiments, including the following.

1) Logistic regression (LR): A linear classification model.
2) MLP: A multilayer feed-forward neural network with the nonlinear activation function (ReLU) for each neuron.
3) Decision tree (DT): The decision tree learning model.
4) Random forest (RF): An ensemble learning method which combines the *bagging* sampling and random selection of features to alleviate overfitting.
5) XGBoost (XGB) [28]: An optimized gradient boosting framework for decision trees, which is used widely by

TABLE III
COMPARISON OF METRICS IN OFFLINE EVALUATION

Method	Server 1			Server 2			Server 3		
	HR@5	MRR@5	NDCG@5	HR@5	MRR@5	NDCG@5	HR@5	MRR@5	NDCG@5
LR	<u>0.1571</u>	0.0959	<u>0.1344</u>	0.0733	0.0343	0.0438	0.0399	0.0212	0.0259
MLP	0.1473	<u>0.1013</u>	0.1068	0.0846	0.0424	0.0529	0.0332	0.0183	0.0220
DT	0.1492	0.0946	0.1080	0.0705	0.0322	0.0416	0.0352	0.0165	0.0211
RF	0.1111	0.0640	0.0756	0.0860	0.0455	0.0555	0.0366	0.0182	0.0227
XGB	0.1206	0.0624	0.0765	0.0790	0.0354	0.0460	0.0406	0.0219	0.0265
GCN	0.0808	0.0682	0.0671	<u>0.2453</u>	<u>0.1667</u>	<u>0.1856</u>	<u>0.1120</u>	<u>0.0466</u>	<u>0.0627</u>
GuildNet	**0.2290**	**0.1472**	**0.1676**	**0.3304**	**0.2231**	**0.2501**	**0.1294**	**0.0535**	**0.0689**

data scientists to achieve state-of-the-art results on many machine learning challenges in the industry.

6) GCN [21]: This model only considers the player-guild interaction graph and does not include the player–player friendship graph, i.e., the player–player interaction matrix in formula 1 is given as $\mathbf{A}_f = [0]_{M\times M}$. Since it is basically equivalent to using only guild membership in our framework, we use the results of this model to compare and verify the advantages of friendship relation for guild recommendation.

3) Evaluation Metrics: As we focus on recommending top-N guilds for each player, we evaluated the system by three widely adopted ranking metrics in information retrieval: hit ratio (HR), mean reciprocal rank (MRR), and normalized discounted cumulative gain (NDCG). Specifically, HR measures the number of guilds that the player is interested in the test data that has been successfully predicted in the top-N ranking list. MRR considers the rank position of the first relevant guild for the player in the ranking list. NDCG measures the usefulness (gain) of a guild based on its position in the ranking list, and additionally consider the discount factor at lower ranks. In our experimental settings, both MRR and NDCG will give a higher score if the useful guilds appear in the top positions. For all these metrics, the higher the values, the better the performance.

4) Implementation: In our experiments, we implement the GCN and our proposed GuildNet model by PyTorch [29] with the Adam optimizer [30], and other comparable models are implemented based on Scikit-learn,[5] a simple and efficient tools for predictive data analysis. We also employ early stopping and dropout techniques to prevent over-fitting. The free embedding size d is fixed to 32 for GCN and GuildNet. The hidden sizes d_0, d_1, d_2 are set to $64, 256, 128$, respectively, for both GCN and GuildNet. The batch size is fixed to 128. We apply grid search for tuning the hyperparameters of the model: the learning rate is tuned amongst $\{0.0001, 0.0005, 0.001, 0.005, 0.01\}$, the coefficient of $L2$ regularization is searched in $\{10^{-5}, 10^{-4}, \ldots, 1, 10^{1}\}$, and the dropout ratio in $\{0.0, 0.1, \ldots, 0.5\}$. The set of possible hyperparameter values was determined on early validation data which are not included in the training dataset.

5) Results and Analysis: We conducted comparative experiments on the data collected from the game servers, as shown in Table III (best method in bold, second best underline). The experimental results show that our proposed model outperforms other baseline methods. Meanwhile, from these experimental comparison results, we also have the following findings.

Fig. 10. GUI of guild recommendation system in the game.

1) On the whole, conventional machine learning methods, such as LR and XGB, perform worse than GCN-based methods, i.e., GCN and GuildNet. Especially, the comparison between the GCN and GuildNet model highlights the effectiveness of social relationships between players in improving the performance of guild recommendation.
2) The GCN-based models consistently outperform other baselines, especially on newly opened servers. We speculate that the propagation of relationships between players on newly opened servers is very important, since most guilds are in their infancy and the spread of friendship is beneficial to the growth of the guild.
3) On the mature servers (i.e., server 1), simple machine learning methods such as LR can also achieve good results. We believe this is due to the gradual stability of player preferences and guild attributes, whereas the influence of relationships is reduced.

B. Online Evaluation

The proposed recommendation model has been deployed in production for more than one year in the game *Justice Online*. For the performance comparison with the original method, we

[5][Online]. Available: https://scikit-learn.org/stable/

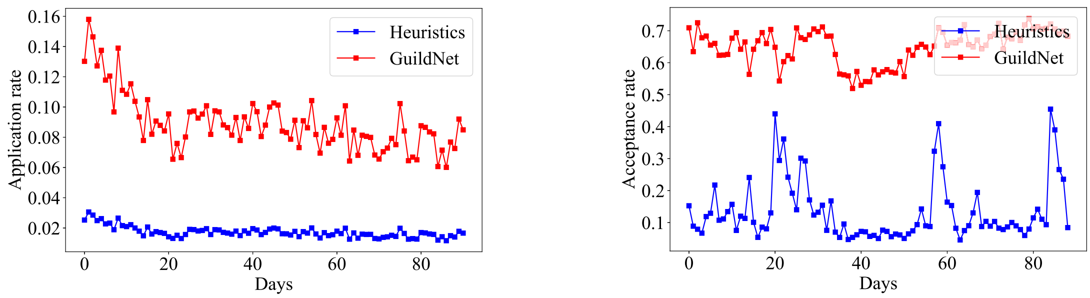


Fig. 11. Experimental results in online setting. The left part is the comparison of application rates and the right part is the comparison of acceptance rates.

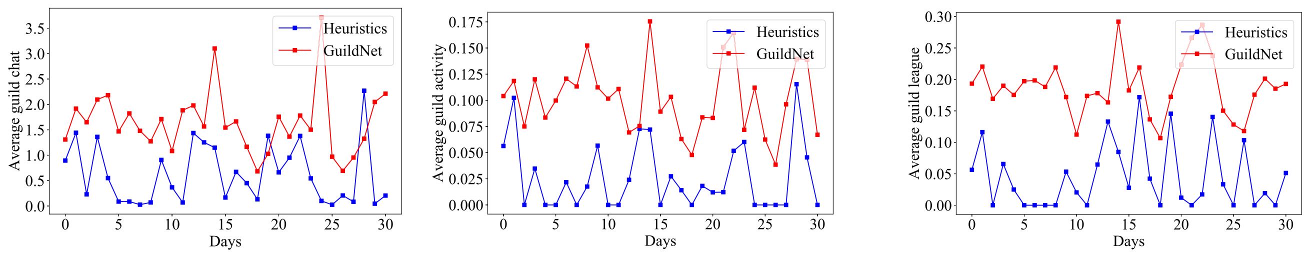


Fig. 12. Player engagement after joining the recommended guild. From left to right: comparison of average guild chat, average guild activity, and average guild league, respectively.

conducted online A/B testing. To this end, we randomly distribute recommendation requests to two channels: the original method and the proposed GuildNet method. The original recommendation method is designed with a heuristic rule based on the game level of players, because the level system is one of the core game mechanics in MMORPGs. To be specific, we first calculate the average level of all the members for each guild, and then rank the guilds according to the distance between the target player's level and the average ones of the guilds, i.e., the closer the average level is to the target player's level, the higher the ranking of guild recommendation.

We make some brief introductions to the graphical user interface (GUI) of the guild recommendation system, which is shown in Fig. 10. First, the player can open the interface to join the guild in the game (area 1), and then the recommendation engine (either the original heuristics or our proposed GuildNet model) will recommend several personalized guilds suitable for this player (areas 2 and 3 correspond to recommended and randomly arranged guilds, respectively). Once the player clicks on a guild, the tenet and some guild tags that reflect the characteristics of the guild will be displayed on the interface (areas 4 and 5), which can help the player to further understand the corresponding guild. Finally, the player can apply to join one or several guilds as he prefers (area 6), although he will only join the one guild when the guild administrator agrees to the application. It is worth noting that the recommended guild is indicated by a mark in the GUI (red box area).

After the system was launched, we have collected the online feedback data of a server from January 1, 2020 to March 31, 2020. We compare the recommendation performance of our model with the heuristic-based method. As shown in Fig. 11(a) and (b), the launch of the model yielded significant gains on application rate, which means players have applied for our recommended guilds, and acceptance rate, which means players have applied for our recommended guilds and the guild administrator also accept the application, making the player successfully become one of the guild members. Further, since just the pure fact that a guild has let somebody in may not be a proof that the new member suits the guild well. In order to verify the effectiveness of the guild recommendation system, we have collected the engagement measures of players after entering the guild for a period of time. Based on the knowledge of game experts and our observations on the game, we select three of the most representative indicators to measure player engagement after joining the guild.

1) *Guild chat:* This measure indicates the player's willingness to communicate with other guild members, which may involve giving help to other members, or seeking help from other members, etc.
2) *Guild activity:* This measure indicates the player's willingness to participate in the daily activities of the guild, involving members' daily activeness within the guild.
3) *Guild league:* This measure indicates the player's willingness to collaborate within the guild when competing with other guilds, involving communication and collaboration with other members.

We show these engagement measures of a server for our proposed model and the heuristics from March 1, 2020 to March 31, 2020. The comparison results of degree of engagement is shown in Fig. 12, from which we can see that although there are some scattered exceptions, overall, our recommendation model outperforms the heuristic rules in all involved metrics. It seems that our recommendation model has more obvious advantages in the guild activity and guild league metrics. These results verify the feasibility and effectiveness of our method in practical applications, and also our proper use of the friend relationship between players can indeed improve the performance of the recommendation system.

VI. Conclusion

In this article, we focus on the problem of guild recommendation in the context of online games. We present a framework to combine the strengths of deep learning technology and expert knowledge in the game field, which makes full use of player behavior preference and the social network of the virtual world. The central component is a graph convolutional network that is trained to estimate the likelihood, for each player, of whether the player matches the guild. We successfully integrate the entire system into the production environment of *Justice Online*, one of the most popular commercial MMORPG developed by Netease Games. Offline and Online experiment results show that the proposed framework led to significant improvement in conversion rate over baselines. The results also demonstrate that a purely data-driven approach can enrich the game design on the original mechanism and bring a better game experience to the player.

Our framework, while still can be improved in some aspects (such as incorporating more distinguishing features and deploying a real-time system with faster feedback), gives insights into how deep learning techniques can be used as an effective tool for tackling decision problems in online games, especially scenarios that require a lot of expert experiences. Future work will try and show its applicability to other similar scenarios, such as items recommendation in MMORPGs. We are also very interested in analyzing the evolution of guilds from the perspective of group dynamics using deep learning methods. We believe that such research topics in online games could also bring some inspirations to the research of social groups in the real world.

References

[1] T. L. Taylor, *Play Between Worlds: Exploring Online Game Culture*. Cambridge, MA, USA: MIT Press, 2009.

[2] N. Ducheneaut, N. Yee, E. Nickell, and R. J. Moore, "'Alone together?': Exploring the social dynamics of massively multiplayer online games," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2006, pp. 407–416.

[3] D. Williams, N. Ducheneaut, L. Xiong, Y. Zhang, N. Yee, and E. Nickell, "From tree house to barracks: The social life of guilds in World of Warcraft," *Games Culture*, vol. 1, no. 4, pp. 338–361, 2006.

[4] S. P. Borgatti, A. Mehra, D. J. Brass, and G. Labianca, "Network analysis in the social sciences," *Science*, vol. 323, no. 5916, pp. 892–895, 2009.

[5] D. Centola, "The spread of behavior in an online social network experiment," *Science*, vol. 329, no. 5996, pp. 1194–1197, 2010.

[6] M. L. Barnes, J. Lynham, K. Kalberg, and P. Leung, "Social networks and environmental outcomes," *Proc. Nat. Acad. Sci.*, vol. 113, no. 23, pp. 6466–6471, 2016.

[7] Y. Pisan, "My guild, my people: Role of guilds in massively multiplayer online games," in *Proc. 4th Australas. Conf. Interactive Entertainment*, 2007, pp. 1–5.

[8] M. Szell, R. Lambiotte, and S. Thurner, "Multirelational organization of large-scale social networks in an online world," *Proc. Nat. Acad. Sci.*, vol. 107, no. 31, pp. 13636–13641, 2010.

[9] A. F. Seay, W. J. Jerome, K. S. Lee, and R. E. Kraut, "Project massive: A study of online gaming communities," in *Proc. Extended Abstr. Hum. Factors Comput. Syst.*, 2004, pp. 1421–1424.

[10] A. R. Kang, J. Park, J. Lee, and H. K. Kim, "Rise and fall of online game groups: Common findings on two different games," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 1079–1084.

[11] N. Ducheneaut, N. Yee, E. Nickell, and R. J. Moore, "The life and death of online gaming communities: A look at guilds in World of Warcraft," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, 2007, pp. 839–848.

[12] C. S. Ang and P. Zaphiris, "Social roles of players in MMORPG guilds: A social network analytic perspective," *Inf. Commun. Soc.*, vol. 13, no. 4, pp. 592–614, 2010.

[13] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 191–198.

[14] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[15] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.

[16] S. Rendle, "Factorization machines," in *Proc. IEEE Int. Conf. Data Mining*, 2010, pp. 995–1000.

[17] H.-T. Cheng *et al.*, "Wide & deep learning for recommender systems," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 7–10.

[18] J. Tang, X. Hu, and H. Liu, "Social recommendation: A review," *Social Netw. Anal. Mining*, vol. 3, no. 4, pp. 1113–1133, 2013.

[19] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 931–940.

[20] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 287–296.

[21] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–14.

[22] L. Wu, P. Sun, Y. Fu, R. Hong, X. Wang, and M. Wang, "A neural influence diffusion model for social recommendation," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2019, pp. 235–244.

[23] W. Fan *et al.*, "Graph neural networks for social recommendation," in *Proc. World Wide Web Conf.*, 2019, pp. 417–426.

[24] A. F. del Róo, P. P. Chen, and Á. Periánez, "Profiling players with engagement predictions," in *Proc. IEEE Conf. Games*, 2019, pp. 1–4.

[25] J. Klicpera, A. Bojchevski, and S. Günnemann, "Predict then propagate: Graph neural networks meet personalized pagerank," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–15.

[26] G. Li, M. Müller, A. Thabet, and B. Ghanem, "DeepGCNs: Can GCNs go as deep as CNNs," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9266–9275.

[27] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Uncertainty Artif. Intell.*, 2009, pp. 452–461.

[28] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining*, 2016, pp. 785–794.

[29] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–15.