This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

On New Group Popularity Prediction in Event-Based Social Networks

Guangyu Li, Yong Liu, Bruno Ribeiro, and Hao Ding

Abstract—Event-based social networks (EBSN) have recently emerged as an important complement to online social networks. They enjoy the advantages of both online social networks and offline social communities: offline social events can be conveniently organized online, and users interact with each other face-to-face in the organized offline events. Although previous work has shown that member and structural features are important to the future popularity of groups in EBSN, it is not yet clear how different member roles and the interplay between them contribute to group popularity. In this paper, we study a real-world dataset from Meetup — a popular EBSN platform — and propose a deep neural network based method to predict the popularity of new Meetup groups. Our method uses group-level features specific to event-based social networks, such as time and location of events in a group, as well as the structural features internal to a group, such as the inferred member roles in a group and social substructures among members. Empirically, our approach reduces the nRMSE of the popularity prediction (measured in RSVPs) of a group's future events by up to 12%, against the state-of-the-art baselines. Through case studies, our method also identifies member and structure patterns that are most predictive of a group's future popularity. Our study provides new understanding about what makes a group successful in EBSN.

Index Terms—Event-based Social Networks, Group Popularity Prediction, Circular Fingerprints, Role Discovery.

1 Introduction

As online social networks become more prevalent, people's face-to-face interactions are reshaped by these networks. In this work we focus on *event-based social networks* (EBSN), online social networks whose members hold in-person events. Meetup [1] is a popular EBSN that allows its members to find and join online interest groups, and organize face-to-face events in different categories, such as politics, books, games, movies, health, pets, careers, and hobbies, etc. While it is relatively easy to establish new groups in EBSN, it takes much more effort from the group organizers and members to make a group popular and sustainable. It is therefore important to understand the key factors contributing to the popularity and sustainability of groups in EBSN, especially newly established groups. Insights obtained from such a study can be used to guide the promotion, recommendation and investment on EBSN groups by EBSN platforms and investors.

In this paper, we study the problem of group popularity prediction in EBSN, with a special focus on new groups. More specifically, we focus on predicting the popularity (measured in the number of RSVPs) of newly established interest groups in Meetup. The main questions that we want to answer are: *1). can we predict the future success of new groups? 2). what are the observable factors that best predict a group's success?* Different from the previous studies on group popularity prediction in the traditional online social networks, our study takes into account the unique features of EBSN in the following key aspects:

- *Get-out-of-the-couch Effort:* To participate in an offline social event, a user must be physically present at specific venue at a scheduled time. Clearly, this takes more effort and commitment than participating in a pure online event. As a result, a user's attention (the ability to be active in multiple groups) is a severely limited resource, which must be accounted for in predicting the successes of competing groups.
- *Face-to-face Social Interactions with Implicit Social Relationships:* Once users meet in person, they may form stronger bounds than online interactions. Thus, to predict a group's success one must account for stronger and more sustainable social ties established in EBSN than in online communities. However, EBSNs normally don't have the explicit social relations between group members, which are readily available in online social networks, such as *"friends-with"* (Facebook) and *"follower"* (Twitter).

1.1 Contributions

We develop a novel approach to predicting the popularity of newly formed groups in EBSN, achieving the state-of-the-art accuracy. Our approach considers various factors: i) the group-level features, such as the past popularity of the group and the number of events; ii) the event-based features, such as location and schedule of events; iii) user-level features related to user's attention: how active is a user in individual groups, and how does the user distribute her activity/attention among multiple groups. Based on these features, the first key idea of our approach is the use of role discovery to determine the importance of users in a group and the roles they play. Similar to any social community in real life, a group in EBSN is more than a simple collection of members. A group's characteristics are mostly determined by the interaction among group members, and its success ultimately hinges on the "*chemistry*" among its members. Coincidentally, the second key aspect of our approach is the extension of circular and neural fingerprints techniques developed in chemistry [2], [3] to study how social ties between different types of users contribute to group success. Specifically, armed with each group member's role, and event co-participation graphs generated from those members'

- *Guangyu Li, (gl1257@nyu.edu), Yong Liu (yongliu@nyu.edu) and Hao Ding (hao.ding@nyu.edu) are at NYU Electrical and Computer Engineering Department. Bruno Ribeiro (ribeiro@cs.purdue.edu) is at the Purdue Computer Science Department.*

Manuscript received xxxxx; revised August xxxxx.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

activities, we combine the members' roles with these activity networks to predict a group's success, extending the fingerprints techniques developed to correlate the characteristics of atoms along with the neighboring bonds to other atoms to determine a molecule's function. Our extension also accounts for a user's limited attention by incorporating "attention-based" features, such as how many groups a user joined and how much time the user spent in each group into the member-level feature set.

By applying our novel approach to the Meetup dataset, we obtain interesting findings w.r.t. features that predict a group's success: (1) The user roles we discover in a group's social network are better predictors of the group's popularity than any other member-level features. (2) The most relevant user roles contributing to a group's popularity are not "organizer-like members", but "ordinary members" who have similar activity levels with their friends. (3) The most important substructures (interaction patterns) are not combining all the most important roles, but follow different combination patterns for different types of groups.

1.2 Outline

The rest of the paper is organized as follows. Section 2 introduces the related work on group behavior prediction in social networks, role discovery techniques and structural feature extraction. Section 3 describes the method we propose to solve the problem of predicting the group future popularity. In Section 4, we evaluate the performance of the proposed method using the Meetup dataset. Finally we conclude our work in Section 5.

2 Related Work

2.1 Event-based Social Networks

Event-based social networks (EBSNs) were first studied by Liu et al. [4]. They are differentiated from other social networks such Facebook and Twitter because of their unique capability of linking the online and offline social worlds. The event-based social networks have drawn a lot of attention from academia in recent year since they have more similar characteristics with the social communities in our daily life. Researchers have studied this type of social networks from different angles. With millions of users and thousands of events held every week, the EBSNs have the problem of recommending events to interested users. [5] exploits social signals based on group memberships, location signals based on the users' geographical preferences, and temporal signals derived from the users' time preferences, then they combine the proposed signals for learning to rank events for personalized recommendation. In [6] a recommender named "LCARS" is proposed to automatically learn the users' behavior offline and combine the learnt interest of the querying user and the local preference of the querying city to produce the top-k recommendations online. Group recommendation problem is studied in [7] where the location features, social features, and implicit patterns are simultaneously considered in a unified model. [8] develops a sliding-window based machine-learning model that effectively combines user features from multiple channels to recommend users to new events.

Instead of focusing on individual users' behaviors, some studies focused on the group behavior within a group's lifespan. A group in event social network is composed of all the users that subscribe to it and periodically attend the events held by this group. In some related works, group-level features have been investigated to explain a group's success or failure. In [9] Meetup group success metrics are defined and used to generate labels for machine learing models. Liu et al. [10] have investigated group-level features especially the structural features of the social graphs to make prediction of the future failure of a group. Our work also focuses on predicting group popularity. We not only use a very different (and more effective) methodology, but also expand the feature set to include unique group-level features in EBSNs, such as event venue and schedule, as well as the combination of member roles and the social network structure.

2.2 Group Popularity Prediction

There are two major lines of research for this problem. One focuses on characterizing the evolution of online social network popularity by applying mean-field epidemic models to the time series of the "Daily Active Users" (DAU), without user-level or network structural information. [11] proposes a combination of reaction-diffusion-decay processes whose resulting equations seem not only to describe well the observed DAU time series but also provide means to roughly predict their evolution. In [12] both diffusion and non-diffusion mechanisms in the group joining process, and power-law decay in group quitting process are observed and then a refined group evolution prediction model is proposed according to this observation. The other focuses on using general group features to make predictions [9], [10], [13]. In contrast, our approach uses richer information and convolutes member roles, member's attention-capacity features, with their activity network structure to achieve higher prediction accuracy.

In contrast of [9] where four metrics including group size, average attendance, attendance growth rate and group size growth rate have been proposed to "vote" out a successful group, we choose the RSVP number as the the only criterion of group success. The reasons are three-fold: 1) group size may not be an effective measure since we observe that in Meetup, there are many inactive users who are not likely to attend any events in future, while group sizes take all those users into consideration; 2) We compare the number of RSVPs over a fixed time interval, so the growth rates over these intervals can be automatically considered; 3) to avoid the dominance of extremely large groups, we (a) choose the "Normalized RMSE" as our performance metric, and (b) measure it only over "new groups" that begin at the same time point

Our basic approach and preliminary results were published in a poster paper [14]. In this work we will perform deeper analysis of the relation between member role&structure and group popularity, analyzing interesting user behavior patterns such as seasonal patterns of groups with certain topics, and conduct comprehensive comparison of our approach with more advanced machine learning algorithms, such as XGBoost and LightGBM.

2.3 Role Discovery

The goal of network role discovery is to classify network nodes according to the roles they play in the network. Three types of roles: (i) graph-based roles, (ii) feature-based roles, and (iii) hybrid roles are defined in [15] [16] and a framework has been proposed to detect them. In our work, we consider the "hybrid roles" of the users that represent both network and feature characteristics of each node in event-based social network as features to predict future success of a group. For detecting roles in a network, [17] proposes a framework that detects a node's role

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

and certain community simultaneously. [18] proposes a method that efficiently detects communities in a network based on nodes' influences. Given a graph along with node features, the process of role discovery relies on defining node equivalence. Various types of equivalences have been introduced in previous studies, such as graph-based equivalences, feature-based equivalences, and hybrid equivalences [15]. To capture both structural and feature equivalences between members, the information that we use for role discovery represents both members' intrinsic behaviors, such as the numbers of events/groups that they have participated in, as well as the partial structural behaviors of members by including their one-hop-neighbors' features (see Table 1 for more details).

2.4 Structural Feature Extraction

Besides using each user's social roles as the features to predict group success in future, we also exploit structural features to make the prediction. Several studies on group popularity prediction have found that a group's main characteristics can be largely related to the interaction patterns among their members [9], [10], [13]. These patterns can be represented by the node attributes and link structures of a graph generated based on members' interactions. While multiple methods, such as "deepwalk" [19] and "node2vec" [20] have been proposed for mapping structured data to real-valued feature vectors, these mainly focus on transforming graphs into node features, instead of obtaining features for the graphs. Some recent studies [21], [22] also focus on transforming substructures or subgraphs in large graphs to feature vectors, however the resulting latent vectors cannot be easily interpreted to obtain insights about group popularity. In contrast, we extend the circular and neural fingerprints techniques in chemistry informatics to gain important understanding on how subgraphs among different types of members contribute to group success.

3 Proposed Method

In this section, we first define a metric of a group's popularity and describe our prediction problem in the context of the Meetup EBSN. We then propose our prediction method, leveraging on the group-level features and member-level features. Finally, we present our overall method combining these group-level and member-level features to make predictions.

3.1 Meetup Group Popularity Prediction

Meetup is an event-based social network in which users can form and join different interest groups online, and organize and participate in face-to-face social events offline. The group organizers create events, and each event has specified time, location and topic. The information about new events will be sent to group members through emails or website notifications. Each group member decides whether she will participate in the new events based on her time, location, and topic preferences, and then respond by sending RSVPs ("yes", "no", or "maybe"). Figure 1 illustrates the main components in Meetup social network. The left part of Figure 1 shows 2 different groups are represented as big dashed circles. Group organizers, events and ordinary users are marked as rectangles, triangles and circles respectively. Whenever two users co-attend the same event we assume there is a virtual link between them. In the right part of the figure the resulting social graph from left part of the figure is shown. Note that in this graph the nodes of events are eliminated and there are users who attend both groups (node 5). With the definitions of groups, events and users in Meetup, the group popularity prediction problem can be defined as:

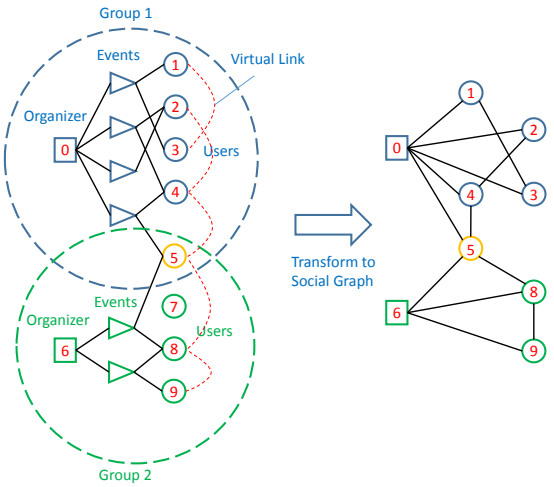

Fig. 1. Components in Meetup Social Network. The left-hand side shows components of two groups: every solid line from a user to an event represents a "yes" RSVP; When two users participate in the same events, a "virtual link" between them is created to represent their potential interaction; User 5 is the "social spanner" who joins two groups and user 7 is an inactive user who hasn't participated in any event; The right-hand side: the resulting event co-participation social graph among group members.

Definition 1. Given the activities of a group within a time window of $[0, n]$ months, and a time interval of m months, predict the group's total RSVP number (popularity) within a future time window of $[n + m, 2n + m]$ months.

The time interval of m months can be chosen to eliminate the effect of seasonal event holding patterns. For example, a skiing group may have high activity level only in winter (October $\sim$ December) while the RSVP numbers in summer could be small. Using winter RSVP numbers to predict summer RSVP numbers, or vice versa, is not a meaningful prediction task. In our study, we calculate prediction accuracy for multiple choices of m, then take the average to represent the overall accuracy.

3.2 Group-level Features

The most straight-forward features to use for popularity prediction are the summary statistics of each group. So we start with extracting "group-level features", which are various summary statistics of a group without examining the detailed features of each member in the group. We list the descriptions of fourteen group-level features for each Meetup group in Table 1, such as the scheduled time distributions of its events, the location distributions over its venues, and RSVP counts of all members, etc. However only using group-level features may not give the best prediction performance since they cannot reflect subscribers' behavior patterns thus overlook the contribution of certain users and their social roles to the group's success. In the next section we will introduce the internal group features that take the users' social role and interaction pattern into account.

3.3 Internal Group Features

Internal features of a group can be defined as all the features that are related to each individual member in the group. These

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

features should include the first-order features that can be directly calculated using basic statistics, such as the past attendances of a member and how many groups a member has joined. They should also include the second-order features that require further processing, such as the member role discovery and the structural feature extraction.

3.3.1 Member-level Feature Extraction

We start with constructing social graph for each group from which the features are extracted. For each given group g, the event co-participation social graph about its members is defined as:

Definition 2. $G^g = (U^g, E^g, W^g)$, where U^g is the member set in group g, E^g denotes all edges between members and W^g represents all edge weights. Two members u_i and u_j are connected if they co-participated in at least one event in group g. The weight on each edge is calculated as the number of events the two members have co-participated in.

In Figure 1, the event co-participation social graph between Meetup users in the left subgraph is illustrated in the right subgraph. Based on the event co-participation social graph, we propose twelve member-level features listed in Table 1: Feature m1∼m6 represent "*who you are*", i.e., the features related to the member's own characteristics, and feature m7∼m12 represent "*who you know*", i.e., the features related to the characteristics of her neighbors in the group's social graph. Note that we deliberately design and incorporate features related to a member's cross-group activities — "m3: number of groups the member has joined", "m4: entropy of the numbers of events the member participated in all groups", "m5: entropy of the total numbers of events in all groups, and "m6: entropy of the fractions of events the member has participated in all groups" — into the prediction so that the cross-group activities are taken care by our algorithm.

3.3.2 Member Role Discovery

To find role features of each group member, we use Non-negative Matrix Factorization (NMF) [23]. For a given a member-feature matrix $\mathbf{X} \in \mathbb{R}^{n \times f}$, we generate a rank-r approximation ($r < min(n, f)$) $\mathbf{MF} \approx \mathbf{X}$, where each row of $\mathbf{M} \in \mathbb{R}^{n \times r}$ represents a node's membership in each role and each row of $\mathbf{F} \in \mathbb{R}^{r \times f}$ represents how membership of a specific role contributes to the estimated feature values. With a distance measure $\|\cdot\|$, the problem can be simplified as:

$$\min_{\mathbf{F} \in \mathbb{R}^{r \times f}, \mathbf{M} \in \mathbb{R}^{n \times r}} \|\mathbf{MF\text{-}X}\|$$

subject to $\mathbf{M}, \mathbf{F} \geq 0$. In practice, considering the feature number $f = 12$, for the purpose of dimension reduction, we choose the role number to be $r = 6$.

Taking the member-role sub-matrix $\mathbf{M^g}$ for group g generated by the role discovery method, we sum over all the members (rows) and get the group's role distribution vector $\mathbf{V^g} = \left\{\mathbf{V^g_j} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{M^g_{ij}}, \mathbf{1} \leq \mathbf{j} \leq \mathbf{r}\right\}$, then we stack all the vectors $\{\mathbf{V^g}\}$ to form a group-role matrix $\mathbf{\Omega} \in \mathbb{R}^{\mathbf{p} \times \mathbf{r}}$ where p represents the number of groups and r is the number of roles. Thus the correlation between role X and group popularity can be calculated as the correlation between the column of role X in $\mathbf{\Omega}$ and all groups' RSVP numbers.

In Figure 2 (top) we calculate the Pearson correlation between the member feature vectors and their group's popularity. The weak correlations (in the range of $[-0.2, +0.2]$) indicate that no individual member feature plays an important role in the popularity of group.

Figure 2 (bottom) shows that role discovery (i.e. combining multiple features to form a role) significantly increases the correlation, where roles are now positively and strongly correlated with group popularity (in the range of $[+0.45, +0.75]$). Each role is a linear combination of multiple raw member features. Thus by analyzing the roles, we can get a better understanding of how certain combination of features contributes to the group's popularity, and creates a role profile for each member.

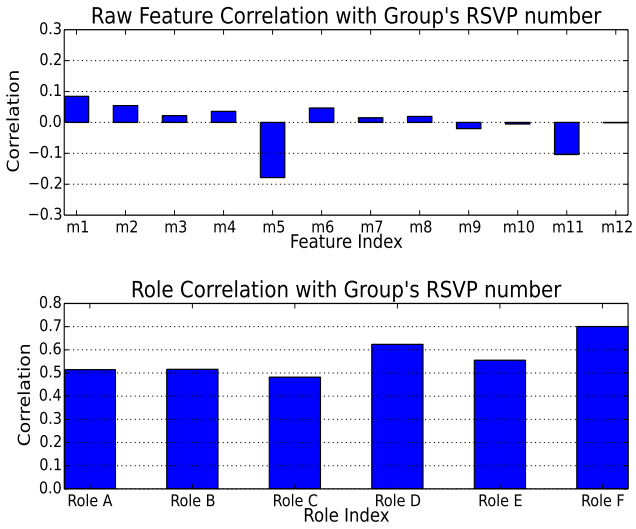

Fig. 2. Feature Correlation with Popularity. Role discovery method combines multiple raw member features into one distribution to represent a member's profile.

3.3.3 Structural Features

As observed in the previous studies, structural features of a group play important roles in affecting the group's characteristics. However, the existing work does not consider the roles of members within the group when detecting structural patterns, while intuitively the members' roles should be taken into account. Let's consider an example group, "NY Tech", the largest group in the Meetup social network. Every week the organizers of "NY Tech" create a technical conference-like event and sometimes invite an expert on some topic to be the speaker (Figure 3). In this case, without differentiating between the two roles, "organizer" and "speaker", there is no difference between the two social network graphs. However, with the knowledge of their roles, one may easily notice that the subgraph surrounding the organizers (in the upper right) is more "stable"[1] than the one surrounding the speaker (in the lower right) since the "speaker" is very likely to be changed in the next event, while the group's organizer remains the same.

3.3.4 Circular Fingerprints

In order to extract structural features embedded with member roles, we use the "circular fingerprints" algorithm [2]. Circular fingerprints is a popular tool for handling graph-structured data in chemistry. It was first designed for molecular characterization,

1. The stability we mentioned here merely aims to give readers an intuition of the difference between two similar structures containing different roles, thus not formally defined.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

Group-level features	g1. Entropy of the time distribution over all times the events are held g2. Average distance between any two events the group held g3. Variance of the "event-event" distances g4. Average distance between any event and any participating member g5. Variance of the "event-member" distances g6. Average distance between any member and any other member in the same group g7. Variance of the "member-member" distances g8. Entropy of the location distribution over all venues the event are held g9. Density of the group's social graph g10. Total degree of the group's social graph g11. Event number the group has held g12. Average RSVP number of all past events g13. Variance of RSVP numbers of past events g14. Sum RSVP number of past events.
Member-level features	m1. Total degree of the member m2. Number of events the member has participated in current group m3. Number of groups the member has joined m4. Entropy of the numbers of events the member has participated in the groups the member has joined. m5. Entropy of the total numbers of events of the groups the member has joined m6. Entropy of the fractions of events the member has participated in the groups the member has joined m7. Average degree of the member's 1-hop neighbors m8. Average event number the "1-hop neighbors" have participated m9. Average group number the "1-hop neighbors" have joined m10. Average entropy of "1-hop neighbors'" attendance distribution over the groups they have joined m11. Average entropy of "1-hop neighbors'" event number distribution over the groups they joined m12. Average entropy of "1-hop neighbors'" event fraction distribution over the groups they joined

TABLE 1
Group-level and Member-level Features

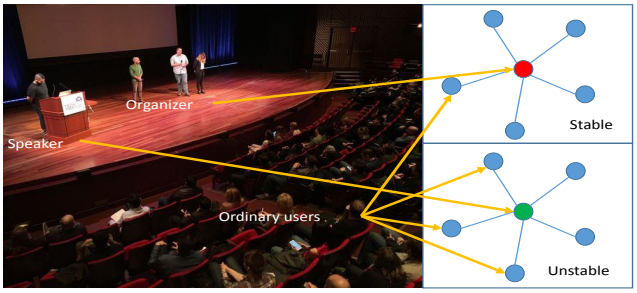

Fig. 3. A subgraph in the "NY Tech Meetup" Group

similarity searching, and structure-activity modeling. A molecule consists of atoms with different types. How different types of atoms are bounded together is the key factor that determines the characteristics of the molecule. In the context of event-based social network, we draw the analogy between a molecule and a group. We assume the members of a group are analogous to the atoms of a molecule, and the social ties between members are analogous to the chemical bonds (Figure 4). Then we can study how the subgraphs between members contribute to the group popularity using the circular fingerprints framework. The fingerprint generation process consists of two main steps:

1) The algorithm starts with assigning an initial identifier to each atom (member) in the molecule (group). This identifier captures some basic information of the atom (member) such as atomic number (member's feature), connection count, etc. In our case the "basic information" is the role distribution (Figure 2) attached to each member.
2) Then, a number of iterations are performed to combine the initial atom (member) identifiers with identifiers of neighboring atoms (members) until a specified radius (number of hops from this atom) is reached. For example, in iteration 1, the identifiers of all "one-hop neighbors" of the target atom are combined with the identifier of the target atom to generate a new identifier. Each iteration captures larger and larger circular neighborhoods around each atom (member), which are then encoded into single integer values using a suitable hashing method, and these identifiers are collected into a list. In this way, each subgraph is generated by a member along with her neighbors within a certain radius.

The identifier list (also called "fingerprints") is then used to characterize the properties of the molecule. In our case, we use the fingerprints as structural features to predict the group's popularity.

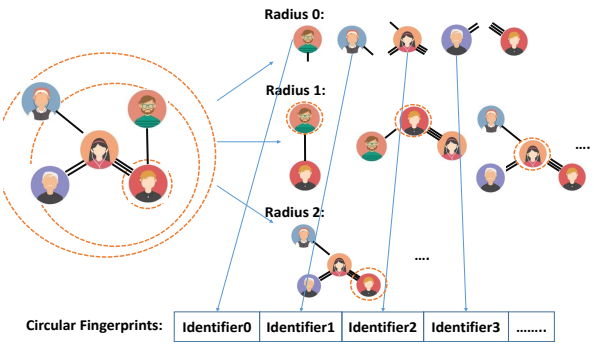

Fig. 4. Subgraphs Detected by Circular Fingerprints in Social Network. Circular Fingerprint scans the network for all subgraphs under certain radius. Each subgraph is then encoded into an integer identifier. Integer identifiers of all subgraphs constitute the circular fingerprints.

3.3.5 *Group-Role Neural Fingerprints*

Although circular fingerprints is a convenient tool to study social graph, it has several limitations: 1) the algorithm can only handle graphs with fixed sizes; 2) even if the graphs vary a little bit, the resulting fingerprints can be quite different, making the features highly vulnerable to noise. Overcoming these limitations, Duvenaud et al. [3] proposed a convolutional neural network, where each neural network layer simulates the updating operation in circular fingerprints (Figure 5).

We now extend the convolutional fingerprint algorithm of Duvenaud et al. to solve our problem. We denote this approach as the *Group-Role Neural Fingerprints*. For radius 0, the first hidden layer in the network takes the initial member-role matrix which is produced by the previous role discovery step as the input, then the output of this layer goes in two directions: in one direction the output is directly calculated as the radius 0 fingerprint; in the other direction, the output is updated with the adjacency matrix through a "feature update" operation. In this update operation,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

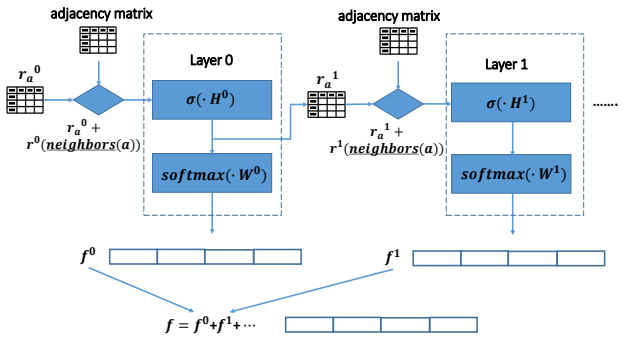

Fig. 5. Group-Role Neural Fingerprints Algorithm. "Feature Update" operation: each member's feature is updated so that the 1-hop neighbors' features and neighboring edge features are added to the initial member feature.

the member-role matrix is updated so that each member's 1-hop neighbors' role distribution vectors are added to the corresponding row of member-role matrix. In this way, the algorithm iterates until a certain radius is reached. After each iteration, more and more local structural information is captured. The detailed operations are presented in Algorithm 1. As the result, the final fingerprints are calculated as the summation of the fingerprints at each radius and then taken as input of Multilayer Perceptron to produce prediction of the RSVP number (popularity). The number of hidden layers equals to the given radius R. The neural network's weights $\mathbf{H^0}, ...\mathbf{H^R}$ and $\mathbf{W^0}, ...\mathbf{W^R}$ are learned from the training process. The σ and *softmax* functions are given as:

$$\sigma(\mathbf{x})_j = \frac{1}{1+e^{-\mathbf{x}_j}}, \quad softmax(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \text{ for j=1...K.}$$

Since it is a convolutional network-like structure, the number of hidden units at each hidden layer and the length of fingerprints f are both m. For experiments, we choose m to be 10 and the size of input layer to be 6 (equals to the length of role distribution vector) as these parameters give best performance in validation phase.

3.3.6 Computational Cost Analysis

The computation cost for the Group-role neural fingerprint of depth R, fingerprint length L of a social graph with N nodes that have F related role features is $\mathcal{O}(RNFL + RNF^2)$. In practice, training a GRNF with fingerprint length of 10 and convolution layer sizes of $(10 \times 10 \times 10)$ takes on the order of one minute for each training batch which contains 100 social graphs on a computer with 2.67Hz single-thread CPU and 96G RAM.

3.4 Combining Group-level and Internal Features

We now have two types of features for group popularity prediction, namely, the group-level features and our convolution of the member-level features with the group's social graph (Group-Role Neural Fingerprints). We proceed to explore different ways to combine the group-level features and internal features to make better prediction than only using one channel of them. We investigate three combination methods:

- *Method 1: Combined Prediction.* Predict group popularity using group-level and member-level features independently, then combine the predictions as $\alpha\times$(group-level prediction)+(1-α)$\times$(member-level prediction), where $\alpha \in [0, 1]$. The value of α is optimized using the validation data.
- *Method 2: Clustered Combination.* We first cluster the groups based on the group-level features, and then find the optimal α weight for each cluster. Whenever a newly formed group arrives, we calculate the inverse of the distances from the new group to the existing group clusters to find the optimal weight for the new group. The optimal weight is calculated as:

$$\alpha^* = \frac{\sum_{i=1}^{N} \alpha_i / d_i}{\sum_{i=1}^{N} 1/d_i}. \quad (1)$$

where d_i is the average distance between the new group and all groups in cluster i.
- *Method 3: Deep Neural Network (DNN) based Combination.* As illustrated in Figure 6, a deep neural network is used to combine group-level and member-level features. It is a combination of our neural fingerprints network with two Multilayer Perceptrons (MLPs). As will be shown in our evaluation, it is the best combination method.

Algorithm 1: Forward Pass

Input : group's social graph, members-role matrix, radius R, hidden layer weights $\mathbf{H^0}, ...\mathbf{H^R}$, output layer weights $\mathbf{W^0}, ...\mathbf{W^R}$, length of role distribution vector t, fingerprint length m.

Initialize: fingerprint vector $\boldsymbol{f} \Leftarrow 0_s$

for *each member a in social graph* **do**
  $\boldsymbol{r_a}^0 \Leftarrow$ a's role distribution vector
  $\mathcal{N}(a) =$ the set of neighbors of a
end
for $L = 0$ *to* R **do**
  $\boldsymbol{f}^L \Leftarrow 0_s$
  for $a \in \{$*members in social graph*$\}$ **do**
    $\boldsymbol{v_1} \Leftarrow \boldsymbol{r_a}^L + \sum_{i \in \mathcal{N}(a)} \boldsymbol{r_i}^L$
    $\boldsymbol{v_2} \Leftarrow \sigma(\boldsymbol{v_1}^T \mathbf{H}^L)$
    $\boldsymbol{f}^L \Leftarrow \boldsymbol{f}^L + softmax(\boldsymbol{v_2}^T \mathbf{W}^L)$
    $\boldsymbol{r_a}^{L+1} \Leftarrow \boldsymbol{v_2}$
  end
  $\boldsymbol{f} \Leftarrow \boldsymbol{f} + \boldsymbol{f}^L$
end
Return : real-valued vector $\boldsymbol{f}$

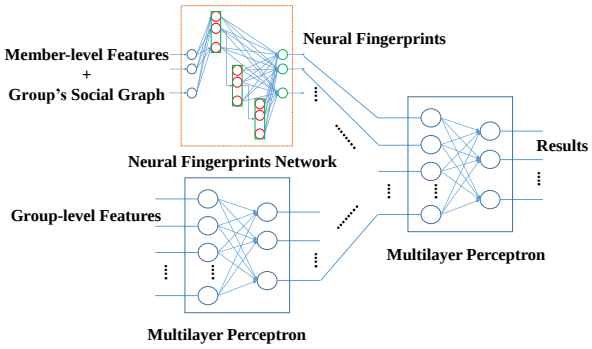

Fig. 6. Combining Group-level Features and Internal Group Features using DNN

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

4 Performance Evaluation

To evaluate the performance of our proposed method, we conduct experiments on the Meetup dataset. We first provide a brief description about our dataset. Then we test the prediction power of group-level features, member-level features, and group-role neural fingerprints respectively. Next, we compare the accuracy of the three methods of combining group-level features and member-level features with three competitive baselines. Finally, we analyze the importance of various types of member roles along with the interaction patterns (social structures) between the roles for predicting group success.

4.1 Dataset Description

Using the Meetup's dataset API from its website, we crawled all Meetup groups located within 50 miles of New York City (NYC), from March 2003 to February 2015, including all the related meta-data. Table 2 summarizes the salient statistics of the collected dataset.

Name	Value
Number of groups	17,234
Number of users	1,101,336
Number of events	1,025,719
Number of RSVPs	8,338,382
Number of venues	93,643
Avg. Members per group	274.13
Avg. Groups a user joins	3.54
Avg. Events per group	72.26
Avg. Participants per event	5.67
Avg. Events per active user	9.38

TABLE 2
Dataset Statistics

4.2 Seasonal Pattern of Group Activities

As mentioned in Section 3.1, there may be seasonal patterns in the groups' activities depending on the topics of the groups. For example, some "Outdoor&Adventure" groups tend to have their events concentrated only in a few months in a year, such as "skiing-related" and "soccer-related" groups, while some other groups with "seminar-related" and "meditation-related" topics tend to have more even distributions of events over the 12 months. To show this, we choose four sets of groups with the topics of "skiing", "soccer", "seminar" and "meditation" from the 100 most active groups based on keywords searching. In Fig 7 we plot the average RSVP numbers of each month with 12 months in a year. We observe that for ski and soccer related groups, the peak activities occur in November∼February and June∼September respectively, while the seminar and meditation related groups have flat distributions over 12 months. This is due to the fact that the indoor activities are less affected by seasons. For this reason of seasonal pattern, we will present the results regarding any prediction interval instead of averaging the performance over all months in the following sections.

4.3 Group Popularity Prediction

In this section, we show how member-level and group-level features can be used to predict the popularity of groups at different prediction intervals. Unlike the experiment settings introduced by

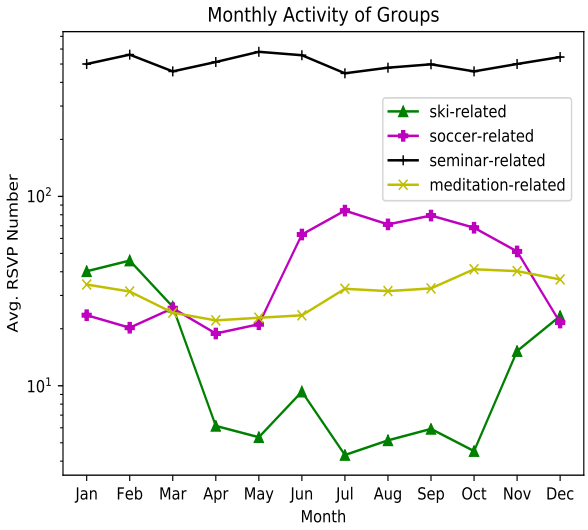

Fig. 7. Monthly Activity of Groups with Different Topics

[10], [9] and [13], which include all groups of different sizes and ages, we only focus on predicting future RSVP numbers of new groups in each year. In our experiments, features are extracted from the first three months starting from the time when a newly formed group held its first event. We then make prediction of the RSVP number within another time window of three months in the future after a time interval ranging from one month to ten months. The predicted RSVP numbers are tested against the true RSVP numbers. We use the Normalized Root Mean Squared Error (nRMSE) to measure RSVP prediction accuracy:

$$nRMSE = \sqrt{\frac{1}{n}\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\hat{y}_i{}^2}},$$

where y_i is a target group's actual RSVP number, $\hat{y}_i$ is the predicted RSVP number and n is the number of groups. Since we only focus on predicting the RSVP number of newly created groups, the nRMSE is **not** dominated by very small or very large groups in our dataset.

After filtering out the groups without valid information to calculate the features, we have more than 7,000 new groups along with their features and RSVP numbers. We randomly choose 6,000 groups and perform five-fold cross-validation to select the best hyper-parameters used in all the baselines and GRNF algorithm. The parameters used in the baselines are shown in Table 3.

Baseline Parameters	Value
Tolerance for stopping criterion	1e-3
C/epsilon/kernel/degree in SVR	1.0/0.1/rbf/3
Layer sizes in MLP	(10×10)
Activation function in MLP	relu
Number of estimators in RF	200

TABLE 3
Parameters for Training Baselines

4.3.1 GRNF vs. Raw Member Features

To demonstrate the Group-role Neural Fingerprints (GRNF) can truly improve the prediction accuracy and avoid the influence of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

group-level features, we first conduct group popularity prediction with raw member features using the classic machine learning methods (Linear Regression, Support Vector Regression, Multi-layer Perceptron, and Random Forest) as well as two advanced machine learning methods (XGBoost and LightGBM). We then input the raw member features with adjacency matrix of the social graph to our proposed group-role neural fingerprints algorithm. Finally, the obtained GRNF are input to MLP to predict group popularity. The parameters used to train the neural network are shown in Table 4.

GRNF Parameters	Value
Number of batches	50
Batch size	100 (graphs)
Learning rate	1e-3
Activation function	relu
L1 regularization (λ_1)	1e-4
L2 regularization (λ_2)	1e-4
Fingerprint length	10
Convolution layer sizes	$(10 \times 10 \times 10)$
Fully connected layer sizes	(3×2)

TABLE 4
Parameters for Training GRNF

In Table 5 we show the nRMSE of each method at different prediction intervals. Predictions for nearer future (e.g. 1 month in future) are more accurate than the ones in farther future (e.g. 10 month in future). Based on this observation we may extrapolate that the nRMSE for more than 1 year in future could increase to more than 20% and it also shows that predictions of RSVP number of new groups could be difficult. When comparing any baseline with GRNF we first calculate the average errors of the baseline as u1 and average errors of GRNF as u2, we then formulate our null hypothesis as: $H0 : u1 = u2$, and the alternative hypothesis as: $Ha : u1 > u2$. We choose the critical p-value as 0.05 so that if the resulting p-values are smaller than 0.05 we can reject the null hypothesis. We can see from Table 5 that GRNF can significantly reduce the prediction errors of raw member feature based predictions at different prediction intervals (1 to 10 months). The performance improvement is statistically tested by the t-test scores presented in last two columns. The last column shows the largest p-value among all "baseline-GRNF" comparison. The reason that GRNF can outperform classical methods (LR, MLP, SVM and RF) as well as more advanced boosted algorithms (XGBoost and LightGBM) is due to the ability of exploiting extra information of role-based structural features extracted from the social graphs.

4.3.2 Comparison with Baseline Methods

We compare our method which uses both group-level features and internal features (shown in Table 1) with three competitive baselines:

- Baseline 1 [10]: in addition to meta information about the groups, such as "number of group members", "number of events" and "group join mode" etc., it also uses the averaged member-level features, such as "average event attendance of members" and "standard deviation of event attendance of members" etc. "Structural features" are introduced based on a bipartite graph generated by events and users, without distinguishing the user types.
- Baseline 2 [13]: it demonstrates that structural features like triads counts and clustering coefficients have strong predictive power for predicting the longevity of the group's lifecycle in an online social messaging network.
- Baseline 3 [11]: it uses epidemic model of differential equations to fit the evolution curve of group's popularity. One advantage of this model is that it provides decent accuracy by only using the time series of daily active users (DAU). This simple baseline acts as a sanity-check to whether the time series of the group's past popularity (the number of RSVPs) alone could determine the group's future popularity.

For baseline 1&2, we implement most of the original features and apply four classical machine learning algorithms: Linear Regression, Support Vector Regression, Multilayer Perceptron, and Random Forest with the best parameters (Table 3) selected by cross-validation, and choose the one with the best performance as the representative for each baseline. For baseline 3, we use the DAU of a group in the first three months to fit the curve and the rest time to test the accuracy. The parameter tuning for baseline 3 follows the procedure in the original paper [11].

Table 6 shows nRMSE for our proposed method comparing with 3 baselines regarding to increasing prediction intervals. It shows that our final proposed approach (Combination Method 3) clearly outperforms all the baselines in all prediction horizons, ranging from predicting the average 3-month RSVP numbers in the immediate next three months to predicting this quantity ten months after the last record in the training data. As expected, for all methods, the error of predicting nearer future is smaller. Thus, it is important to compare the accuracy gains of our method against all baselines, which range from 3.91% to 12.32%. The statistical significance of the performance improvement is verified by the p-values reported in the last column of Table 6. The p-values shown in the table represent the largest p-values among all three "baseline-GRNF" comparison. We also see similar results for different averaging windows (4 and 5 months). Due to space limit, we don't include the results here.

4.3.3 Impact of Different Combination Methods

Since our predictions are made through two independent feature sets, we can combine features using different methods proposed in the previous section. In Table 7 we compare the accuracy of the three combination methods. For method 2, we try two clustering methods: K-means and DBSCAN (choose the one with better performance). We can see that method 3, the DNN-based combination, yields the best average nRMSE=11.94% for all prediction intervals, improving the best baseline (baseline 1) by 11.3%.

4.4 Member Role Analysis

We now analyze the results produced by role discovery method and try to answer the question: "*who are the members contributing the most to a group's future popularity?*"

As detailed in the previous section, we can get the role distribution vector for a group by aggregating the role distribution vectors of all its members. In addition, every member is assigned to a role according to the largest element in her role distribution vector. By averaging the feature vectors of all members assigned to each role, we obtain a representative feature vector for each role to study user's typical behavior.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

TABLE 5
nRMSE of Raw Member Features vs. nRMSE of Group-Role Neural Fingerprints (GRNF)

Interval (months)	Raw Member Features(%)						GRNF(%)	Gain(%)	$\|t-stat\|$	p-value
	LR	SVR	MLP	RF	XGBoost	LightGBM				
0	10.98	11.02	20.81	11.05	10.83	10.81	**10.51**	2.71	> 1.14	< 0.009
1	11.65	11.62	13.09	11.94	11.73	11.91	**10.57**	9.08	> 1.17	< 0.009
2	12.41	12.32	18.11	13.19	12.97	13.19	**11.41**	7.41	> 1.15	< 0.01
3	13.04	13.04	29.34	13.74	13.35	13.21	**11.88**	8.86	> 1.07	< 0.016
4	13.91	13.92	22.94	14.15	13.91	13.77	**12.29**	10.77	> 1.17	< 0.014
5	14.27	14.26	24.54	14.13	13.78	13.77	**12.93**	6.14	> 1.26	< 0.008
6	15.27	15.26	15.54	15.05	14.57	14.71	**13.61**	6.61	> 1.69	< 0.005
7	16.73	16.58	40.27	16.95	16.56	16.66	**15.08**	8.94	> 1.33	< 0.008
8	17.33	17.25	16.43	17.21	16.71	16.88	**15.61**	4.97	> 1.24	< 0.01
9	17.44	17.44	42.28	17.82	17.31	17.51	**15.81**	8.64	> 1.11	< 0.01
10	18.01	18.05	20.24	18.16	17.44	17.55	**16.75**	3.97	> 1.13	< 0.009

Horizon (months)	Baseline 1 (best perf.)	Baseline 2 (best perf.)	Baseline 3	Proposed Method	Gain over Best Baseline (p-value)
0–3	9.86	11.11	12.16	**8.78**	**10.90%** (< 0.012)
1–4	10.55	11.90	12.39	**9.44**	**10.61%** (< 0.012)
2–5	11.72	12.62	12.64	**10.41**	**10.68%** (< 0.015)
3–6	12.47	13.29	12.77	**11.27**	**8.53%** (< 0.022)
4–7	12.21	12.77	12.76	**11.73**	**3.91%** (< 0.021)
5–8	13.81	13.05	12.72	**11.36**	**10.73%** (< 0.012)
6–9	13.44	13.78	13.78	**11.73**	**12.69%** (< 0.009)
7–10	14.86	16.41	14.85	**13.02**	**12.32%** (< 0.011)
8–11	15.67	17.65	16.28	**13.87**	**11.51%** (< 0.014)
9–12	16.18	16.37	16.51	**14.34**	**11.39%** (< 0.014)
10–13	16.90	17.77	18.05	**14.90**	**11.42%** (< 0.011)

TABLE 6
Prediction Accuracy (nRMSE) Comparison with Baseline Methods

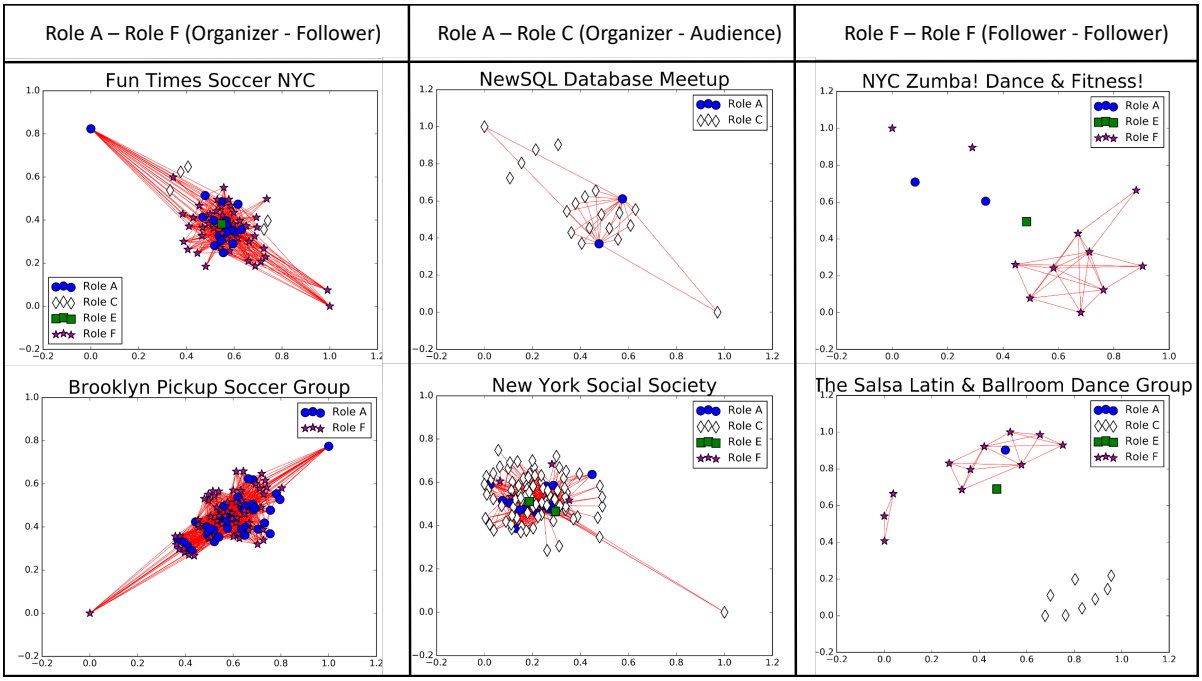

Fig. 8. Subgraphs Contributing the Most to Group Popularity. Three types of learned subgraphs (radius=1) are: role A-F, role A-C, and role F-F. For each subgraph type, two example groups with high popularity (top-100 popular in Meetup) that contain this type of subgraph are shown.

Inspired by [24], we interpret each role by analyzing the representative feature vector for each role and comparing each role's "own-feature part" and its "neighbor-feature part" in its representative feature vector, that is, we infer the role of a user by analyzing "who she is" and "who she knows". For example, if we observe a user has many more connections with her friends (reflected by the degree of the node representing the user in the social graph defined in Definition 2) than all her friends have, then we infer she is probably the most active user in her local social network, and we can further infer that there is a good chance that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

Combining Method	Method 1	Method 2	Method 3
Average nRMSE	12.77%	12.74%	11.94%
Gain from Best Baseline (%)	5.41	5.42	11.3

TABLE 7
Performance of Three Combination Methods

she is an event/group organizer in reality. In our experiments, we set the role number to six to achieve the best prediction result, then we get the representative feature vectors for the six roles. Note that the roles obtained from NMF are latent and cannot be explicitly obtained from data. Our interpretation of the roles based on the typical features obtained by the NMF is speculative, but is nevertheless corroborated by other observations from the data.

Role A (Group Organizers): The user potentially interacts with a large number of people (average own degree=3,568.24) and is more active than her neighbors (average neighbor's degree=473.14). Together with other information about Role A's behaviors, such as joining very few groups (average group number=1.12) and attending as many events as possible (attended event fraction=0.73), we infer that the user is potentially a successful event organizer. We also find that 33.6% of the users assigned to Role A have hosted (sent the first RSVP "yes") at least one event, which is higher than the percentages in other roles. The correlation between Role A and group RSVP number is 0.514.

Role B (Inactive Followers): Compared with other roles, the user's neighbors are much more active than the user herself (average own degree=2.19, while average neighbors' degree=5,550.03), indicating that role B may be an inactive user who only participates in the events with the user's friends who have strong social influence. The correlation with group RSVP number is 0.515.

Role C (Conference Audience): The user is normally just a participant of a seminar. Large average own degree (4206.95) and behavior similarity with her neighbors suggest that Role C may be one of the conference audience. This role has the lowest correlation with the RSVP number (correlation=0.482).

Role D (Inactive Users): Similar to role B, but the user's neighbors have less interactions with others and her neighbors have less "influence" than Role B. So role D may be an inactive member who follows her own interest. The correlation with group RSVP number is 0.623.

Role E (Small Group Organizers): The user seems like a *scaled-down* version of role A, with lower own and neighbor activity levels. This user could be an event organizer who specializes in small events. The correlation with group RSVP number is 0.555.

Role F (Dedicated Followers): Compared to other roles, this user has high activity level (average own degree=225.35) and potentially prefers attending social-based events than conference-based events, since the average event size (225.35/10.39=21.68) she attends is much lower than Role C (4206.95/70.58=59.61). Also, the high activity level of her neighbors (average neighbor degree=1578.18) suggests that Role F may represent the dedicated followers of the "social influencers". This role has the highest correlation with the RSVP number (correlation=0.700).

4.5 Social Structure Analysis

In GRNF, we select the members whose related features activate the hidden units the most at each radius, then use their connections with neighbors to represent the subgraphs contributing the most to the group's success. In Figure 8 we show some sampled subgraphs at radius=1, each member is labeled with her assigned role. All the sampled subgraphs come from the most popular groups whose 3-month RSVP numbers are ranked within top 100 among all groups in Meetup. The structures of their social graphs in the first 3 month exhibit clear "Role A - Role F", "Role A - Role C" or "Role F-Role F" pattern. On the other hand, when we examine unpopular groups, we couldn't find the previous three clear graph patterns.

Role A (Organizers) - Role F (Followers): Normally seen in social-based groups such as popular soccer groups. There is an organizer (role A) for each game, and several active players (role F). Role C (Conference Audience) seldom exists in such group.

Role A (Organizers) - Role C (Audience): Frequently seen in technical seminars. A large number of role C (Audience) members exist in the group with only one or two organizers. The intensity of social connections between members in this type of groups is weaker than what is common in more social-based groups.

Role F (Followers) - Role F (Followers): Normally seen in dancing groups. Similar with Role A - Role F structure, the social intensity in this type of groups tends to be strong. They differ in that the connections among ordinary members are stronger.

Interestingly, we find that topics of the groups are highly correlated to the role distribution and structures of their social graphs. For example, soccer groups like "*Fun Times Soccer NYC*" and "*Brooklyn Pickup Soccer Group*" tend to have a lot of "Role A-Role F" structures, technical seminars like "*New SQL Database Meetup*" and "*New York Social Society*" normally are full of "Role A-Role C" structures, and dancing groups, such as "*NYC Zumba! Dance and Fitness*" and "*The Salsa Latin&Ballroom Dance Group*", are likely to have more "Role F-Role F" structures. On the other hand, we can see that if a group organizer wants to gain more group popularity in the future, she probably needs to build up such interaction patterns in an early stage.

5 Conclusion

In this paper, we proposed a deep neural network method to predict the future popularity of groups in event-based social networks. Our method outperformed all the state-of-the-art methods. Along the way, we have analyzed a few key factors contributing the most to these predictions. Specifically, we showed that location and time are important group-level features. We also demonstrated that member roles and interaction among members with different roles, characterized by neural fingerprints, can better represent the intrinsic member behaviors and the social structure of a group than the raw member features and the activity graph. Through case studies in Meetup, we showed that the most relevant user roles to a group's popularity are not "organizer-like members", but "ordinary members". We also found that the most important substructures are not combining all the most important roles, but follow different combination patterns for different types of groups.

References

[1] (2017) Meetup. [Online]. Available: https://en.wikipedia.org/wiki/Meetup_(website)

[2] D. Rogers and M. Hahn, "Extended-connectivity fingerprints," *Journal of chemical information and modeling*, vol. 50, no. 5, pp. 742–754, 2010.

[3] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in neural information processing systems*, 2015, pp. 2224–2232.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

TABLE 8
Typical Feature Values for Different Roles (role number=6)

	Role A	Role B	Role C	Role D	Role E	Role F
Correlation	0.514	0.515	0.482	0.623	0.555	0.700
total degree	3568.24	2.19	4206.95	2.66	66.38	225.35
event number	47.64	2.33	70.58	1.32	14.31	10.39
group number	1.12	1.78	19.58	1.34	8.64	1.07
cross group entropy	0.043	0.04	2.28	0.03	1.41	0.024
attended event fraction	0.73	0.02	0.22	0.02	0.25	0.58
attended event fraction entropy	0.022	0.01	1.97	0.03	1.19	0.016
neighbor's total degree	473.14	5550.03	3236.31	61.47	0.001	1578.18
neighbor's event number	17.43	7.59	36.22	6.64	26.97	38.92
neighbor's group number	12.98	6.11	12.31	5.32	5.36	8.932
neighbor's cross group entropy	1.94	1.19	1.87	1.23	1.02	1.42
neighbor's event fraction	0.21	0.26	0.21	0.17	0.24	0.22
neighbor's event fraction entropy	1.64	0.88	1.58	1.18	0.78	1.14

[4] X. Liu, Q. He, Y. Tian, W.-C. Lee, J. McPherson, and J. Han, "Event-based social networks: linking the online and offline social worlds," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 1032–1040.

[5] A. Q. Macedo, L. B. Marinho, and R. L. Santos, "Context-aware event recommendation in event-based social networks," in *Proceedings of the 9th ACM Conference on Recommender Systems*. ACM, 2015, pp. 123–130.

[6] H. Yin, Y. Sun, B. Cui, Z. Hu, and L. Chen, "Lcars: a location-content-aware recommender system," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 221–229.

[7] W. Zhang, J. Wang, and W. Feng, "Combining latent factor model with location features for event-based group recommendation," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 910–918.

[8] H. Ding, C. Yu, G. Li, and Y. Liu, "Event participation recommendation in event-based social networks," in *International Conference on Social Informatics*. Springer, 2016, pp. 361–375.

[9] S. Pramanik, M. Gundapuneni, S. Pathak, and B. Mitra, "Can i foresee the success of my meetup group?" in *Advances in Social Networks Analysis and Mining (ASONAM), 2016 IEEE/ACM International Conference on*. IEEE, 2016, pp. 366–373.

[10] X. Liu and T. Suel, "What makes a group fail: Modeling social group behavior in event-based social networks," in *Big Data (Big Data), 2016 IEEE International Conference on*. IEEE, 2016, pp. 951–956.

[11] B. Ribeiro, "Modeling and predicting the growth and death of membership-based websites," in *Proceedings of the 23rd international conference on World Wide Web*. ACM, 2014, pp. 653–664.

[12] T. Zhang, P. Cui, C. Faloutsos, Y. Lu, H. Ye, W. Zhu, and S. Yang, "Come-and-go patterns of group evolution: A dynamic model," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 1355–1364.

[13] J. Qiu, Y. Li, J. Tang, Z. Lu, H. Ye, B. Chen, Q. Yang, and J. E. Hopcroft, "The lifecycle and cascade of wechat social messaging groups," in *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 311–320.

[14] G. Li, Y. Liu, B. Ribeiro, and H. Ding, "On group popularity prediction in event-based social networks," in *Twelfth International AAAI Conference on Web and Social Media*, 2018, Poster.

[15] R. A. Rossi and N. K. Ahmed, "Role discovery in networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 4, pp. 1112–1131, 2015.

[16] S. Gilpin, T. Eliassi-Rad, and I. Davidson, "Guided learning for role discovery (glrd): framework, algorithms, and applications," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 113–121.

[17] Y. Ruan and S. Parthasarathy, "Simultaneous detection of communities and roles from large networks," in *Proceedings of the second ACM conference on Online social networks*. ACM, 2014, pp. 203–214.

[18] Y. Zhang, B. Adhikari, S. T. Jan, and B. A. Prakash, "Meike: Influence-based communities in networks," in *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 2017, pp. 318–326.

[19] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 701–710.

[20] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 855–864.

[21] L. F. Ribeiro, P. H. Saverese, and D. R. Figueiredo, "struc2vec: Learning node representations from structural identity," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 385–394.

[22] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017, pp. 135–144.

[23] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001, pp. 556–562.

[24] K. Henderson, B. Gallagher, T. Eliassi-Rad, H. Tong, S. Basu, L. Akoglu, D. Koutra, C. Faloutsos, and L. Li, "Rolx: structural role extraction & mining in large graphs," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 1231–1239.

Guangyu Li is a Ph.D. candidate at the Electrical and Computer Engineering department of New York University. He received his bachelor and master degree in the field of automatic control from Electrical and Computer Engineering department at Tianjin University, Tianjin, China in July 2011 and 2014 respectively. His general research interests lie in data mining methods in networking and recommender system.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TNSE.2019.2916893, IEEE Transactions on Network Science and Engineering

Yong Liu is an associate professor at the Electrical and Computer Engineering department of New York University. He joined NYU-Poly as an assistant professor in March, 2005. He received his Ph.D. degree from Electrical and Computer Engineering department at the University of Massachusetts, Amherst, in May 2002. He received his master and bachelor degree in the field of automatic control from the University of Science and Technology of China, in July 1997 and 1994 respectively. His general research interests lie in modeling, design and analysis of communication networks. His current research include P2P systems, overlay networks, network measurement, online social networks and recommender systems. He is the winner of ACM/USENIX Internet Measurement Conference (IMC) Best Paper Award in 2012, IEEE Conference on Computer and Communications (INFOCOM) Best Paper Award in 2009, and IEEE Communications Society Best Paper Award in Multimedia Communications in 2008. He is a Fellow of IEEE.

Ribeiro is an Assistant Professor of Computer Science at Purdue University. Bruno Ribeiro received his PhD from University of Massachusetts Amherst in 2010 and was a postdoctoral fellow at Carnegie Mellon University between 2013 and 2015. His interests are in machine learning and modeling complex dynamical systems.

Hao Ding received a BS degree in Physics from University of Science and Technology of China in 2008 and a PhD degree in Electrical Engineering from New York University in 2017. He is a senior data scientist in the Oath Inc. His research spans the area of machine learning and data mining with an emphasis on online social network based recommender systems. He has work experience in several social network tech companies. He helped design the machine learning based recommender for LinkedIn email system, music playlist recommender for mobile music MILK at Samsung, and personalized place recommender in the Places team at Facebook. He was also one of the chief designers of Facebook 3rd-party app, MvRock, a personalized music video player and recommender. Now his main work at Oath is designing blog recommenders for Tumblr.