Proceedings of the Twelfth International AAAI Conference on Web and Social Media (ICWSM 2018)

On Group Popularity Prediction in Event-Based Social Networks

Guangyu Li,[1] Yong Liu,[1] Bruno Ribeiro,[2] Hao Ding[1]
[1]The Department of Electrical and Computer Engineering, New York University, Brooklyn, NY
[2]The Department of Computer Science, Purdue University, West Lafayette, IN
guangyu.li, yongliu, hao.ding@nyu.edu, ribeirob@purdue.edu

Abstract

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 RMSE of the popularity prediction (measured in RSVPs) of a group's future events by up to 12%, against the state-of-the-art baselines.

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 (Meetup.com 2018) 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.

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?*

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. Specifically, armed with each group member's role, and event co-participation graphs generated from those members' 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.

Related Work

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 user", without user-level or network structural information (Ribeiro 2014). The other focuses on using general group features to make predictions (Liu and Suel 2016; Qiu et al. 2016). 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 accuracies.

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.

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 responds by sending RSVPs ("yes", "no", or "maybe"). With the definitions of groups, events and users in Meetup, the group popularity prediction problem can be defined as:

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.

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.

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 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.

Member-level Feature Extraction We start with constructing social graph for each group from which the features are extracted. 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.

Member Role Discovery To find role features of each group member, we use Non-negative Matrix Factorization (NMF) (Lee and Seung 2001).

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 $\mathbf{V^g} = \left\{\mathbf{V^g_j} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{M^g_{ij}}, \mathbf{1 \le j \le r}\right\}$, then we stack all the vectors $\{\mathbf{V^g}\}$ to form a group-role matrix $\mathbf{\Omega} \in \mathbb{R}^{\mathbf{p \times r}}$ where p represents the number of groups and r is the number of roles.

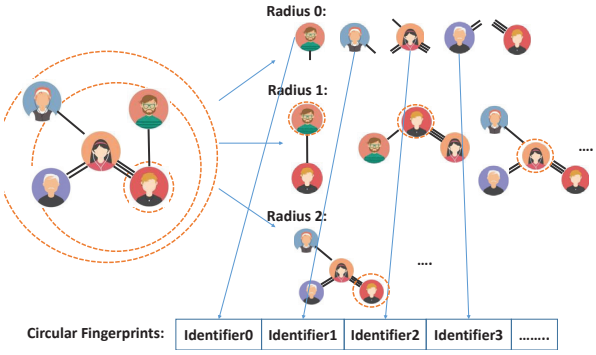

Figure 1: Subgraphs Detected by Circular Fingerprints in Social Network. Circular Fingerprint scans the network for all subgraphs under certain radius. Each subgraph is then encoded into an integer identifier. Integer identifiers of all subgraphs constitute the circular fingerprints.

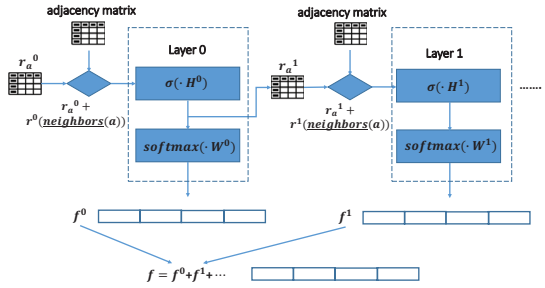

Figure 2: Group-Role Neural Fingerprints Algorithm. "Feature Update" operation: each member's feature is updated so that the 1-hop neighbors' features and neighboring edge features are added to the initial member feature.

Circular Fingerprints In order to extract structural features embedded with member roles, we use the "circular fingerprints" algorithm (Rogers and Hahn 2010). Circular fingerprints is a popular tool for handling graph-structured data in chemistry. 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 1). Then we can study how the subgraphs between members contribute to the group popularity using the circular fingerprints framework. 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.

Group-Role Neural Fingerprints Overcoming some limitations, Duvenaud et al. (Duvenaud et al. 2015) proposed a convolutional neural network, where each neural network layer simulates the updating operation in circular fingerprints (Figure 2). We now extend the convolutional finger-

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. Event number the member has participated in current group m3. Group number the member has joined m4. Entropy of member's attendance distribution over the groups the member joined m5. Entropy of event number distribution over the groups the member joined m6. Entropy of event fraction distribution over the groups the member 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

print algorithm 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, 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 are captured.

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.

Combining Group-level and Internal Features We use 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).

Performance Evaluation

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

Group Popularity Prediction

Unlike the experiment settings introduced by (Liu and Suel 2016) and (Qiu et al. 2016), 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 Root Mean Squared Error (RMSE) to measure RSVP prediction accuracy. After filtering out the groups without valid information to calcu-

Horizon (months)	Baseline 1 (best perf.)	Baseline 2 (best perf.)	Baseline 3	Proposed Method	Gain over Best Baseline (p-value)
0–3	83.92	94.61	103.53	**74.78**	**10.90%** (< 0.012)
1–4	89.85	101.31	105.44	**80.32**	**10.61%** (< 0.012)
2–5	99.81	107.45	107.64	**88.62**	**10.68%** (< 0.015)
3–6	106.15	113.11	108.71	**95.96**	**8.53%** (< 0.022)
4–7	103.91	108.74	108.61	**99.84**	**3.91%** (< 0.021)
5–8	117.58	111.13	108.32	**96.69**	**10.73%** (< 0.012)
6–9	114.38	117.28	117.28	**99.85**	**12.69%** (< 0.009)
7–10	126.52	139.67	126.41	**110.83**	**12.32%** (< 0.011)
8–11	133.41	150.28	138.61	**118.07**	**11.51%** (< 0.014)
9–12	137.75	139.36	140.54	**122.04**	**11.39%** (< 0.014)
10–13	143.83	151.24	153.67	**126.86**	**11.42%** (< 0.011)

Table 3: Prediction Accuracy (RMSE) Comparison with Baseline Methods

late 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.

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 (Liu and Suel 2016): in addition to meta information about the groups, it also uses the averaged member-level features, such as "average event attendance of members" and "standard deviation of event attendance of members" etc.
- Baseline 2 (Qiu et al. 2016): 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 (Ribeiro 2014): 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).

The results in Table 3 show that our final proposed approach clearly outperforms all 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 contrast the accuracy gains of our method against all baselines, which range from 3.91% to 12.32%.

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.

References

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

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

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

Meetup.com. 2018. Meetup.com — Wikipedia, the free encyclopedia. [Online; accessed 19-January-2018].

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

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

Rogers, D., and Hahn, M. 2010. Extended-connectivity fingerprints. *Journal of chemical information and modeling* 50(5):742–754.