Physica A 505 (2018) 222–232

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Inferring parent–child relationships by a node-remove centrality framework in online social networks

Xiao-Ke Xu [a,b], Xue Wang [a], Jing Xiao [a,*]

[a] *College of Information and Communication Engineering, Dalian Minzu University, Dalian 116600, China*
[b] *Guizhou Provincial Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China*

HIGHLIGHTS

- Parent–child links are weak social embeddedness (weak tie) instead of strong embeddedness (strong tie).
- A novel node-remove framework is proposed to measure the node centrality considering its direct and indirect effect.
- The proposed algorithm is validated in online social networks and better than several representative algorithms.
- Our method may not only infer parent–child and familial relationships, but also detect other heterogeneous links.

ARTICLE INFO

Article history:
Received 30 October 2017
Received in revised form 16 February 2018
Available online 29 March 2018

Keywords:
Online social networks
Node centrality
Dispersion
Embeddedness
Strong ties

ABSTRACT

Online social networks can represent various kinds of relationships between users, so a significant task is to infer specific relationships, especially family members or romantic partners, by analyzing network topologies. In this study, we explored to infer a special kind of family links (i.e., parent–child links) in the QQ social network (the largest online social network of China) based on multiple node centrality algorithms. We found that most parent–child links are weak social embeddedness (weak tie) instead of strong embeddedness (strong tie), resulting that such kind of links are very difficult to detect in the user's ego social network. To get a high accurate detection, we firstly utilized user profile information such as age and gender to filter out the set of potential links and then combined network structure mining. We proposed a novel node-remove framework to measure a node centrality considering its direct and indirect effect on network topologies, and our method obtained a 11.3% higher performance than "dispersion" which is a new network centrality proposed in a recent study. We also found that calculating indirect effect by using a semi-local structure can obtain a better performance than using local or global structures, which suggests that parent–child links have a meso-scopic topology effect. Our method may not only infer parent–child relationships effectively, but could also detect other hierarchical relationships, such as manager–subordinate and advisor–advisee ties in online social networks.

1. Introduction

In the last ten years, more and more people have begun to embed themselves in online social networks and reorganize social relationships based on social network services. The heavy users of online social networks not only tend to connect to

[*] Corresponding author.
E-mail address: hrbeuxiaojing@aliyun.com (J. Xiao).

https://doi.org/10.1016/j.physa.2018.03.041
0378-4371/© 2018 Published by Elsevier B.V.

their friends driven by homophily principle [1,2], but also link their family members to maintain intimate relationships by online interactions instead of offline face to face communications [3,4]. For example, Madden et al. found that two thirds of parents of teens aged 12–17 at least use a social networking site, and 80% of them have friended their children [5]. Furthermore, nearly 40% of Facebook users have either the parent or child on the site, and their communication frequencies do not decrease with the increasing of geographic distance [3]. Recently, using real-life data from a large-scale sample of Facebook users, Backstrom and Kleinberg found that romantic partnerships in social network neighborhoods can be accomplished with high accuracy according to the dispersion of social ties [6].

In this study, we are interested in how to infer the parent or child link based on a user's ego network structure [7]. Parent–child relationship is a singular type of social strong ties that plays a powerful role in social processes over a person's whole life course [8]. Parent–child links can be classified into two categories: parent-to-child and child-to-parent. Inferring parent-to-child links means that we have gotten the ego social network of a parent and we try to identify the hidden child in his/her friend list. Parent-to-child links include four sub-categories: father–son, father–daughter, mother–son, and mother–daughter. Here "father–son" means that the center user is a father, and he has tagged one of his QQ friend as his son. Child-to-parent relationship contains other four types of links: son–father, son–mother, daughter–father, and daughter–mother. Similar to "father–son", "son–father" means that the center user is a son, and he has tagged one of his QQ friend as his father.

Strong embeddedness [9,10] is the classic view to investigate the special importance of parent–child links. However, our results show that parent–child links are rarely found in the tie sets of the strong network embeddedness (strong ties). Especially, the results of the QQ dataset indicate that the ratio of children regarding their parents as the most strong embedding ties are less than that parents regarding their children as the most strong embedding ties. The embeddedness strengths of parent–child links are not as strong as we expect, which makes them difficult to be extracted from the center user's social network neighborhood. The reason for this is that there is a huge generation age difference between parent and child, so we try to utilize each user's age and gender as the filter to obtain a set of potential links before inferring parent–child links. By combining node centrality statistics, we can greatly improve the prediction accuracy for such kind of links. Our results suggest that a relative strong embeddedness instead of an absolute strong one exists between parents and children (the strongest tie in all the pairs of users who have an age gap of 15–40 years), because their social circles are not strongly overlapped induced by parent–child huge age difference.

Dispersion is another view to consider the importance of parent–child links. Dispersion was proposed in [6] to infer romantic relationships in Facebook by measuring whether two persons' mutual friends are well-connected. If the mutual friends of two users are not well connected to one another and have a long shortest path length, the two users have a high dispersion value. Compared with other node centrality statistics, dispersion centrality obtains the best performance for inferring parent–child links. However, several formations can be adopted to calculate dispersion, and it is difficult to know which formation is the best. In this study, we developed a novel node-remove framework to explain why the threshold-3 format of dispersion centrality can obtain the best performance. Furthermore, the proposed node-remove framework fuses the information of both embeddedness and dispersion centrality, which leads to a better detection performance. At last, the results suggest that parent–child links have a meso-scopic topology effect, which means calculating the indirect effect by using a semi-local structure can obtain a higher accuracy performance than using local or global structure information. Our findings can be used to not only infer parent–child links and other relative links, but also detect other heterogeneous types of links, such as manager–subordinate and Ph.D. advisor–advisee relationships in online social networks [11].

2. Dataset and problem description

2.1. Dataset description

Tencent QQ, popularly known as QQ in China, is an instant messaging software service. As of the first quarter of 2017, QQ had 861 million monthly active user accounts. At the highest peak, more than 266 million people were using QQ simultaneously. As a scientific research cooperation project with Tencent company, we have been permitted to use a large-scale dataset of randomly sampled QQ users who declare his/her child or parent in their friend lists. How to draw samples that can represent the QQ OSNs has remained a formidable task because of a number of conceptual and methodological reasons [12]. Here we take the uniform sampling method. Uniform sampling is a classic method based on probability theory that has been widely used in sampling of physical and human subjects for centuries [13]. When applied to network sampling, all nodes of any OSN under study are given an equal chance (hence the name of "uniform") to be sampled. Large number theorem ensures that a uniform sample, with a sufficiently large number of nodes, will represent accurately the underlying population (i.e., the QQ OSN). The dataset consists of 842 QQ users' ego network neighborhoods and 245,371 users, selected uniformly at random from all active users who list a parent or child in their friend lists. The neighborhood of each user has an average of 291 friends, which means that the accuracy of random selecting parent–child link is only about 0.3%. More detail information on this dataset can be found in Table 1. All the data in our analysis was used anonymously, and all the analysis was done in aggregate.

All the links between parent and child can be divided into two types: parent-to-child and child-to-parent. Parent-to-child links can be classified as four categories: father–son (fa–son), father–daughter (fa–dau), mother–son (mo–son), and mother–daughter (mo–dau). Here, the "father–son" relationship means that the center user is a father, and he has tagged one of his QQ friend as his son. In this study, our aim is to identify which user is his son based on the father's QQ ego social network.

Table 1
Basic description of the QQ social network dataset. N represents the number of each kind of users, M represents the total number of friends for each kind of users. R represents the value of M/N, which is used to measure the difficulty of picking out the specific link in all the neighborhood links. Fa is the abbreviation of father, mo is the abbreviation of mother, dau is the abbreviation of daughter, pa is the abbreviation of parent, and chi is the abbreviation of child.

Statistic	fa–son	son–fa	fa–dau	dau–fa	mo–son	son–mo	mo–dau	dau–mo	pa–chi	chi–pa	all
N	97	107	26	164	63	120	91	174	277	565	842
M	20 794	35 138	4904	50 995	12 248	47 777	15 708	57 807	53 654	191 917	245 371
R	214	328	189	311	194	398	173	332	194	340	291

And child-to-parent links can be classified as another four categories: son–father (son–fa), daughter–father (dau–fa), son–mother (son–mo), and daughter–mother (dau–mo). We search uniformly at random from active users to select the user who has a parent–child or child–parent relationship, and the number of child-to-parent links is far more than the number of parent-to-child links (565 vs. 277). Comparing the neighborhoods of parent-to-child and child-to-parent links in Table 1, the user who declares a parent in its friend list has more friends than the user who declares a child in the friend list (340 vs. 194). The main reason for the above results is that the QQ users are mainly young people (the average age of users in our dataset is 24.3 years old) and they are also more active than older users.

2.2. Problem description

In this study, we try to identify parent–child relationships in Tencent QQ users' ego social networks. We use a dataset of randomly sampled QQ users who declare his/her child or parent in their friend lists. Each QQ user can name himself/herself with a nickname in the profile and the nickname will be shown to other users who are friending with him/her. If other users know this user's real name or the relationship with the user, they can choose to remark him/her the real name or link type. We random select the users who list one of their friends as "father", "mother", "son" and "daughter" in Chinese in their ego social networks. Then we ask how to utilize the user's network neighborhood (the set of all friends and the links among them, which is shown in Fig. 5(a)) to identify the hidden parent or child.

We need to emphasize that our work is strongly encouraged by the pioneer works of Backstrom and Kleinberg based on a Facebook dataset [6]. At the beginning of this study, we also try to infer the romantic relationships in QQ social networks using the same frameworks proposed in [6], but we find it is difficult to verify our results. One reason is that QQ and Facebook are different types of online social networks. QQ is an anonymous personal instant messaging software (like MSN Messenger), so a male user can mark many female friends as girl friends (or wife) while no one can verify the information is true or not. The other reason is that even one user's girl friend or spouse really exists in the user's friend list, he may not mark her using a common name like "girl friend" or "wife". For example, a young man may name his girlfriend as "Sweetheart" or "Love" instead of "girl friend". Diverse nicknames for "girl friend" increase the difficulty to extract users' romantic partner information, especially in the case of Chinese natural language processing. Anyway, there are many difficulties hindering us from inferring romantic relationships by the method proposed in [6] in the QQ online social network. Fortunately, the family members can also be inferred by the same method in the pioneering work [6] and Tencent company was willing to provide the corresponding data, so we try to infer parent–child links in this study.

The parent–child relationships has two types: parent-to-child and child-to-parent. In each ego social network, we only identify one type of parent–child relationships. Although QQ is an anonymous social network software, we believe that the users will not declare false child-to-parent relationships. According to an internal survey in Tencent company and the single-child policy in China, we found that most QQ users do not mark multi children in their ego social networks. Parents (father and mother) may coexist in a child's QQ friend list and the child may have remarked them in his/her profile panel. In order to eliminate the interference between each other, we do not select the users who declare two children or parents in his/her QQ friend list. In our dataset, people usually use "father", "mother", "son" and "daughter" in Chinese instead of real name to remark their parents or children.

3. Inferring parent–child links by node centrality statistics

3.1. Review of node centrality statistics

In graph theory and network analysis, centrality statistics are used to identify the most important node within a graph (network). In most cases, a centrality that is the most optimal for one application may be sub-optimal for another different application, so researchers on social networks have proposed many different centrality statistics [14]. The primary and simplest statistic is degree centrality, which is defined as the number of ties that a node has [15]. The degree can be interpreted as the immediate risk of a node for catching a virus spreading through the network. In connected graphs there is a natural distance metric between all pairs of nodes, defined by the length of their shortest paths. The farness of a node is defined as the sum of its distances from all other nodes, and its closeness is defined as the reciprocal of the farness. Thus, if a node possesses the lower total distance from all other nodes, it will have the higher closeness centrality [16].

Table 2
Experimental performances of node centrality statistics.

Statistic	pa–chi	chi–pa	all	top3
Embeddedness	**0.170**	0.034	0.078	0.091
Betweenness	0.054	0.009	0.024	0.036
Closeness	0.116	0.019	0.051	0.074
Subgraph	**0.170**	0.035	0.080	0.117
Dispersion ($T = 3$)	0.152	**0.087**	**0.108**	**0.145**

Betweenness centrality is to quantify the number of times that a node acts as a bridge along the shortest path between two other nodes. It was introduced as a measure for quantifying the control of a human on the communication between other humans in a social network by Freeman [17]. In his conception, vertexes that have a high probability to occur on the shortest path between two randomly chosen vertexes possess a high betweenness [18]. Besides the above three frequently-used centrality statistics (degree, closeness and betweenness), there are more various types of centralities, such as eigenvector centrality [19], Katz centrality [20], load centrality [21], subgraph centrality [22,23]. Because all the above centrality statistics have been realized in NetworkX (a Python library for studying graphs and networks) [24], it is easy for us to use them to rank nodes in our datasets.

In [6], the authors used two other centrality statistics to infer romantic relationships: embeddedness and dispersion. Embeddedness was proposed in [25] to quantify the relative topological overlap of the neighborhood of two users i and j, representing the proportion of their common friends as

$$emb(i, j) = \frac{n_{ij}}{(k_i - 1) + (k_j - 1) - n_{ij}}. \tag{1}$$

In the equation n_{ij} is the number of common neighbors of i and j, and $k_i(k_j)$ denotes the degree of node $i(j)$. Therefore, $emb(i, j)$ can measure the importance of node j in the ego-network of node i. If i and j have no common acquaintances, then we have $emb(i, j) = 0$. In this case, the link between i and j represents the potential bridge between the two different communities. If i and j share the same circle of friends, then $emb(i, j) = 1$. In most cases, the value of $emb(i, j)$ is in the range of $(0, 1)$.

In the ego network of a center user i, the degree centrality of its friend j can be regarded as n_{ij} and it has no denominator of embeddedness as a normalization parameter, that is

$$C_{degree}(i, j) = n_{ij}. \tag{2}$$

Because embeddedness and degree centrality usually get the same rank order for every nodes in our dataset, so we do not distinguish the two statistics and only use the embeddedness centrality as a representative. Using other datasets or addressing other problems, the results of the two centralities might be very different.

In [6], the authors proposed a sequence of definitions that capture the idea of dispersion. To begin, they take the subgraph G_i of i and all neighbors of i (See Fig. 5(a)). For a node j in G_i, they define C_{ij} to be the set of common neighbors of i and j, and n_{ij} is the number of common neighbors of i and j. To express the idea that pairs of nodes in C_{ij} should be far apart in G_i, they do not consider the T-step paths through i and j themselves. Therefore, they define the absolute dispersion of the i–j link, to be the sum of all pairwise distances between nodes in C_{ij}, that is

$$disp(i, j) = \sum_{s,t \in C_{ij}} \theta(d_j(s, t) - T), \tag{3}$$

where $d_j(s, t)$ is the distance of node s and t in the set of $G_i - \{i, j\}$. If $d_j(s, t)$ is greater than or equal to T, $\theta(d_j(s, t) - T) = 1$; in other conditions, $\theta(d_j(s, t) - T) = 0$.

Here T is a significant distance threshold for calculating the value of dispersion, so the different choices of T will give rise to different measures of absolute dispersion. If we select $T = 1$, the value of $disp(i, j)$ is proportional to the degree of node j in the neighborhood of node i. If we select $T = 2$, $disp(i, j)$ is proportional to the number of nodes which are not directly linked in $G_i - \{i, j\}$. If we select $T = 3$, $disp(i, j)$ is equal to the number of pair nodes whose shortest path length is greater than 2 in G_i. The results in our QQ dataset show the best performance for $T = 3$, so we only show the result of dispersion centrality in the situation of $T = 3$ in the following paragraphs.

3.2. Experimental performances of node centrality statistics

In above section, we have introduced several definitions of node centrality algorithms. Implying parent–child links can be regarded as a task to find a special kind of important nodes, so all the node centrality methods can be used to detect parent–child links in each user's ego network. To simplify our table, here we only show the results of the following five methods. Embeddedness (degree) centrality, closeness centrality and betweenness centrality are the three classical centrality methods. Compared with other centrality statistics that we have introduced in Section 3.1, subgraph centrality can obtain the best performance, so we show the results of this centrality statistic and dispersion recently proposed in [6].

Table 3
The results of inferring romantic relationship and family member in Facebook.

Statistic	Romantic partner	Family member	Ratio
Degree	0.247	0.144	1.72
Dispersion	0.506	0.182	2.78
Photo	0.415	0.113	3.67
Profile view	0.301	0.088	3.42

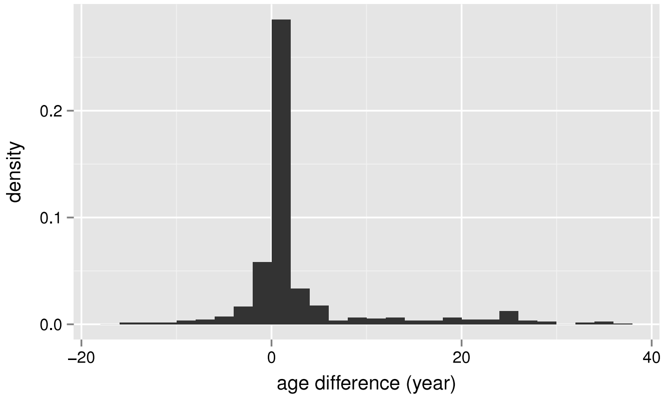

Fig. 1. The age difference between the center user who declares his/her parent is in the friend list and his/her largest embedding friend.

The experimental results for all these methods are shown in Table 2. Here we use the statistic of Accuracy to measure the performance of all the methods. The value of Accuracy is the ratio of the correct classification to the total sample size. For the cases of "pa–chi", "chi–pa" and "all", we only select one potential user as the inferred relationship for each center user. For the case of "top3", we select the three most probable users as the inferred relationship, so the performances of "top3" should be the highest in all the cases. The performances of all the experimental results are very low and the accuracy of all the results is in the range of [0.024, 0.108]. Especially, the performance for the centrality statistic of betweenness is only 2.4%, which is far below what we expect. Although the centrality statistic of dispersion can get the best performance, its accuracy of inferring parent–child links is only 10.8%. We would like to know whether parent–child links are difficult to identify or our dataset is not suitable for identifying such kind of relationship. Actually, the above result is consistent with the findings in [6]. We reuse the results of Figs. 4 and 5 in [6] to compare the results of inferring family members and romantic partners in Table 3. The result indicates that the accuracy of inferring romantic partner is far higher than that of inferring family members. Furthermore, the range of family members includes not only parents and children, but also grandparents, sisters, brothers, and so on. Therefore, inferring parent–child links in Facebook dataset is also more difficult than inferring romantic partners.

To find out why all the node centrality statistics cannot work well on inferring parent–child links, the age difference between the center user who declares his/her parent is in the friend list and his/her largest embedding friend is shown in Fig. 1. We can find that the age difference is almost very small and no more than 10 years, but we know the age difference between parent and child should be no less than 15 years. The results show that the most strong embedding ties of the young QQ users are their friends with the similar age instead of their parents. We also get the set of the most three strong embedding ties for every center user to verify whether the center user' parent or child is in the set. This result is shown in the last column of Table 2 ("top3"), and again the performance is not good enough.

All the results imply that most parent–child links are not strong ties in our QQ datasets, which is consistent with the conclusion from the mobile phone datasets in [8]. They found that the "best" (closest) friend of a young person often is his/her romantic partner for investment in reproduction. They also found that the "best" (closest) friends of parents show a higher probability to be their children. This result suggests us that implying parent-to-child may be easier than child-to-parent, which can be verified by comparing the results of "pa–chi" and "chi–pa" in Table 2 (from 1.75 times to 6.11 times). The age difference between the center user whose child is in the friend list and his/her largest embedding friend is shown in Fig. 2. For these parents, they show a higher probability to regard the children as their closest friends. In [8], Palchykov et al. also found this phenomenon and they explained it based on a parental care theory.

3.3. *Filtering out potential parent–child links by age and gender information*

Dispersion centrality can get a high performance for inferring romantic relationships in [6] because most romantic partners have a small age difference and their social circles have a strong overlap. However, there is a generation gap between

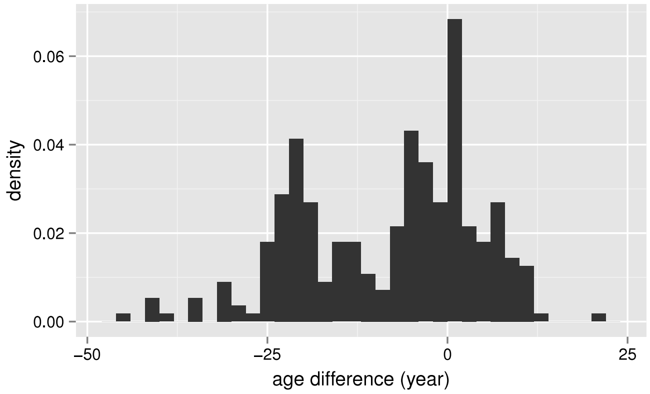

Fig. 2. The age difference between the center user who declares his/her child is in the friend list and his/her largest embedding friend.

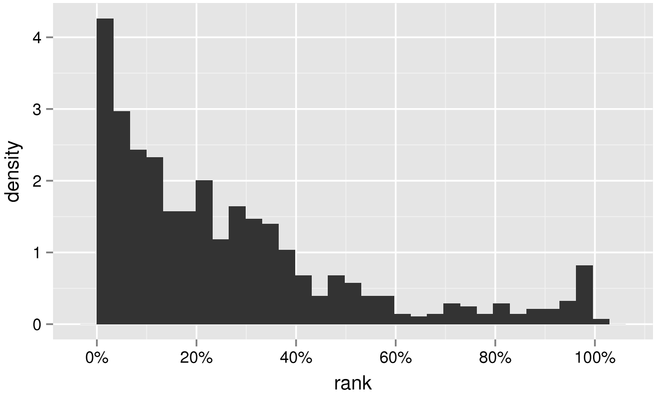

Fig. 3. The rank of the embeddedness centrality for a parent (or child) in the friend list. The vertical axis is the percentage of the density.

Table 4
Dataset basic description after considering age and gender information. N represents the number of each kind of users. M_1 represents the total number of potential links for each kind of users. R_1 represents the value of M_1/N, which is used to measure the difficulty of picking out the specific link in all the potential links.

Statistic	pa–chi	chi–pa	all
N	277	565	842
M_1	6920	7934	14 854
R_1	25	14	18

parents and children, which leads to weak embeddedness in their online social lives. We conjecture that is the reason we cannot take centrality statistics to infer parent–child links effectively. A very easy and effective method is to utilize users' age and gender information to filter out the set of potential parent–child links and then take centrality statistics to infer parent–child links again.

Before identifying parent–child links, we set a safe age gap between parent and child: the lower boundary is 15 years and the higher boundary is 45 years. Before having a child, men and women both need to reach sexual maturity, so our setting 15 years as the sexual mature age is reasonable. Except the most remote rural region, the single-child policy has been implemented well in China. Most QQ users live in a city or a town, so the parents will not bear children when they are old enough for over 45 years old. If we attempt to imply a father–son relationship, the female friends of the center user will not be considered, for they cannot be the son of any person. After employing the users' age and gender information as a filter, we can get a set of potential parent–child links and the set is far less than the number of all the user's friends (see Table 4).

It should be emphasized that QQ is an anonymous personal instant messaging software, so a significant proportion of users may fill the false age and gender information on their personal profiles. Before we got this QQ dataset, the researchers of Tencent company had corrected possible false age and gender information based on their other credible social network

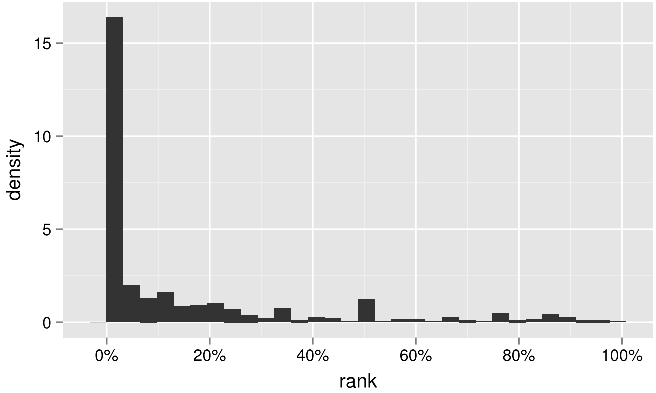

Fig. 4. The rank of the embeddedness centrality for a parent (or child) in the potential links. The vertical axis is the percentage of the density.

Table 5
Experimental performances of node centrality statistics after filtering out the potential users by considering age and gender information.

Centrality	pa–chi	chi–pa	all	top3
Embeddedness	0.459	0.464	0.463	0.597
Betweenness	0.264	0.315	0.298	0.375
Closeness	0.439	0.464	0.455	0.631
Subgraph	**0.489**	0.490	0.489	0.663
Dispersion ($T = 3$)	0.444	**0.531**	**0.503**	**0.721**

services (such as WeChat) and their internal correction algorithm. Although they cannot guarantee all the information is correct, in this study we regard the information they provided as the accurate information.

We also compared the ranks of a parent (or child) based on embeddedness centrality in the friend list (Fig. 3) and in the potential links (Fig. 4). Although parent–child links are often not the most strong embedding links in all the friends, they show a far higher probability to be the most strong embedding links in the potential links. We use multi node centrality statistics to infer parent–child links after filtering other links using age and gender information, and the results are shown in Table 5. We find that the results show far better performance than those in Table 3, especially dispersion centrality can get the best performance about 50%.

3.4. Combining embeddedness centrality and dispersion centrality

Embeddedness centrality shows a low performance for identifying romantic partners in [6], and the performance of dispersion centrality can be twice more than that of embeddedness (50.6% vs. 24.7%). However, embeddedness centrality shows an acceptable result for inferring parent–child links and its performance is only a little lower than that of dispersion (46.3% vs. 50.3%). Moreover, the performance of betweenness is far better than that of embeddedness for romantic partners (44.1% vs. 24.7%), while its performance is far less than that of embeddedness for inferring parent–child relationships (29.8% vs. 46.3%). The above results suggest that different node centralities can uncover distinct topology characters for network structures.

The authors of [6] discussed the difference between embeddedness and dispersion ($T = 3$) based on the theory of social foci, and explained why dispersion can better imply a person's romantic partner and family members. Basically, co-workers or classmates, which are different from family members, tend to form a large cluster and many people within the same social group know each other, which lead to have a high embeddedness. On the contrary, our romantic partners may have lower embeddedness, while they often know some of our friends belonging to different foci, and even the friends do not know each other. Because dispersion can measure whether the mutual friends of two users are well connected to one another or have a long shortest path length (belong to different foci), it shows the better performance than embeddedness to imply romantic partners in Facebook [6].

Considering node i and j in an ego social network, we use $disp(i, j)$ to represent the dispersion value between i and j, and use $emb(i, j)$ to denote the embeddedness of the i–j link. The performance monotonically increases in $disp(i, j)$ and monotonically decreases in $emb(i, j)$ for inferring romantic relationships in [6]. And the authors in [6] proposed a simple combination of the two quantities by the expression, that is

$$norm(u, v) = disp(i, j)/emb(i, j). \tag{4}$$

Table 6
Performance comparison for the embeddedness, dispersion and their combination methods.

Method	pa–chi	chi–pa	all	top3
Embeddedness	0.459	0.464	0.463	0.597
Dispersion	0.444	0.531	0.503	0.721
Dispersion/embeddedness	0.476	0.499	0.491	0.687
Embeddedness+semi-local (dispersion)	**0.578**	**0.563**	**0.568**	**0.788**

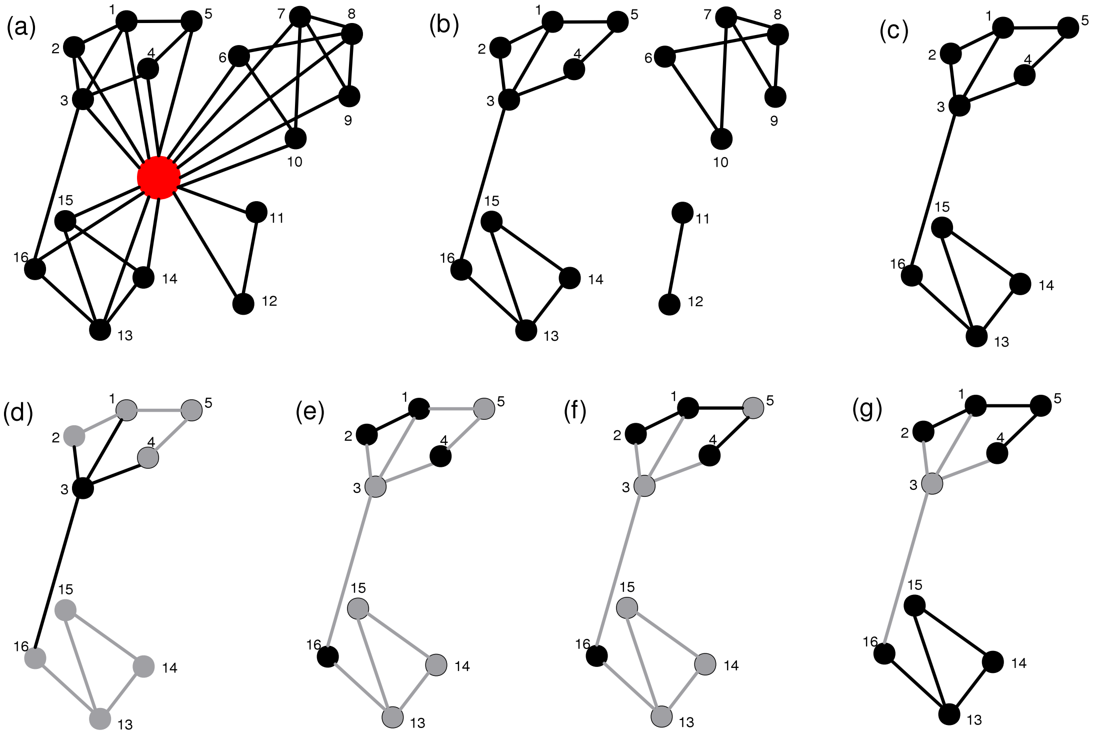

Fig. 5. The proposed framework to detect parent–child links. (a) the whole ego social network structure of a demonstrate user, (b) the network structure after removing the center user, (c) the network structure after removing other un-connected components, (d) calculating the direct effect by the node degree centrality, (e) calculating the indirect effect by the local structure, (f) calculating the indirect effect by the semi-local structure, and (g) calculating the indirect effect by the global structure.

They termed $norm(u, v)$ as the normalized dispersion because it normalizes the absolute dispersion by the embeddedness. The statistic of $norm(u, v)$ shows a better performance for identifying spouses and romantic partners on Facebook. Therefore, we also use it to infer parent–child links in QQ social network and the result is shown in Table 6. Although the result of $norm(u, v)$ is better than that of embeddedness, it is not as good as dispersion centrality. This result indicates that romantic relationships and parent–child links are different types of links, and they play different roles in social networks.

Here we try to emphasize the difference of parent–child links and romantic relationships. Romantic relationships are one type of strong embeddedness links, although they are not embedding enough to be always the most strong embeddedness in the user's ego social network. The most strong embeddedness links may be either romantic partners or co-workers (classmates for a student), so dispersion is a useful statistic to pick out the romantic partners. In this case, the normalized dispersion can get a better performance for it normalizes the absolute dispersion by the embeddedness.

However, parent–child links can be regarded as a kind of weak embeddedness, because the social circles are not strongly overlapped induced by the huge age difference between parents and children. Therefore, node centrality statistics are not enough to infer such kind of specific links, and it is required to build a preliminary filter system based on demographic variables (i.e., age and gender). Among all the pairs of users who have an age gap of 15–40 years, parent–child links are the most strong embeddedness relationships. Dispersion is another view to consider the importance of these links, so the linear composition for two kinds of statistics (embeddedness and dispersion) may achieve a better performance. We define $comp(i, j)$ as another simple combination of $emb(i, j)$ and $disp(i, j)$, that is

$$comp(i, j) = disp(i, j) + \lambda \cdot emb(i, j). \tag{5}$$

In this study, we only try to know which can better infer parent–child links for Eqs. (4) or (5), and we do not particularly concern how to get the optimal λ, so we select $\lambda = 1$ in Eq. (5). We find that $comp(i, j)$ can get the best performance in Table 6. It obtains a 11.3% higher performance than the method based on dispersion (56.8% vs. 50.3%). Furthermore, the computational complexity of $comp(i, j)$ is far lower than betweenness and closeness centralities.

3.5. A node-remove centrality framework beyond dispersion

Although the linear composition for embeddedness and dispersion can achieve a high precision for identifying parent–child links in Table 6, it is difficult to explain the essential mechanism behind this method. We try to build a node-remove framework to measure node centrality [26]. The node-remove framework defines the centrality value of a node is the total loss after removing this node from the network, and the total loss includes two parts: direct loss and indirect loss, that is

$$L_{total}(i, j) = L_{direct}(i, j) + L_{indirect}(i, j). \tag{6}$$

If we want to know the centrality statistic of a specific node, we remove the node and all its links from the center user's ego network to consider the total loss of the network. The direct loss is the degree of the specific node in the ego network for the center node. That is to say, if we remove the specific node, first we will lose all the links of this node. Therefore, the direct loss is the number of common friends between the specific node and the center node, which is

$$L_{direct}(i, j) = emb(i, j). \tag{7}$$

We demonstrate this process in Fig. 5(a)–(d). Firstly we get the whole ego social network structure of a demonstrate user, which is shown in Fig. 5(a). Then we remove the center user, because it connects to all the other nodes. After this step, we get the network structure in Fig. 5(b). Here we want to obtain the total loss after removing node 3. Because removing node 3 has not any effect on the two subgraphs that have no connection with node 3 on the right side of Fig. 5(b), we do not show the two un-connected components with node 3 any more in Fig. 5(c). It is easy to get the degree (embeddedness) of node 3 in Fig. 5(d), which is its direct loss ($L_{direct}(i, 3) = 3$).

After knowing the direct loss, the next step is to measure the indirect loss. Here we propose three methods to measure the indirect loss, which are based on the local, semi-local (dispersion) and global network topologies respectively. Firstly we get the common neighbors of a specific node (node 3) and the center user, that is, the set of nodes $C_{i3} = \{1, 2, 4, 16\}$. For the local method, we only calculate the indirect loss for the nodes in this set and only use the links among the nodes in this set. Then we calculate the shortest distance between any pair of nodes in C_{i3} by only using the links among the nodes in this set. For example, nodes 1 and 2 both belong to C_{i3}, so we can use the link between them and $distance(1, 2) = 1$ in Fig. 5(e). However, node 5 does not belong to C_{i3}, so we cannot use the links 1–5 and 4–5 to calculate the shortest distance between nodes 1 and 4 (i.e., $distance(1, 4) = \infty$). Before removing node 3, the shortest distance between any pair of nodes in C_{i3} should not be greater than 2, for node 3 is the common neighbor of them. Therefore, we calculate the indirect loss by

$$L_{indirect}(i, j) = disp(i, j) = \sum_{s,t \in C_{ij}} \theta(d_j(s, t) - 3), \tag{8}$$

where C_{ij} is the set of common neighbors of i and j, and $d_j(s, t)$ is the distance of node s and t in the set of C_{ij}. If $d_j(s, t)$ is greater than or equal to 3, $\theta(d_j(s, t) - 3) = 1$; in other conditions $\theta(d_j(s, t) - 3) = 0$. For the semi-local method, we calculate the indirect loss for the nodes C_{ij} while we can use the links not belonging to the nodes in this set $G_i - \{i, j\}$. To calculate the shortest distance between 1 and 4, although node 5 does not belong to C_{ij}, we can use the links 1–5 and 4–5 to calculate the shortest distance to get $distance(1, 4) = 2$ in Fig. 5(f). Comparing the calculation process and the definition of dispersion in Eq. (3), we can find they are the same. For the semi-local method, we can get that

$$L_{total}(i, j) = L_{direct}(i, j) + L_{indirect}(i, j) = emb(i, j) + disp(i, j) = comp(i, j). \tag{9}$$

For the global method, we calculate the indirect loss for all the nodes in $G_i - \{i, j\}$ instead of the nodes in C_{ij} and we can use all the links in Fig. 5(g). The performance comparison for the local, semi-local and global methods is shown in Table 7. Even taking the broader network structure into account, the semi-local method (embeddedness+dispersion) acquires the best performance. The above result indicate that parent–child links have semi-local effect instead of local or global effect in user's ego social network. Especially, the phenomenon of the semi-local effect is curious to explain that the formation of dispersion ($T = 3$) obtains the best performance than a shorter and longer distances. Our results suggest that a relative strong embeddedness instead of an absolute strong one exists between parents and children (the strongest tie in all the pairs of users who have an age gap of 15–40 years), because their social circles are not strongly overlapped induced by the parent–child huge age difference. Especially, the ratio of children regarding their parents as the most strong embedding ties are less than that parents regarding their children as the most strong embedding ties. The proposed node-remove framework uncover the mechanism why the linear composition for embeddedness and dispersion can achieve a high precision for identifying parent–child links.

Table 7
Performance comparison for the local, semi-local and global methods.

Method	pa–chi	chi–pa	all	top3
Embeddedness+local	0.559	0.541	0.547	0.742
Embeddedness+semi-local (dispersion)	**0.578**	**0.563**	**0.568**	**0.788**
Embeddedness+global	0.548	0.540	0.543	0.735

4. Conclusion

In this study, we explore to infer a special kind of family links (i.e., parent–child links) in the QQ social network based on multiple node centrality algorithms. We found that most parent–child links are weak social embeddedness instead of strong embeddedness, so we utilized users' age and gender information to filter out the set of potential links and then combined network structure mining. This result indicates that romantic relationships and parent–child links are different types of links, and they play different roles in social networks. We proposed a novel node-remove framework for measuring node centrality considering its direct and indirect effect on network topologies, and our method obtains a 11.3% higher performance than "dispersion" which is a new network centrality proposed in a recent study. Moreover, our node-remove framework also uncovered why the linear composition of embeddedness and dispersion is an efficient method for identifying parent–child links.

In this study, we only focus to discuss how the network structure affects the result, and ignore another important influential factor: the strength of social ties. Traditionally, the strength of social ties is regarded as the volume of interaction between two people. In online social networks, the strength of social ties shows different representation forms, such as viewing profiles, sending private messages, co-presence at events, and leaving a message on user's wall [27,28]. To define the interaction strength of QQ users and extract necessary data is an unfinished work. In future, we will try to combine network structure and interaction strength to improve the accuracy of predicting parent–child links.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61773091, 61374170, 61603073), the Open Project Program of Guizhou Provincial Key Laboratory of Public Big Data, China (2017BDKFJJ001), the CCF-Tencent Open Research Fund, China (AGR20130112), and the Dalian Youth Technology Star Project, China (2015R091). We thank Xiao-Fan Liu, Cheng-Jun Wang, Ya-Ding Yue, Chuan Chen, and Peng He for valuable discussions.

References

[1] M. McPherson, L. Smith-Lovin, J.M. Cook, Birds of a feather: Homophily in social networks, Annu. Rev. Sociol. 27 (2001) 415–444.
[2] H. Yin, Z. Rong, G. Yan, Development of friendship network among young scientists in an international summer school, Physica A 388 (17) (2009) 3636–3642.
[3] M. Burke, L.A. Adamic, K. Marciniak, Families on facebook, in: Proceedings of the 7th International Conference on Weblogs and Social Media, 2013, pp. 41–50.
[4] Y. Lan, M. Zhang, F. Zhu, J. Jiang, E.P. Lim, When a friend online is more than a friend in life: Intimate relationship prediction in microblogs, in: Asia-Pacific Web Conference, 2016, pp. 196–207.
[5] M. Madden, S. Cortesi, U. Gasser, A. Lenhart, M. Duggan, Parents, teens, and online privacy. pew internet and american life project, 2012. URL http://www.pewinternet.org/Reports/2012/Teens-and-Privacy.aspx.
[6] L. Backstrom, J. Kleinberg, (2014) Romantic partnerships and the dispersion of social ties: a network analysis of relationship status on facebook, in: Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing, 2014, pp. 831–841.
[7] Y. Wu, N. Pitipornvivat, J. Zhao, S. Yang, G. Huang, H. Qu, egoSlider: Visual analysis of egocentric network evolution, IEEE Trans. Visual. Comput. Graph. 22 (1) (2015) 260–269.
[8] V. Palchykov, K. Kaski, J. Kertesz, A.-L. Barabasi, R.I.M. Dunbar, Sex differences in intimate relationships, Sci. Rep. 2 (2012) 370.
[9] S. Sintos, P. Tsaparas, Using strong triadic closure to characterize ties in social networks, in: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014, pp. 1466–1475.
[10] B. Brown, B. Brown, Embeddedness and sequentiality in social media, in: ACM Conference on Computer-Supported Cooperative Work & Social Computing, 2016, pp. 1052–1064.
[11] M. Jaber, P.T. Wood, P. Papapetrou, S. Helmer, Inferring offline hierarchical ties from online social networks, in: Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion, 2014, pp. 1261–1266.
[12] X.-K. Xu, J.J.H. Zhu, Flexible sampling large-scale social networks by self-adjustable random walk, Physica A 463 (2016) 356–365.
[13] J.S. Vitter, Random sampling with a reservoir, ACM Trans. Math. Software 11 (1) (1985) 37–57.
[14] L.C. Freeman, Centrality in social networks conceptual clarification, Social Networks 1 (3) (1978) 215–239.
[15] A. Bavelas, Communication patterns in task-oriented groups, J. Acoust. Soc. Am. 22 (6) (1950) 725–730.
[16] G. Sabidussi, The centrality index of a graph, Psychometrika 31 (4) (1966) 581–603.
[17] L.C. Freeman, A set of measures of centrality based on betweenness, Sociometry 40 (1) (1977) 35–41.
[18] U. Brandes, A faster algorithm for betweenness centrality, J. Math. Sociol. 25 (2) (2001) 163–177.
[19] M. Newman, Networks: An Introduction, Oxford University Press, Inc., 2010.
[20] L. Katz, A new status index derived from sociometric analysis, Psychometrika 18 (1) (1953) 39–43.
[21] K.-I. Goh, B. Kahng, D. Kim, Universal behavior of load distribution in scale-free networks, Phys. Rev. Lett. 87 (2001) 278701.
[22] J.A. Ernesto Estrada, Subgraph centrality in complex networks, Phys. Rev. E 71 (5) (2005) 12240304.
[23] E. Estrada, N. Hatano, Communicability in complex networks, Phys. Rev. E 77 (3) (2008) 19111731.

[24] A.A. Hagberg, D.A. Schult, P.J. Swart, Exploring network structure, dynamics, and function using NetworkX, in: Proceedings of the 7th Python in Science Conference, SciPy2008, 2008, pp. 11–15.
[25] J.-P. Onnela, J. Saramaki, J. Hyvonen, G. Szabo, D. Lazer, K. Kaski, J. Kertesz, A.-L. Barabasi, Structure and tie strengths in mobile communication networks, Proc. Natl. Acad. Sci. 104 (18) (2007) 7332–7336.
[26] Y. Wang, J. Xu, Y. Xi, The core and coritivity of a system (IV) Relations between a system and its complement, J. Syst. Eng. Electron. 4 (2) (2012) 28–34.
[27] Q. Guo, F. Shao, Z.L. Hu, J.G. Liu, Statistical properties of the personal social network in the Facebook, Europhys. Lett. 104 (2) (2013) 28004.
[28] J.G. Liu, X.L. Liu, Q. Guo, J.T. Han, Identifying the perceptive users for online social systems, PLoS One 12 (7) (2017) e0178118.