计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2015.20148136
52(11):2535-2544,2015

社会网络上支持任务分组的团队形成方法

孙焕良[1] 金洺宇[1] 刘俊岭[1,2] 于 戈[2]
[1](沈阳建筑大学信息与控制工程学院 沈阳 110168)
[2](东北大学信息科学与工程学院 沈阳 110004)
(sunhl@sjzu.edu.cn)

Methods for Team Formation Problem with Grouping Task in Social Networks

Sun Huanliang[1], Jin Mingyu[1], Liu Junling[1,2], and Yu Ge[2]
[1](*Information and Control Engineering Faculty*, *Shenyang Jianzhu University*, *Shenyang* 110168)
[2](*College of Information Science and Engineering*, *Northeastern University*, *Shenyang* 110004)

Abstract Team formation problem in social network is gaining prominence in the research field of social network analysis and data mining. Previous study about team formation aimed at finding a team with the lowest communication cost. Some practical applications, such as large-scale software development and large-scale scientific research teams, usually need to divide a task. Based on this requirement, this paper presents a problem named grouping supported team formation in social network, which finds a team of experts to satisfy a complex grouping task and minimize the communication cost. The input of this problem is not a set of keywords of the traditional team formation problem, but a grouping task graph. Meanwhile, we also prove that this problem is NP-hard. Based on team communication models in organizational behavior, we define communication cost criterions for measuring grouping task teams, and propose multiple corresponding greedy searching strategies. The experimental results on real datasets demonstrate that different search strategies are suitable for different communication cost criterions and prove the effectiveness of the proposed algorithm.

Key words social networks; team formation; communication networks; greedy strategy; grouping task

摘 要 社会网络的团队形成问题已经逐渐成为社会网络分析以及数据挖掘领域的研究热点,现有团队形成问题的目标集中在查询一个成员间沟通代价最小的团队.在实际应用中,对于大规模任务通常需要按照模块进行任务划分,例如大型软件开发、大型科研项目等,因此完成任务的团队也需要进行分组.基于此需求,提出了社会网络上支持任务分组的团队形成问题,即从专家社会网络中查询出满足复杂任务分组且沟通代价最小的专家团队.该问题的查询输入不再是传统团队形成问题中的技能集合,而是输入一个分组任务图,证明了该问题是NP难问题.依据组织行为学中的团队沟通模型,定义了任务分组的团队沟通代价度量,并提出了基于不同贪心搜索策略的算法.采用真实数据集对所提出的算法进行了实验评估,实验结果表明依据不同的贪心策略实现的算法能够适用于不同的沟通代价度量方法,证明了算法的有效性.

关键词 社会网络;团队形成;沟通网络;贪心策略;分组任务

中图法分类号 TP311.1

收稿日期:2014-10-17;**修回日期**:2015-01-20

基金项目:国家自然科学基金项目(61070024,61272179);教育部高等学校博士学科点专项科研基金项目(20120042110028);教育部-英特尔信息技术专项科研基金项目(MOE-INTEL-2012-06)

社会网络的团队形成问题已经逐渐成为社会网络分析以及数据挖掘领域的研究热点.传统的团队形成问题只需要找出能够提供所需技能的成员集合,不必考虑成员间的沟通代价.然而,实际应用需求中团队的绩效不仅取决于成员的专业技能,而且要求团队成员间能够进行高效的沟通.随着社交平台的兴起,大量用户通过互加好友或相互关注等方式,形成了具有大规模结点的社会网络[1-3].通过社会网络可以有效度量团队成员间的沟通代价[4],从而使基于社会网络的团队形成问题得到广泛关注[5-6].

给定一个技能需求的集合(称为任务),社会网络上的团队形成问题就是要找到一个能够满足任务要求并且沟通代价最小的团队.现有的社会网络上的团队形成问题,均针对单个技能集合的任务需求[2-4,6-12].然而在实际应用中,存在较多大规模复杂任务,而一个管理者能够有效直接领导、指挥和监督的下属人数的上限一般为 12 人[13].因此,需要对大规模任务的团队按照任务模块进行分组,以保证团队能够高效地运行管理.

基于上述需求,我们提出了支持任务分组的团队形成问题.给定一个专家社会网络和一个包含若干子任务的任务,其中每个子任务均包含一组所需技能,目标是找到一个能够完成项目中所有子任务的专家团队,团队中的每个专家分组能够分别完成相应的子任务,并且团队的整体沟通代价最小.

据我们所知,现有的团队形成问题研究认为团队是一个关系紧密的整体[2-4,6-12],每 2 个成员之间都要进行沟通,然而在实际应用场景中存在任务分组的需求,分组后成员间不必全部沟通.如图 1 所示,项目经理要组织一个团队开发某软件项目,图 1 中任务分为 t_1,t_2,t_3,t_4 这 4 个子任务,每个子任务都有一个所需技能的集合.为了完成此任务,需要分别为它们找到一组符合技能需求的人员,同时各子任务具有相对的独立性,因此负责开发不同模块的技术人员通常不必进行沟通,只需要模块间组长沟通即可.

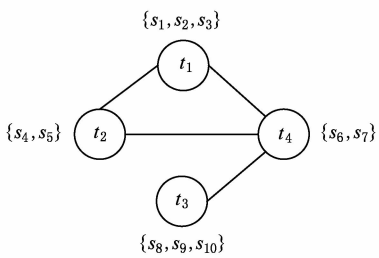

Fig. 1 Grouping task graph.

图 1 分组任务图

支持任务分组的团队形成问题需要同时考虑专家小组内部和小组之间的沟通代价,因此,该问题的挑战在于:1)如何平衡专家小组内部与小组之间的沟通代价;2)如何结合组内与组间代价,设计出能够从专家社会网络中查询出有效专家团队的算法.

为解决上述问题,依据组织行为学中的团队沟通模型,定义了任务分组的团队沟通代价度量,并且提出了基于不同贪心搜索策略的算法.

本文的主要贡献如下:

1) 根据实际应用需求,提出了社会网络上支持任务分组的团队形成问题,定义了任务分组的团队沟通代价度量;

2) 基于不同的沟通模型,提出了相应的搜索策略,并且设计了相应的算法;

3) 使用真实数据集对所提出的算法进行了充分实验,验证了算法的有效性.

1 相关工作

传统的团队形成问题需要找到专家与所需技能之间的最优匹配,例如运筹学中的团队形成[9,14],解决此类问题的方法有模拟退火法[14]、分支限界法[10]和遗传算法[11]等.

随着社会网络的发展,一些学者研究基于社会网络的团队形成问题[2-4,6].这些研究将专家之间的社会关系抽象为沟通代价,用于对团队进行评价,研究不同的沟通代价度量方法下的搜索算法.现有的团队沟通代价的度量方法包括直径沟通代价[6]、最小生成树沟通代价[6]、距离之和[2]、领导者距离[1]和瓶颈代价[7]等.这些代价模型满足不同的应用需求,如直径沟通代价和瓶颈沟通代价的度量方法避免了团队内存在过大的沟通代价;距离之和、领导者距离的度量方法可以减小团队的总沟通代价;最小生成树则假设团队中每个人与最容易沟通的成员沟通即可.现有的团队搜索策略只需要考虑搜索一个任务需求的应用,而不必考虑对团队进行分组,因此现有方法难以直接应用于求解本文所提出的支持任务分组的团队形成问题.

另一类社会网络中团队形成问题为基于约束条件的团队形成问题,这类问题针对特定应用场景、适应不同约束条件的团队形成.文献[3,8]提出了在多任务情况下使各专家的最大工作量最小化的问题.

文献[15]研究了在社会网络上某时间段内查询具有熟人约束的群体.文献[16-18]针对多目标任务的团队形成,分别提出了2种不同的解决方法:1)通过对不同的目标设定平衡系数[16],将多目标问题转化为单一目标问题;2)为多目标团队形成问题找出一组帕累托最优解[17-18].

与上述工作相比,本文提出了支持任务分组的团队形成问题,即在单任务团队形成问题的基础上将任务划分成多个子任务,并且指定各子任务之间的相关性,最终形成的专家团队由满足各子任务的专家小组构成.

2 问题定义

专家社会网络是一个无向带权图 $G=(V,E)$,其中 $V=\{v_1,v_2,\cdots,v_n\}$ 为结点集,$E\subseteq V\times V$ 是图 G 的边集.在本文中,结点集中的一个结点代表一名专家,边表示2个专家 v_i 与 v_j 具有合作关系,边上的权值表示专家 v_i 与 v_j 间沟通代价.结点间边权越小,表示专家之间沟通代价越小.$S=\{s_1,s_2,\cdots,s_m\}$ 表示 m 个技能的集合.每个专家 v_i 都拥有一组技能,表示为 $P(v_i)$,并且 $P(v_i)\subseteq S$.如果 $s_j\in P(v_i)$,表示专家 v_i 拥有技能 s_j.如果一个专家子集 $V'\subseteq V$ 拥有技能 s_j,则说明 V' 中至少有一个专家拥有技能 s_j.对于每个技能 s_j,称所有包含此技能的专家集合为技能 s_j 的候选集,表示为 $C(s_j)=\{v_i\mid v_i\in V,s_j\in P(v_i)\}$.一个任务 $t_i=\{s_1,s_2,\cdots,s_p\}$ 表示完成该任务所需要的技能集合.如果一个专家子集 $V'\subseteq V$ 满足 $\forall s_j\in t_i,\exists v_i\in V',s_j\in P(v_i)$,则称 V' 能够完成任务 t_i.

结点 v_i 和 v_j 之间的距离是它们在图 G 中最短路径上的边权之和,表示为 $d(v_i,v_j)$.我们定义一个结点 v_i 与一个结点子集 V' 的距离为 $d(v_i,V')=d(v_i,v_j)$,其中,$v_j\in V'$,$\forall v_k\in V'$ 都满足 $d(v_i,v_j)\leqslant d(v_i,v_k)$.如果 $V'=\varnothing$,则定义 $d(v_i,V')=\infty$.另外,定义2个结点子集 V'_a 和 V'_b 之间的距离为 $d(V'_a,V'_b)$,表示2个子集间结点的最短距离.

定义1. 分组任务图.$Q(T,R)$ 表示一个任务的分组模式,$T=\{t_1,t_2,\cdots,t_q\}$ 中的每个结点 t_i 表示任务 T 的一个子任务,是一组所需技能集合 $t_i\subseteq S$;R 是 Q 中的边集,R 中每条边 r_{ij} 表示任务 t_i 与 t_j 之间存在沟通关系.

定义2. 专家小组.给定一个专家集合 V 和一个任务 $t_i=\{s_1,s_2,\cdots,s_p\}$,能够完成任务 t_i 的专家小组 V'_i 是一个〈技能,专家〉二元组的集合,表示为 $\{\langle s_1,v_{s_1}\rangle,\langle s_2,v_{s_2}\rangle,\cdots,\langle s_p,v_{s_p}\rangle\}$.其中,$\langle s_i,v_{s_i}\rangle$ $(1\leqslant i\leqslant p)$ 表示在任务 t_i 中专家 v_{s_i} 提供 s_i 技能,假设一名专家仅属于一个小组且可以提供多个技能.

定义3. 分组专家团队.给定分组任务图 $Q(T,R)$,能够完成任务 T 的一个分组专家团队 V_r 是一个〈任务,专家小组〉二元组的集合 $\{\langle t_1,V'_1\rangle,\langle t_2,V'_2\rangle,\cdots,\langle t_q,V'_q\rangle\}$.如果任务 t_i 与 t_j 在分组任务图中有边相连,则团队 V_r 中对应的专家小组 V'_i 与 V'_j 之间存在沟通关系.

问题1. 支持任务分组的团队形成问题.给定一个专家社会网络 G 和一个分组任务图 Q,基于任务分组的团队形成问题需要找到一个能够完成任务 T 的分组专家团队 V_r,并且此分组专家团队的沟通代价 $V_r.Cost$ 最小.

当查询只有一个任务的专家团队时,问题1中的每个子任务即是文献[19]中提出的团队形成问题.文献[19]已证明了该问题为NP难问题,因此问题1也为NP难问题.

3 沟通代价模型

组织行为学中有3种最基本的组织沟通模型,分别为链式沟通模型、轮式沟通模型和全通道式沟通模型[12].其中,链式沟通模型指团队成员只能与一个其他成员进行沟通;轮式沟通模型指团队中有一个领导者,其他成员只能够与此领导者进行沟通;全通道式沟通模型是指团队中的每一个成员都能与其他任何人进行直接沟通.依据这3种基本的组织沟通模型,本文将定义3种专家小组的组内沟通代价计算方法和2种组间的沟通代价计算方法.

3.1 组内沟通代价

定义4. 链式沟通代价.给定一个任务 t_i 和一个能够完成任务 t_i 的专家小组 V'_i.若专家小组 V'_i 采用链式沟通模型进行沟通,则小组的链式沟通代价定义为小组在社会网络图 G 上形成的最小生成树的边权之和,表示为 $Cost\_MST_{\text{intra}}(V'_i)$.

定义5. 轮式沟通代价.给定一个任务 t_i 和一个能够完成任务 t_i 的专家小组 V'_i.若专家小组 V'_i 采用轮式沟通模型进行沟通,即每个专家小组中有一个组长 $V'_i.L$,则小组的轮式沟通代价如式(1)定义为

$$Cost\_STAR_{\text{intra}}(V'_i)=\sum_{i=1}^{q}d(v_{s_i},V'_i.L),\quad(1)$$

其中 v_{s_i} 表示在此任务中提供技能 s_i 的成员,小组的组长 $V'_i.L$ 规定为小组内使得 $Cost\_STAR_{\text{intra}}(V'_i)$ 的值最小的小组成员.我们认为在轮式沟通模型下,小组中每个组员在完成每个技能需求时都需要与组长进行沟通,所以提供所需技能越多的组员与组长之间的沟通越频繁.

定义 6. 全通道式沟通代价.给定一个任务 t_i 和一个能够完成任务 t_i 的专家小组 V'_i.若 V'_i 采用全通道式沟通模型进行沟通,则小组的沟通代价如式(2)定义为

$$Cost\_SUM_{\text{intra}}(V'_i)=\sum_{i=1}^{q}\sum_{j=i+1}^{q}d(v_{s_i},v_{s_j}). \quad (2)$$

与轮式沟通代价一样,我们认为在采用全通道式沟通模型时,小组中提供越多任务所需技能的组员在小组中与其他组员的沟通越重要.因此,将全通道式沟通代价定义为每两对不同技能提供者在社会网络图 G 上的距离之和.

3.2 组间沟通代价

给定一个分组任务图 $Q(T,R)$ 和一个能够完成任务 T 的专家团队 $V_r=\{\langle t_1,V'_1\rangle,\langle t_2,V'_2\rangle,\cdots,\langle t_q,V'_q\rangle\}$,如果小组 V'_i 和 V'_j 所对应的任务 t_i 和 t_j 在分组任务图 Q 中有边相连,则需要计算小组 V'_i 和 V'_j 间的沟通代价.若任务 t_i 和 t_j 在分组任务图 Q 中没有边相连,则他们的组间沟通代价为 0.

定义 7. 平衡系数.给定一个专家团队 $V_r=\{\langle t_1,V'_1\rangle,\langle t_2,V'_2\rangle,\cdots,\langle t_q,V'_q\rangle\}$,平衡系数定义为 $\rho_{i,j}=(|t_i|+|t_j|)/2$.平衡系数用于调整组间沟通代价与组内沟通代价之间的比例.

定义 8. 最近邻组间沟通代价.给定一个分组任务图 $Q(T,R)$ 和一个能够完成任务 T 的专家团队 $V_r=\{\langle t_1,V'_1\rangle,\langle t_2,V'_2\rangle,\cdots,\langle t_q,V'_q\rangle\}$.若团队中小组采用链式沟通模型或全通道式沟通模型时,定义该团队的最近邻组间沟通代价为

$$Cost\_NN_{\text{inter}}(V_r)=\sum_{i=1}^{q}\sum_{j=i+1}^{q}\rho_{i,j}d(V'_i,V'_j). \quad (3)$$

定义 9. 组长组间沟通代价.给定一个分组任务图 $Q(T,R)$ 和一个能够完成任务 T 的专家团队 $V_r=\{\langle t_1,V'_1\rangle,\langle t_2,V'_2\rangle,\cdots,\langle t_q,V'_q\rangle\}$.若团队中小组采用轮式沟通模型时,定义该团队的组长组间沟通代价为

$$Cost\_LL_{\text{inter}}(V_r)=\sum_{i=1}^{q}\sum_{j=i+1}^{q}\rho_{i,j}d(V'_i.L,V'_j.L). \quad (4)$$

3.3 团队沟通代价

给定一个分组任务图 $Q(T,R)$ 及一个能够完成任务 T 的专家团队 $V_r=\{\langle t_1,V'_1\rangle,\langle t_2,V'_2\rangle,\cdots,\langle t_q,V'_q\rangle\}$.采用以上组内和组间沟通代价,给出团队沟通代价定义.

若团队 V_r 中小组采用链式沟通模型,则 V_r 的沟通代价定义为

$$V_r.Cost\_MNN=\sum_{i=1}^{q}Cost\_MST_{\text{intra}}(V'_i)+Cost\_NN_{\text{inter}}(V_r). \quad (5)$$

若团队 V_r 中小组采用轮式沟通模型,则 V_r 的沟通代价定义为

$$V_r.Cost\_SLL=\sum_{i=1}^{q}Cost\_STAR_{\text{intra}}(V'_i)+Cost\_LL_{\text{inter}}(V_r). \quad (6)$$

若团队 V_r 中小组采用全通道式沟通模型,则 V_r 的沟通代价定义为

$$V_r.Cost\_SNN=\sum_{i=1}^{q}Cost\_SUM_{\text{intra}}(V'_i)+Cost\_NN_{\text{inter}}(V_r). \quad (7)$$

下面的表 1 总结了本文中定义的沟通代价.

Table 1 Communication Cost

表 1 沟通代价符号表示

Serial Number	Total-Communication Cost	Intra-Communication Cost	Inter-Communication Cost
1	$Cost\_MNN$	$Cost\_MST_{\text{intra}}$	$Cost\_NN_{\text{inter}}$
2	$Cost\_SLL$	$Cost\_STAR_{\text{intra}}$	$Cost\_LL_{\text{inter}}$
3	$Cost\_SNN$	$Cost\_SUM_{\text{intra}}$	$Cost\_NN_{\text{inter}}$

4 搜索策略

在本节中,首先给出了查询算法的总框架,然后提出了 4 种基于贪心策略设计的算法.

4.1 总体框架

因为问题 1 是一个 NP 难问题,本文采用近似算法搜索最优团队.算法总体框架如算法 1 所示,输入为专家社会网络图 $G(V,E)$ 和分组任务图 $Q(T,R)$.其中,对于每个技能 s_i,图 G 中所有拥有技能 s_i 的专家集合 $C(s_i)$ 可以通过预先建好的倒排索引得到.算法返回一个近似最优专家团队 $V_r\in V$.

因为需要支持分组任务,算法 1 中包括 2 个重要的函数 $FindGroup$ 和函数 $Select\_T$.其中,函数

FindGroup 通过不同的贪心策略搜索出一个能够完成某个子任务的专家小组;函数 *Select_T* 则是通过不同的贪心策略确定下一个将要进行专家搜索的子任务,并选出该专家小组的第 1 位专家.

算法 1. 总体框架.

输入:$G(V,E)$,$Q(T,R)$;

输出:V_r.

① 初始化变量 V_r,$V_r.Cost$,$init\_v$,T',V_{r0};

② 找到 T 中具有最少候选结点的技能 S_{rare};

③ $T'.Add$(需要技能 S_{rare} 的子任务 t);

④ for each $v\in C(S_{rare})$ do{

⑤ while $T'\neq\varnothing$ do{

⑥ $(current\_t,init\_v)\leftarrow Select\_T(V_{r0},T')$;

⑦ $T'.delete(current\_t)$;

⑧ $V'\leftarrow FindGroup(current\_t,init\_v)$;

⑨ $T'.Add$(所有与 $current\_t$ 相关的子任务);

⑩ $T'.delete(current\_t)$;

⑪ $V_{r0}.insert(V')$;

⑫ $V_{r0}.Cost\leftarrow V_{r0}.Cost+Cost\_MST_{intra}(V')$;}

⑬ $V_{r0}.Cost\leftarrow V_{r0}.Cost+Cost\_NN_{inter}(V_{r0})$;

⑭ if $V_r.Cost>V_{r0}.Cost$ then

⑮ 将 V_r,$V_r.Cost$ 替换为 V_{r0},$V_{r0}.Cost$;}

⑯ return V_r.

在 4.2~4.5 节中,将对函数 *Select_T* 和函数 *FindGroup* 进行介绍.提出 4 种搜索策略,包括基于链式沟通模型的搜索策略 MST(multi spanning tree)、基于轮式沟通模型的搜索策略 STAR、基于全通道式沟通模型的搜索策略 SUM 以及基于收益变量 *Profit* 的搜索策略 PRO,基于这些策略设计出适用于不同应用场景的算法.

在算法 1 中,$init\_v$ 表示一个专家小组的起始结点;T' 表示待查询的子任务集合;V_{r0} 表示每一次循环所搜索出的专家团队;二元组 $(current\_t,init\_v)$ 中,$current\_t$ 表示当前要进行查询的子任务,$init\_v$ 表示该子任务的起始结点.

4.2 MST 搜索策略

在 MST 搜索策略中,函数 *Select_T_MST* 每次确定下一个需要查询的子任务时,查询距当前结果中距离最近的结点,同时该结点应包含尚未满足需求的技能.然后,将当前结点作为下一个专家小组搜索的起始结点,判断该结点包含的需求技能可满足的子任务,将它作为下一个进行查询的子任务.函数 *Select_T_MST* 的具体描述如算法 2:

算法 2. *Select_T_MST*.

输入:V_{r0},T';

输出:$(current\_t,init\_v)$.

① $Dist\leftarrow\infty$,$Dist_{min}\leftarrow\infty$;

② for each $t\in T'$ do{

③ $related\ V'.insert$(V_{r0} 中所有与 t 相关任务的专家小组);

④ $candidateSet.Add$(t 中所有技能的候选结点);

⑤ for each $v\in candidateSet$ do{

⑥ $Dist\leftarrow d(v,related\ V')$;

⑦ if $Dist<Dist_{min}$ then

⑧ $Dist_{min}\leftarrow Dist$;

⑨ $(current\_t,init\_v)\leftarrow(t,v)$;}}

⑩ return$(current_t,init_v)$.

函数 *Select_T_MST* 中步骤②~⑨,每次循环从未找到专家小组的子任务集合 T' 中取出一个子任务 t;步骤⑤~⑨每次循环从候选结点集合 *candidateSet* 中取出一个结点 v;步骤⑥计算出结点 v 与 *related*V' 中所有结点之间的最短距离 $Dist$;步骤⑦~⑨比较 $Dist$ 与当前的最短距离 $Dist_{min}$,若 $Dist<Dist_{min}$ 则替换,并记录与已找到的相关联的专家小组具有最短距离的二元组 $(current\_t,init\_v)$,其中 $current\_t$ 记录具有最短距离的子任务,$init\_v$ 记录该子任务的起始结点;最后将这个二元组返回到调用函数中.

组内查找函数 *FindGroup_MST* 根据 Prim 算法[20]思想设计,具体描述见算法 3.算法 3 中步骤④在每次循环中,将与已找到的小组距离最短并且包含此任务需求技能的结点添加到该专家小组中,直到该专家小组满足此任务中所有的技能需求,最后返回得到的满足任务需求的专家小组 V'.

算法 3. *FindGroup_MST*.

输入:$G(V,E)$,$current\_t$,$init\_v$;

输出:V'.

① $V'.Add$($init\_v$ 包含的所需技能,$init\_v$);

② for each $s\in current\_t$ do{

③ $nearest_v\leftarrow$获取 $C(s)$ 中与 V' 距离最短结点 c;

④ $V'.Add(s,nearest\_v)$;}

⑤ return V'.

4.3 STAR 搜索策略

若专家团队中的专家小组采用轮式沟通模型,则每个专家小组 V' 中都有一个组长 $V'.L$,所以在

计算组内沟通代价时只需计算其他组员与组长间的距离,计算组间沟通代价时只计算2组专家的组长间的距离.因此,对于选择下一个需要查询的任务并进行专家小组查询时,只需考虑其他候选结点与当前组长间的距离即可.

与MST策略相比,本节提出的搜索策略将函数*Select_T_MST*和*FindGroup_MST*修改为函数*Select_T_STAR*和函数*FindGroup_STAR*,具体实现见算法4,5.

算法4. *Select_T_STAR*.

替换算法2中步骤⑥代码

① $Dist \leftarrow d(v, related V'$中的专家小组组长集合).

算法5. *FindGroup_STAR*.

替换算法3中步骤③代码

① $nearest_v \leftarrow$获取$C(s)$中与$V'.L$距离最短结点c.

在算法4和算法5中,由于在计算小组之间的沟通代价时,计算各小组长之间的最短路径.因此,在函数*Select_T_STAR*中,只需计算候选结点与已有的专家小组中组长之间的最短路径即可.同样,在函数*FindGroup_MST*中,只需计算候选结点与该小组的组长之间的最短路径即可.

4.4 SUM搜索策略

如果专家团队中的专家小组采用全通道式沟通模型,在计算组内沟通代价时,需要计算所有组员之间的距离.因此将函数*Select_T_MST*中步骤⑧进行替换得到*Select_T_SUM*,见算法6;将*FindGroup_MST*进行替换得到函数*FindGroup_SUM*,见算法7.

算法6. *Select_T_SUM*.

替换算法2中步骤⑤⑥代码

① for each $V' \in related\ V'$ do

② $\{Dist \leftarrow Dist + d(v, V').\}$

算法7. *FindGroup_SUM*.

替换算法3中步骤②~④代码

```
① for each s∈current_t do{
②   for each v_i∈C(s) do{
③     for each v_j∈V' do
④       Dist←Dist+d(v_i,v_j);
⑤     if Dist<Dist_min then
⑥       Dist_min←Dist;
⑦       nearest_v←v_i;}
⑧   V'.Add(s,nearest_v).}
```

在SUM搜索策略中,在选择候选结点时需要计算所有可能的连接路径长度之和,以此作为选取结点的依据,所以在算法6和算法7中,在计算最短路径*Dist*时通过迭代得到最短路径长度之和.算法7的3层嵌套循环中,最外层循环次数为当前任务中的关键字数目k_1,第2层循环最差为社会网络中全部专家数目n,最内层循环次数为当前团队中的专家数目k_2,3层循环的次数为$k_1 \times k_2 \times n$,其中k_1,k_2为常数,所以算法7的时间复杂度为$O(n)$.

定理1. 由算法6和算法7找到专家团队的沟通代价*Cost_SNN*可以满足2-近似.

证明. 假设有一个通过枚举法得到精确解的最优团队*OptTeam*和一个通过算法6、算法7得到近似解的团队*BestTeam*,分组数为q,各专家小组分别表示为$OptGroup_i$和$BestGroup_i$,其中$i \in q$,由文献[2]中的定理2可以知道算法7找到的每一个专家小组都满足2-近似,即

$$BestGroup_i.Cost\_SUM \leqslant 2 \times OptGroup_i.Cost\_SUM, \tag{8}$$

在此不再重复证明.对于算法6,由于输入的任务分组数q和各分组之间的关系数k是可变的,显然k值越大得到的近似解将越差,所以在此考虑一种最差的极端情况,每2组之间都需要进行沟通,即$k=q(q-1)/2$.此时若将各专家小组抽象为一个点,则可将算法6的组间搜索策略等同为组内SUM搜索策略,由此可得式(9):

$$BestTeam.Cost\_NN \leqslant 2 \times OptTeam.Cost\_NN. \tag{9}$$

由式(8)和式(9)可得:

$$BestTeam.Cost\_NN + \sum_{i=1}^{q} BestGroup_i.Cost\_SUM \leqslant 2 \times (OptTeam.Cost\_NN + \sum_{i=1}^{q} OptGroup_i.Cost\_SUM), \tag{10}$$

即$BestTeam.Cost \leqslant 2 \times OptTeam.Cost$,因此可证明由算法6和算法7得到近似解的团队沟通代价最多是精确解的团队沟通代价的2倍. 证毕.

4.5 PRO搜索策略

本文算法1~7都是采用贪心策略,每次找到一个社会网络上距离最小的结点作为局部最优解,但计算沟通代价时并未将结点间距离和结点包含需求技能的个数作为影响因素.因此,在选取局部最优解时,应同时将结点提供的技能数目作为一个局部最

优解的选择因素. PRO 搜索策略中函数 *Select_T_PRO* 实现的描述见算法 8:

算法 8. *Select_T_PRO*.

输入:V_{r0},T';

输出:(*current_t*,*init_v*).

① $dist\leftarrow\infty$,$Profit\leftarrow\infty$,$Profit_{\min}\leftarrow\infty$;

② for each $t\in T'$ do{

③ *related* V'.*insert*(V_{r0} 中所有与 t 相关的任务专家小组);

④ *candidateSet*.*Add*(t 中所有技能的候选结点);

⑤ for each $v\in candidateSet$ do{

⑥ for each $V'\in related\ V'$ do

⑦ $Dist\leftarrow Dist+d(v,V')$;

⑧ $Profit\leftarrow v$ 提供的技能个数/$Dist$;

⑨ if $Profit<Profit_{\min}$ then

⑩ $Profit_{\min}\leftarrow Profit$;

⑪ (*current_t*,*init_v*)$\leftarrow$(t,v);}}

⑫ return(*current_t*,*init_v*).

在算法 8 中步骤⑧定义的收益变量 *Profit* 为所选结点提供的需求技能的个数除以与它有沟通关系的专家小组距离之和,用于记录每次选择结点所获得的收益. 组间沟通代价与人数成正比,收益变量可以实现优先选择具有单位代价下技能数量大的结点,从而减少团队人数降低沟通代价,所以在 PRO 搜索策略中每次贪心选取的局部最优解就是收益最大的结点. 在函数 *FindGroup_PRO* 中同样采取这种贪心策略. 函数 *FindGroup_PRO* 具体的实现细节如算法 9:

算法 9. *FindGroup_PRO*.

输入:$G(V,E)$,*current_t*,*init_v*;

输出:V'.

① V'.*Add*(*init_v* 包含的所需技能,*init_v*);

② while *current_t*$\neq\varnothing$ do {

③ *candidateSet*.*Add*(*current_t* 中所有技能的候选结点);

④ for each $v\in candidateSet$ do{

⑤ $Dist\leftarrow d(v,V')$;

⑥ $Profit\leftarrow v$ 可提供的 *current_t* 中的技能个数/$Dist$;

⑦ if $Profit<Profit_{\min}$ then

⑧ $Profit_{\min}\leftarrow Profit$;

⑨ $nearest\_v\leftarrow v$;

⑩ for each $s\in current\_t$ 且 $s\in P(nearest\_v)$ do

⑪ V'.*Add*(s,*nearest_v*);

⑫ *Current_t*.*delete*(s);}}

⑬ return V'.

5 实验结果与分析

本节主要对提出的问题以及算法进行合理的实验设计,采用真实数据集进行实验,对比分析各算法的查询结果. 现有的团队形成算法中,都没有考虑到团队分组的情况,并未涉及组间沟通代价计算. 因此,现有方法难以实现分组情况下的团队沟通代价计算. 实验中对比了所提出的搜索算法的查询结果.

实验采用的是 DBLP 数据集,数据更新至 2013-06-18[21]. 采用参考文献[6]中的方法对数据集进行预处理后,得到的专家集共包括 6 703 个专家和 3 251 种不同的技能. 若专家 v_i 和专家 v_j 共同发表至少 2 篇文章,则在专家社会网络图上有边相连,利用此阈值保留下来的边共有 9 747 条. 专家 v_i 和专家 v_j 之间的边权 $\omega(v_i,v_j)=1-\frac{|p_{v_i}\cap p_{v_j}|}{|p_{v_i}\cup p_{v_j}|}$,其中 p_{v_i},p_{v_j} 分别为专家 v_i、专家 v_j 发表的论文集合,也就是说边权为图上所有相连结点对之间的 Jaccard 距离.

MST_MST_NF 表示为组内采用 MST 搜索策略,即函数 *FindGroup* 采用 *FindGroup_MST* 实现;组间也采用 MST 搜索策略,即函数 *Select_T* 采用 *Select_T_MST* 实现,而各函数中都只用结点最短路径对所选结点度量沟通代价. MST_SUM_Profit 为组内采用 MST 策略,函数 *Select_t* 采用 *Select_t_SUM* 实现,而各函数中使用收益变量 *Profit* 对所选结点度量沟通代价.

本节实验中,将查询输入设置为图 2 所描述的默认配置,即查询的任务共有 5 个子任务,各子任务的技能个数分别为 2,4,6,8,10,在任务总技能数目增加时,各子任务中的技能个数也按照这个默认的比例增加. 同时,设定输入技能从相关研究领域中出现频率最高的前 100 技能中随机产生.

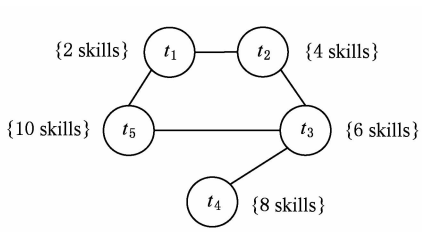

Fig. 2 Default grouping task.

图 2 默认分组任务图

1) *Cost_MNN* 度量团队沟通代价.

如图 3 所示,在组内采用同一种搜索策略 MST 时,组间的 2 种搜索策略对结果的影响虽然较小;但仍可以看出组间采用 SUM 搜索策略时,无论是在团队沟通代价还是团队人数方面,都具有较好的结果. 当任务总技能增多时,对比更明显. 当组内和组间都采用同一种搜索策略时,可以看出应用收益变量 *Profit* 时的搜索结果明显好于没有用到收益变量时的结果,使得团队沟通代价降低同时团队人数也较少.

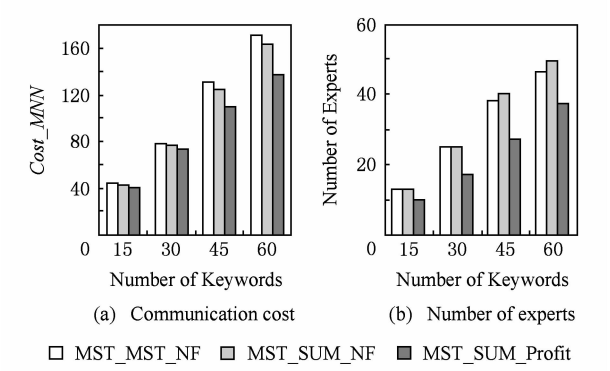

Fig. 3 Comparison of search strategies of inter-groups using *Cost_MNN*.

图 3 以 *Cost_MNN* 度量沟通代价时组间搜索策略对比

图 4 中组间都采用较好的一种搜索策略 SUM,同时各策略中也应用了收益变量 *Profit*. 可以看出,当组内采用 MST 搜索策略时,团队沟通代价相对较低;而组内采用 STAR 策略时,团队的沟通代价最高. 但是在团队人数上,SUM 搜索策略的结果团队人数相对较少. 由实验结果可知,当结果采用最小生成树沟通代价 $Cost\_MST_{intra}$ 对团队进行评价时,组内 MST 搜索策略和组间 SUM 搜索策略组合时能得到较好的结果. 在所有搜索策略中,使用收益变量 *Profit* 均可提高搜索质量.

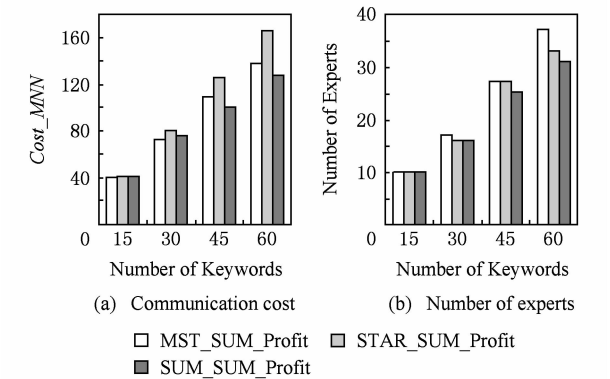

Fig. 4 Comparison of search strategies of intra-groups using *Cost_MNN*.

图 4 以 *Cost_MNN* 度量沟通代价时组内搜索策略对比

2) *Cost_SLL* 度量团队沟通代价

图 5(a)显示了当组内都采用 STAR 搜索策略时,组间采用 SUM 搜索策略得到的团队明显优于组间采用 MST 搜索策略得到的团队,即团队的沟通代价较低;同时由图 5(b)可以看出,组间采用 SUM 搜索策略时,得到的团队人数也较少. 然而,图 5(a)(b)显示了与图 3(a)(b)中相同的结果,当在搜索算法中使用收益变量 *Profit* 时,无论是在团队的沟通代价还是团队人数方面都有明显的优化.

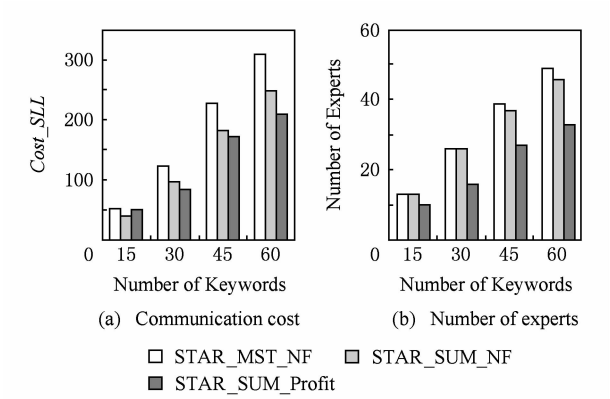

Fig. 5 Comparison of search strategies of inter-groups using *Cost_SLL*.

图 5 以 *Cost_SLL* 度量沟通代价时组间搜索策略对比

图 6 中算法组间均采用 SUM 搜索策略,从图 6(a)可以看出,当最终形成的团队沟通代价由 *Cost_SLL* 度量时,组内采用 STAR 沟通策略得到的团队沟通代价最低. 因为在 *Cost_SLL* 沟通代价中,在计算组内沟通代价时计算各组员与组长之间的距离,在计算组间沟通代价时计算组长之间的距离;而在 STAR 搜索算法中,先确定一个组长,然后再查找其

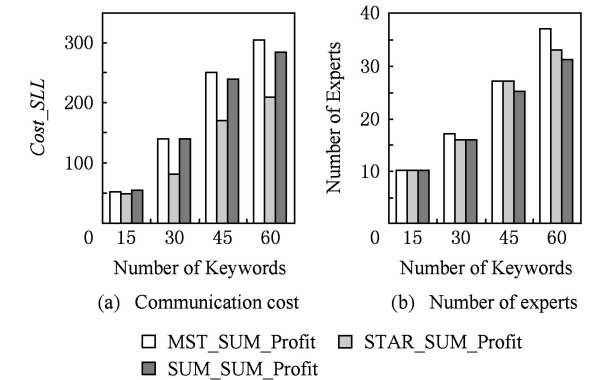

Fig. 6 Comparison of search strategies of intra-groups using *Cost_SLL*.

图 6 以 *Cost_SLL* 度量沟通代价时组内搜索策略对比

他与组长距离较近的组员,这种贪心策略符合团队沟通代价的计算方式.所以在使用 *Cost_SLL* 计算沟通代价时,STAR 搜索策略会得到沟通代价低且人数少的团队.

3) *Cost_SNN* 度量团队沟通代价

图 7 显示了与图 3 和图 5 相似的结果,当团队沟通代价采用 *Cost_SNN* 进行度量时,组间 SUM 搜索策略同样优于组间 MST 搜索策略;同时也说明了收益变量 *Profit* 的优势,只要采用收益变量 *Profit*,在沟通代价和团队人数方面都能得到较好的结果.

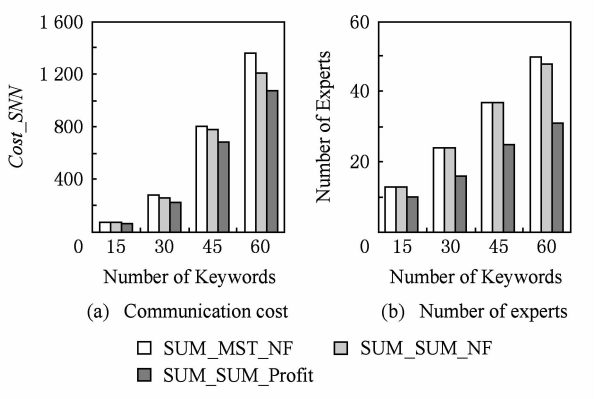

Fig. 7 Comparison of search strategies of inter-groups using *Cost_SNN*.

图 7 以 *Cost_SNN* 度量沟通代价时组间搜索策略对比

图 8(a)显示,当最终形成的团队沟通代价由 *Cost_SNN* 进行度量时,组内采用 SUM 搜索策略时,得到的团队沟通代价不低于其他 2 种组内的搜索策略,随着任务总技能数目的增多,SUM 搜索策略的优势明显.从图 8(b)可以看出,组内的 SUM 搜索策略得到的团队人数更少.

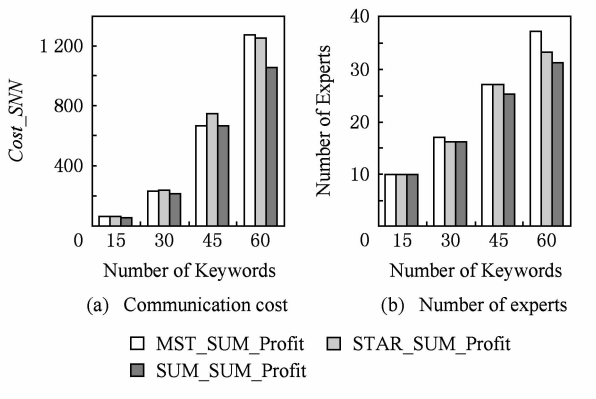

Fig. 8 Comparison of search strategies of intra-groups using *Cost_SNN*.

图 8 以 *Cost_SNN* 度量沟通代价时组内搜索策略对比

在本节实验中,分别在不同的沟通代价度量方法下,对比组内搜索策略和组间搜索策略.实验结果表明,组间 SUM 搜索策略无论在哪种沟通代价的度量方法下,都能得到相对较好的团队;而当团队由沟通代价 *Cost_MNN* 度量时,组内 MST 搜索策略表现最好;当团队由沟通代价 *Cost_SLL* 度量时,组内 STAR 搜索策略能得到较好的团队;当团队由沟通代价 *Cost_SNN* 度量时,组内 SUM 得到的团队沟通代价最低,并且团队人数较少.

6 结 论

本文提出了一种新的团队形成问题:社会网络上支持任务分组的团队形成问题.给出了此问题的相关定义,提出了 4 种有效的搜索策略.本文利用真实 DBLP 数据集对算法的有效性进行了评价.实验表明,本文所提出的贪心策略适用于相应的沟通代价度量,PRO 贪心策略则可以提高所有沟通代价度量下的查询结果.

参 考 文 献

[1] Kargar M, An A. Discovering top-*k* teams of experts with/without a leader in social networks [C] //Proc of the 20th ACM Int Conf on Information and Knowledge Management. New York: ACM, 2011: 985-994

[2] Anagnostopoulos A, Becchetti L, Castillo C, et al. Online team formation in social networks [C] //Proc of the 21st Int Conf on World Wide Web. New York: ACM, 2010: 839-848

[3] Kalyanmoy D. Search Methodologies [M]. Berlin: Springer, 2005: 273-316

[4] Meng Xiaofeng, Li Yong, Zhu Jianhua. Social computing in the era of big data: Opportunities and challenges [J]. Journal of Computer Research and Developent, 2013, 50(12): 2483-2491 (in Chinese)
(孟小峰, 李勇, 祝建华. 社会计算:大数据时代的机遇与挑战[J]. 计算机研究与发展, 2013, 50(12): 2483-2491)

[5] Chhabra M, Das S, Szymanski B. Team Formation in Social Networks [M]. Berlin: Springer, 2013: 291-299

[6] Lappas T, Liu L, Terzi E. Finding a team of experts in social networks [C] //Proc of the 15th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2009: 467-476

[7] Datta S, Majumder A, Naidu K. Capacitated team formation problem on social networks [C] //Proc of the 18th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2012: 1005-1013

[8] Anagnostopoulos A, Becchetti L, Castillo C, et al. Power in unity: Forming teams in large-scale community systems [C] //Proc of the 19th ACM Int Conf on Information and Knowledge Management. New York: ACM, 2010: 599-608

[9] Chen S J, Li L. Modeling team member characteristics for the formation of a multifunctional team in concurrent engineering [J]. IEEE Trans on Engineering Management, 2004, 51(2): 111-124

[10] Zzkarian A, Kusiak A. Forming teams: An analytical approach [J]. IIE Trans, 1999, 31(1): 85-97

[11] Wi H, Oh S, Mun J, et al. A team formation model based on knowledge and collaboration [J]. Expert Systems with Application, 2009, 36(5): 9121-9134

[12] Stephen P R. Mary Coulter: Management [M]. Beijing: Tsinghua University Press, 2007

[13] Fayol H. Administration General and Industrial [M]. Translated by Chi Ligeng, Zhang Xuan. Beijing: China Machine Press, 2007 (in Chinese)
(Fayol H. 工业管理与一般管理[M]. 迟力耕, 张璇, 译. 北京: 机械工业出版社, 2007)

[14] Baykasoglu A, Dereli T, Das S. Project team selection using fuzzy optimization approach [J]. Cybernetics and Systems, 2007, 38(2): 155-185

[15] Yang D N, Chen Y L, Lee W C, et al. On social-temporal group query with acquaintance constraint [C] //Proc of the 37th Int Conf on VLDB. New York: ACM, 2011: 397-408

[16] Gajewar A, Sarma A D. Multi skill collaborative teams based on densest subgraphs [C] //Proc of the 12th SIAM Int Conf on Data Mining. Anaheim, CA: Omnipress, 2012: 165-176

[17] Kargar M, An A, Zihayat M. Efficient Bi-objective Team Formation in Social Networks [M]. Berlin: Springer, 2012: 483-498

[18] Kargar M, Zihayat M, An A. Finding affordable and collaborative teams from a network of experts [C] //Proc of the 13th SIAM Int Conf on Data Mining. Anaheim, CA: Omnipress, 2013: 587-595

[19] David R C, Robert E T. Finding minimum spanning trees [J]. Siam Journal on Computing, 1976, 5(4): 724-742

[20] Li C, Shan M. Team formation for generalized tasks in expertise social networks [C] //Proc of IEEE Int Conf on Social Computing. Piscataway, NJ: IEEE, 2010: 9-16

[21] Michale Ley. DBLP [OL]. [2014-05-21]. http://dblp.uni-trier.de

Sun Huanliang, born in 1969. Professor in Shenyang Jianzhu University. Senior member of China Computer Federation. His main research interests include spatial database and data mining.

Jin Mingyu, born in 1989. MS of Shenyang Jianzhu University. His main research interests include spatial database and data mining.

Liu Junling, born in 1972. PhD candidate at Northeastern University. Member of China Computer Federation. Her main research interests include spatial database and data mining, etc.

Yu Ge, born in 1962. Professor and PhD supervisor of Northeastern University. Senior member of China Computer Federation. His main research interests include database, data mining, RFID, XML and Web data management.