《影响力最大化》研究报告*[二号黑体]*

*[注：方括号括起来并以蓝色斜体显示的文本，用于向作者提供指导，在发布此文档之前应该将其删除。]*

|  |  |  |  |
| --- | --- | --- | --- |
| 文档名称： | 《带时间感知的影响力最大化》研究报告draft1.0 | | |
| 单位名称： | 北京邮电大学网络与交换技术国家重点实验室宽带网研究中心 | | |
| 文档作者： | 王子珩 | 文档版本： | Draft2.0 *[最新的文档版本号]* |
| 创建日期： | 2019/04/01 | 修改日期： | 2019/05/28*[文档最后一次被修改的日期，需更新]* |

修订历史记录

*[在此要求记录每次版本修订和升级的相关内容，版本号为Draft1.2或Version1.0的形式]*

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 说明 | 作者 |
| 2019/05/28 | Draft1.0 | 完成影响力最大化问题的第一版本 | 王子珩 |
| 2019/06/25 | Draft2.0 | 将带时间的影响力最大化问题加入到阅读报告中 | 王子珩 |
| 2019/07/15 | Draft3.0 | 将对齐网络中的影响力问题加入到阅读报告中 | 王子珩 |
|  |  |  |  |
|  |  |  |  |

目 录

[1. 简介 4](#_Toc14270141)

[1.1 编写目的 4](#_Toc14270142)

[1.2 定义 4](#_Toc14270143)

[1.3 首字母缩写词和缩略语 4](#_Toc14270144)

[1.4 概述 4](#_Toc14270145)

[2. 引言 6](#_Toc14270146)

[3 影响力最大化的传播模型 7](#_Toc14270147)

[3.1 独立级联模型（IC）及其变种 7](#_Toc14270148)

[3.2 线性阈值模型（LT） 8](#_Toc14270149)

[3.3 触发模型（TR） 9](#_Toc14270150)

[3.4 其他模型 9](#_Toc14270151)

[3.5 小结 9](#_Toc14270152)

[4. 影响力最大化的算法 10](#_Toc14270153)

[4.1 基于模拟的方法 10](#_Toc14270154)

[4.2 基于代理的方法 11](#_Toc14270155)

[4.3 基于草图的方法 11](#_Toc14270156)

[4.3.1 前向影响草图 11](#_Toc14270157)

[4.3.2 反向可达草图 11](#_Toc14270158)

[4.4 小结 12](#_Toc14270159)

[5. 带时间感知的影响力最大化问题 13](#_Toc14270160)

[5.1带时间感知的影响力最大化的传播模型 13](#_Toc14270161)

[5.1.1 时延感知的独立级联模型（LAIC） 13](#_Toc14270162)

[5.1.2 连续激活和时间限制的IC模型（CT-IC） 13](#_Toc14270163)

[5.1.3 连续时间的传播模型（CT） 14](#_Toc14270164)

[5.1.4 与用户在线行为相关的时间和开销限制的影响力模型（TCIO） 14](#_Toc14270165)

[5.1.5 带相遇事件的独立级联模型（IC-M） 14](#_Toc14270166)

[5.2 带时间感知的影响力最大化的算法 15](#_Toc14270167)

[5.2.1 文献[3]中提出的算法 15](#_Toc14270168)

[5.2.2 文献[9]中提到的算法 16](#_Toc14270169)

[5.2.3 文献[10]中的算法 18](#_Toc14270170)

[5.2.4 文献[12]中的算法 18](#_Toc14270171)

[5.2.5 文献[13]中的算法 21](#_Toc14270172)

[6.对齐网络中的影响力问题 24](#_Toc14270173)

[7. 参考文献 25](#_Toc14270174)

# 1. 简介

## 1.1 编写目的

本文针对国内外影响力最大化的主要研究方法和相关研究现状进行了总结，对现有研究所采用的理论方法和主要思路进行了较详细的分析。

## 1.2 定义

|  |  |
| --- | --- |
| **英文全称或缩写** | **定义** |
| Influence Spread | 一个用户集合S的影响传播表示集合S所能影响到的用户数。 |
| Diffusion Model | 给定一个社交网络G=(V,E)和一个用户集合S，传播模型M捕获了S在G上传播信息的随机过程 |
| Influence Maximization | 给出一个图G=(V,E)，一个扩散模型M和一个正整数k，影响力最大化算法的目标是从V中挑出k个节点组成一个集合S使得它的影响范围最广。 |
| RR Set | 给定一个图G’为图G上的草图，RRG’（v）表示在G’中可以到达v的所有结点。 |

## 1.3 首字母缩写词和缩略语

|  |  |  |
| --- | --- | --- |
| **缩略语** | **英文全称** | **中文解释** |
| IM | Influence Maximization | 影响力最大化 |
| LAIC |  | 时延感知的独立级联模型 |

## 1.4 概述

本研究报告分为概述、前言、国内外研究现状和发展趋势以及总结四个部分：

第一章，概述部分简要说明本文的编写目的，对一些定义和缩略语进行解释，概括整个报告的总体内容；

第二章，前言部分简要阐述IM问题的重要性。

第三章，给出近年来IM问题在国内外研究现状和发展趋势，主要介绍IM的模型。

第四章，给出近年来IM问题在国内外研究现状和发展趋势，主要介绍IM的算法。

第五章，给出近年来带时间感知影响力最大化问题的模型和算法。

第六章，

第七章，

第八章。

*[对报告内容进行概述，并说明每一章的主要内容。]*

# 2. 引言

过去的几十年见证了在线互动的蓬勃发展，产生了数量空前的内容。在线社交网络的普及引起了人们对信息传播的关注，因为一条信息可以通过在网络上朋友之间的“口碑”传播迅速变得无处不在。这种扩散现象已经在许多的应用中被证明是强大的，例如采用政治立场和技术创新。最近的一个例子是唐纳德·特朗普的总统竞选。2016年，Twitter几乎每天都被用作竞选工具。因此，在线社交网络中的信息传播吸引了计算机科学、物理学、流行病学等多个领域的广泛研究[1]

随着科学技术的发展，越来越多的个人和公司使用社交网络来促进他们的业务，传播新闻、思想和观点，社交网络有许多研究应用，如链接预测、谣言控制、影响力最大化。然而，大多数研究工作只考虑单一影响的传播。事实上，不同的观点甚至相反的观点在社交网络中同时传播的情况经常发生。一个显然的问题是，当社会网络中存在竞争时，影响力如何传播，我们称之为影响力竞争问题。影响力阻塞最大化( IBM )问题是一种影响力竞争问题。IBM问题的目标是尽可能阻止竞争对手的影响力的扩散[2]。

# 3 影响力最大化的传播模型

传播模型在影响力最大化问题中非常重要，不同的模型应用不同的机制来捕获用户如何从非活跃的状态到激活的状态。

## 3.1 独立级联模型（IC）及其变种

独立级联模型是一个非常经典并且被研究很多的模型[1]。

传统的独立级联模型的主要思想是当一个节点u被激活时，它会以概率p(u,v)对它未激活的出边邻居节点v尝试激活，这种尝试仅仅进行一次，而且这些尝试之间是互相独立的，即u对v的激活不会受到其他节点的影响。

它的传播过程如下：

·给定初始的活跃节点集合S，当在时刻t节点u被激活后，它就获得了一次对它的邻居节点v产生影响的机会，成功的概率为p(u,v)，是随机赋予的系统参数。

·若v有多个邻居节点都是新近被激活的节点，那么这些节点将以任意顺序尝试激活节点v。如果节点u成功激活节点v，那么在t+1时刻，节点v转为活跃状态。

·在t+1时刻，节点v将对其他节点产生影响，重复上述过程。

·结束条件：网络中不存在有影响力的网络节点。

在文献[2]中，作者提出了竞争的独立级联模型（CIC）：一个社交网络G，每个节点有三种状态，未激活，正向状态，反向状态。每个边有正向传播概率PP（u,v）和负向传播概率PN（u,v）一开始有两个种子集合，一个是正向种子集合一个是负向种子集合。两种传播方式各自独立。在t步时，一个未激活的节点u，变为负向的，他有一次机会对他的未激活邻居节点进行负向传播，反之亦然。如果一个节点被影响到了了，它在t+1时刻被激活。U只有一次机会激活别人。如果在某一步时，没有新的节点被激活，传播结束。

在文献[3]中作者提出了延迟感知的独立级联模型（LAIC）：通过考虑时间延迟来扩展传统IC模式，这与现实社会网络更加一致。在这个模型下，每个结点的状态分为三种，激活，延迟激活，非激活。对于一个在种子集合中的结点u，可以认为他在t=0时刻处于活跃态，如果它成功影响到了一个非活跃邻居节点v，那么节点v变为延迟活跃节点。根据之前定义的影响力延迟的分布，每条边e（u，v）可以扩展到，其中代表了一个非活跃节点v被他的活跃邻居u在一个激活延迟以概率激活。是一个延迟分布。

LAIC模型的过程如下:在任意一步t>0，在种子集S中的结点u可以有一次激活邻居节点v（v是处于非激活状态或者延迟激活状态）与传统IC模型不同的是，被激活的节点v会变味延迟激活状态而不是激活状态，在经过延迟之后，延迟激活节点v会被种子集合S激活。

## 3.2 线性阈值模型（LT）

线性阈值模型的主要思想它对图中每个节点v都有一个激活阈值，当v的入邻居影响力大于它的阈值时，就会被激活。

传统线性阈值模型的传播过程如下：

·集合中的任意节点v随机分配阈值θ[v]∈[0,1]。只有当节点v的新处于激活状态的邻居节点对它的影响力大于该阈值时，节点v才能被激活。

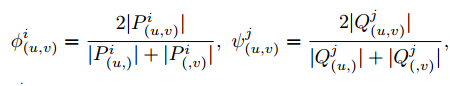
·用权值b[u.v]表示节点v被它的邻居节点u的影响，∑u∈in(v)  b[u,v]≤1表示节点v的处于活跃状态的邻居节点对它的影响力之和。这里in(v)是v的入边邻居节点集合。

·给定初始的活跃节点集合S，在t时刻，所有在t-1时刻处于活跃状态的节点仍保持活跃，并且当这一时刻节点v的邻居节点的影响力之和大于节点b的阈值时，节点v被激活.

·节点v被激活后，下一时刻将对它的邻居节点产生影响，重复上述过程。

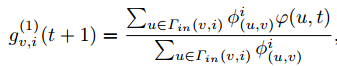
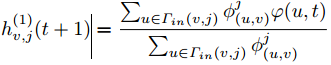
·结束条件：当网络中已存在的所有活跃节点中任意活跃节点的影响力之和都不能激活他们的处于非活跃状态的邻居节点时。

在文献[4]中作者将扩展传统的线性阈值(LT)模型来处理多对齐多关系网络(MMNs)的信息扩散，提出了M&M模型。基于MMNs M = (U, E, R)，通过pathsim估计每对具有不同扩散关系的用户的权重。式中，用户u与v之间的网内(网间)扩散权与关系i(j)定义为:



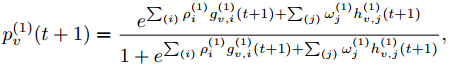
其中是为网内(网间)扩散元路径实例集，根据关系i(j)从u开始到v结束。表示以用户u、v分别作为起始用户和结束用户的元路径实例数。

在传统LT模型的基础上，影响在网络中以离散的步骤传播。在步骤t中，所有激活用户都保持活动状态，如果接收到的影响超过其阈值，则可以激活非激活用户。只有在步骤t时被激活的用户才会影响到其在步骤t+1时的邻居，且在一个网络中(如G(1))用户v在网络内关系i和网络间关系j下的激活概率分别为

其中是用户v在关系i和关系j下的邻居集合。表示用户u在时间t被激活。

通过对网络内部和网络间的各种关系进行聚合，可以得到v(1)的整体激活概率



代表了每个关系的权重，满足

## 3.3 触发模型（TR）

触发模型的主要思想是结合了IC模型和LT模型。在扩散过程中，TR模型会节点v随机选择一个触发集合Tv。

它的传播过程如下;

·定义一个初始集合S，为每个节点随机选择一个触发集合。

·非激活的v在t时刻会被激活的条件为：v的一个属于Tv中的邻居在t-1时刻被激活。

·被激活结点继续上述传播。

## 3.4 其他模型

由于贪心算法代价太大，文献[5]提出了影响力最大化树状结构（MIA）来计算影响的传播。MIA的主要思想如下：给定一个社交网络G，两个结点u，v属于结点集合V，结点u可以影响结点v的条件是两个结点之间有一条路径，这条路径可以表示为，结点u能激活v的概率可以表示为。在MIA模型下，当u和v之间存在多条路径时，选择概率最大的路径，因为通过这条路径u有最大的几率去激活v，作者将这条路径定义为。在MIA模型下，结点u只能通过去影响结点v。

## 3.5 小结

0

# 4. 影响力最大化的算法

虽然影响力最大化问题的计算是非常复杂的，但是当影响函数满足单调性和子模性的时候，该问题可以得出近似的最优解。影响函数的单调性和子模性的定义如下：

单调性：一个影响函数是单调的当且仅当它满足，直观来说就是在集合S中加入一个结点后，它的影响范围不会减少。

子模性：一个影响函数是子模的当且仅当它满足如下所示的条件：

。直观来说就是集合S中加入一个结点，它所带来的边际收益要小于上一个结点的加入带来的边际收益。

影响力最大化算法大都是基于贪心算法的改进。贪心算法的思想是首先初始化一个空的种子集S，在每次迭代中都选择一个有最大边际收益的结点加入到S中，当种子集S中已经有k个种子的时候算法停止。

## 4.1 基于模拟的方法

基于模拟的主要思想是利用蒙特卡洛模拟去估计每个节点的影响范围函数进而计算每个结点的边际收益，之后使用贪心算法每次迭代都选择边际收益最高的结点加入种子集合。为了提高估算边际收益的精确度，此算法一般要跑10000轮再取平均。显然，由于需要进行的蒙特卡洛模拟次数过多，传统的基于模拟的方法无法在一个规模较大的图中应用。所以有学者对其进行改进。

在文献[6]中作者提出了UBLF算法。作者使用代表一个节点v在步骤t被种子集S激活的概率。影响函数可以表示为。对于t=1,2,…N-|S|，可以得到这样一个不等式。随后作者利用矩阵形式对其进行简化，PP是传播矩阵，pp(u,v)是对应的两个节点之间的传播概率。我们使用一个行向量来表示未激活节点在t时刻被激活的概率。上述不等式可以表示为。影响函数的上界可以表示为。UBLF的算法步骤如下：

1. 将带有传播概率的扩散网络信息转换成传播概率矩阵PP。

2. 以PP为处理对象，利用上界计算理论公式，计算每个节点作为初始扩散点的传播范围上界。

3. 依次选取上界较大的节点进行蒙特卡洛模拟，用真实值进行序列对比；若某节点v的真实值比其它点的上界值还要大，那么节点就入选整个网络的关键节点集，如此直到选满k个节点为止。

## 4.2 基于代理的方法

用代理模型的方法去计算影响传播函数，而不是用很繁琐的蒙特卡洛模拟，这样可以使得IM算法在大规模网络中有很好的延展性。基于代理的方法一般都是为特定的传播模型量身定做的，它的通用性不好。

在文献[7]中作者提出了SimPath算法，他是基于LT模型。作者首先使用经典的回溯算法找到所有的简单路径，进而计算出种子集合的影响函数。有了种子集合的影响函数，作者将它代入贪婪算法(用CELF优化)中，得到一个影响力最大化的算法。这构成了SIMPATH算法的核心。为了进一步提高SIMPATH的效率，作者提出了两个优化方法，第一个是顶点覆盖优化也就是找到最小顶点覆盖，可以减少影响函数的计算量，第二个是一个前向的优化。

## 4.3 基于草图的方法

这种方法的主要焦点是提高基于模拟的方法的理论效率，同时使得近似值可以得到保证。为了避免运行大量模拟，基于草图的方法基于特定的扩散模型预先计算多个草图，然后利用草图来评估影响扩散。它主要分为两大类，前向影响草图和反向可达草图。

### 4.3.1 前向影响草图

前向影响草图的主要思想与蒙特卡洛方法相似[1]。在一个图G=(V,E)中，假设每条边上的传播概率为P1,P2,,,Pn。利用蒙特卡洛模拟计算一个节点的影响范围时，首先将图中所有边以1-Pk的概率移除，在这个子图中计算当前节点的影响范围。S的影响传播函数为，其中Gi为去某条边之后的子图，为在子图Gi中种子集合S可以到达的结点的集合。

### 4.3.2 反向可达草图

文献[8]中，作者提出了两阶段的影响力最大化算法（TIM）。

## 4.4 小结

000

# 5. 带时间感知的影响力最大化问题

## 5.1带时间感知的影响力最大化的传播模型

### 5.1.1 时延感知的独立级联模型（LAIC）

LAIC模型通过考虑时间延迟来扩展传统IC模式，这与现实社会网络更加一致。在这个模型下，每个结点的状态分为三种，激活，延迟激活，非激活。对于一个在种子集合中的结点u，可以认为他在t=0时刻处于活跃态，如果它成功影响到了一个非活跃邻居节点v，那么节点v变为延迟活跃节点[10]。

在LAIC模型中，当一个节点u在t步第一次被激活时，他会在步骤对它的当前不活跃节点v以概率进行激活，其中是一个影响延迟。注意，一个节点最多可以被激活一次。如果一个节点有多个邻居影响它，则只关注时间最早的那一次，之后的都会被忽略[9]。

在文献[9]中作者还给出了带时间限制的影响力最大化问题的定义：给定一个社交网络G=（V，E），一个时间限制T，一个正整数K，两个节点之间的传播概率Puv和每个节点的延迟分布，找到一个包含k各节点的种子集合S，使得在时间限制T内，它的影响范围最广。

### 5.1.2 连续激活和时间限制的IC模型（CT-IC）

文献[10]的作者对CT-IC模型传播方式进行了阐述并且给出了在CT-IC模型下IM问题的定义：

CT-IC模型的传播方式为：给定一个有向图G=（V，E）每条边上都有一个概率pp0(u,v)代表节点u在某一步激活v的概率，给定给一个种子集合S，和一个时间限制T，CT-IC模型传播方式如下：在t=0时，初始集合中的种子被激活，随后开始进行传播。使用At表示在时间t的活跃种子，在时间t，所有属于At的节点都要以概率试图去激活它的未活跃邻居。

其中tu是节点u的激活时间。并且将定义为，其中是个单调递减函数并且。作者使用来定义函数。

在CT-IT模型下的IM问题可以被定义为：找到一个种子集合S，使得S的影响范围最广。

### 5.1.3 连续时间的传播模型（CT）

文献[11]的作者阐述了他所提出的连续时间的传播模型。此模型是基于静态网络的。

它的传播过程如下：对于每条边j和i，用一个传输函数连起来，这个数值是随时间变化的。当种子节点集A在t = 0时被激活时，开始扩散过程。种子节点试图感染它们的子节点，当子节点i在时间ti被激活时，他又会去试图激活它的子节点。每一次通过边缘的传输都需要一个随机传输时间，它是从传输函数中得到的。有些时候会出现传输函数得到的数目是无穷大的，那就代表着永远无法激活它的子节点。

### 5.1.4 与用户在线行为相关的时间和开销限制的影响力模型（TCIO）

文献[12]的作者阐述了他所提出的TCIO模型。

TCIO的传播过程如下：用G(V,E)代表一个有向社交网络，V代表着用户集合，E代表着用户之间的关系的集合。每条边上有一个转发概率puv，它代表了当v在线且u已经转发消息时，节点v从节点u获得消息的概率是多少。一天的时间被划分为多个时间段，一个节点可以根据其在线模式辨别在一个时间段是在线还是离线。mvt表示节点v在时间段t中的在线模式，等于0时代表离线，等于1是代表在线。当选择每个节点作为传播消息的种子节点时，它都有一个价格。设T为消息过期时间，t0为消息发布时间。设B为选择种子节点在社交网络中传播消息的总预算。

作者还给出了时间敏感的影响最大化问题的定义：给定一个过期时间T，一个总预算B，一个用户在线行为M，当在集合S中的节点发布消息，使得网络中最多的结点收到消息。每个转发消息的节点都可以根据其在线模式反复影响其邻居，直到其邻居受到影响转发消息或达到过期时间。

### 5.1.5 带相遇事件的独立级联模型（IC-M）

文献[13]的作者提出了IC-M模型。作者认为影响从一个人到另一个人的传播是需要二者之间“相遇”才有机会传播的。这里所谓的相遇不一定非要是两个人面对面在一起，也可以是在社交媒体上的点赞转发。

IC-M过程如下：给定图G=(V,E)，边（u,v）属于E，边上会有一个相遇概率m(u,v)，m（u,v）定义为在0,1之间符合均匀分布。在第0步时，种子节点被激活，在之后的任意步骤中，一个激活的节点u会以概率m(u,v)独立的遇到他的不活跃邻居节点v。如果u与v之间首次相遇，则给u一次尝试激活v的机会，成功概率p(u,v)。如果尝试成功，v在步骤t处变得活跃，并将在t+1处开始传播影响。当所有的活动节点都与它们的邻居相遇，并且没有新的节点可以被激活时，扩散过程就停止了。

### 5.1.6 使用IC模型和FF模型预测社交网络随时间的潜在变化

文献[14]中作者认为经典的影响最大化问题忽略了影响传播的时间，而影响传播的时间在现实中可能是非常重要的，在此期间，潜在的社交网络可能会进化。因此，作者正式地将这个经典的影响力最大化问题重新定义如下。

Propagation Time-conscious Influence Maximization Problem：设G0 = (V0, E0)为t0时刻的当前网络，k为预算。Tr为影响传播时间，并且此时的G0演化到了Gr=（Vr,Er）。Propagation Time-conscious Influence Maximization的目标是在t0时刻选择一个种子集合S属于V0 ，使得在时刻tr最大。并且假设Gr的拓扑在t0时刻是不被知道的。

作者假设传播符合独立级联模型（ICM），并且结合了Forest Fire model（森林火灾模型）来预测从G0到Gr的网络拓扑结构。

## 5.2 带时间感知的影响力最大化的算法

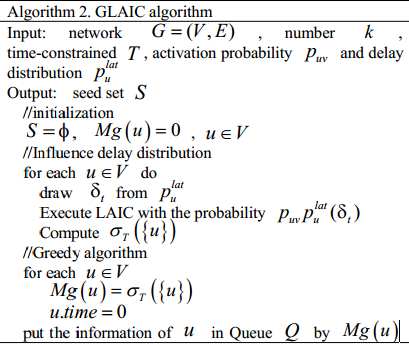
### 5.2.1 文献[3]中提出的算法

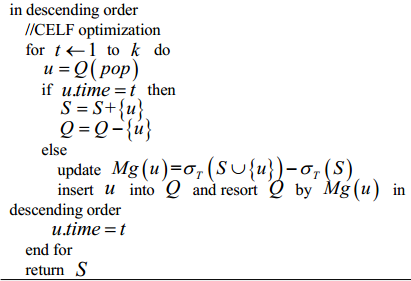
#### 5.2.1.1 用CELF优化贪婪算法

首先利用基本的贪婪算法的思想即每次挑选边际收益最大的结点加入到种子集中。作者使用表示期望激活结点的数目，使用Mg(u)表示结点u边缘收益。如果只是利用贪心算法，在处理大型网络的时候会出现效率低下的问题，所以作者使用CELF对其进行优化。CELF利用影响最大化目标的子模性，显著降低了计算节点影响扩散的成本。CELF优化的原理是，由于节点的子模块性，使得节点在当前步骤中的边际影响扩散不大于前一步的边际影响扩散，从而极大地缩短了计算结点边际收益的时间。更具体地说，为了维护节点的边际影响扩散，CELF维护一个优先队列，将每个结点的边际收益降序排列，因此，不需要在每次迭代时都对每个结点的边际收益进行重新计算，自然会选择顶部节点作为下一个被选择的种子结点。

#### 5.2.1.2 GLAIC算法

作者将经过CELF优化的贪心算法与幂律延迟分布相结合提出了在LAIC模型下的GLAIC算法。在该算法中，作者用T代表时间限制，S代表种子集合，Mg代表边际收益。与其他算法相比，作者的算法首先考虑影响延迟分布来评估影响扩散，作者从中抽取代表一个时间延迟，用代表激活概率，那么结点u可以激活结点v的概率为。算法步骤如下：





### 5.2.2 文献[9]中提到的算法

#### 5.2.2.1 基于影响传播路径（ISP）的算法

作者首先将带影响延迟的社交网络进行扩张：在LAIC模型中，当一个节点u在时间t被激活的时候，他会以概率在时间对他的不活跃邻居节点进行激活。为了加入传播时延，我们使用节点u，v之间的边扩展为，对于每个都会被分配两个值，一个是length（长度），一个是prob（概率）。如下图所示：

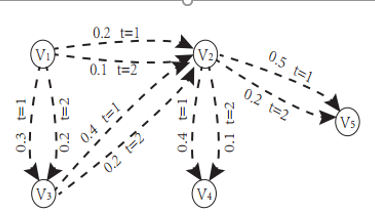
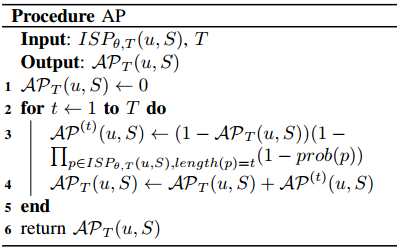


图4-1

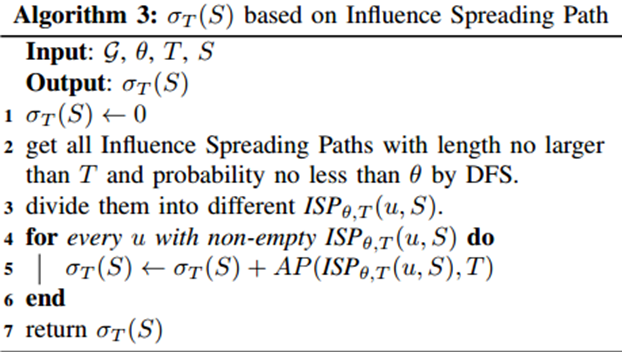
作者随后介绍了带限制的影响传播路径：给定一个种子集S,预期的影响传播时间T, 是预期的激活的节点数量用表示。其中是集合S在时间T内可以激活u的概率，为了评估每个节点u的，作者给出了拓展图中影响传播路径的定义如下：给定一个种子集S和有向多重图G=(V,E)，一个简单路径是影响传播路径的条件是：u1属于S，ui不属于S。激活时间就是每条边的length之和，激活概率就是每条边的概率的乘积。对于一个种子集合S，作者

用去表示所有以u为最终节点的影响传播路径。由于路径数量巨大，作者使用两种方法进行剪枝：第一种是将长度大于T的剪枝，第二种是将概率小于的进行剪枝。最终得到的影响传播路径用表示。

作者随后介绍了基于ISP的激活概率的计算：假设所有路径互相独立，可以计算集合S在时间T内可以激活节点u的概率



最后作者给出了基于ISP的影响函数的算法：

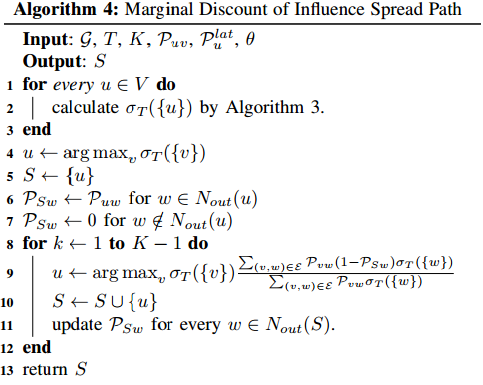


#### 5.2.2.2 更快的边际影响扩展估计

假设当前选择的种子集为S，作者要计算节点v加入S时的边际影响扩散增量，显然这个收益不能大于，然而在选择第一个种子节点的时候是已经知道的。作者将近似为：



其中PSw是w种子节点立马激活的概率。上述公式的原理是随着边际影响的增加，会减小。v的邻居结点被种子集合S激活的概率越大，的折扣也就越大。利用这种边际影响扩散增加近似，提出了求解时间约束影响最大化问题的算法4，如下：



### 5.2.3 文献[10]中的算法

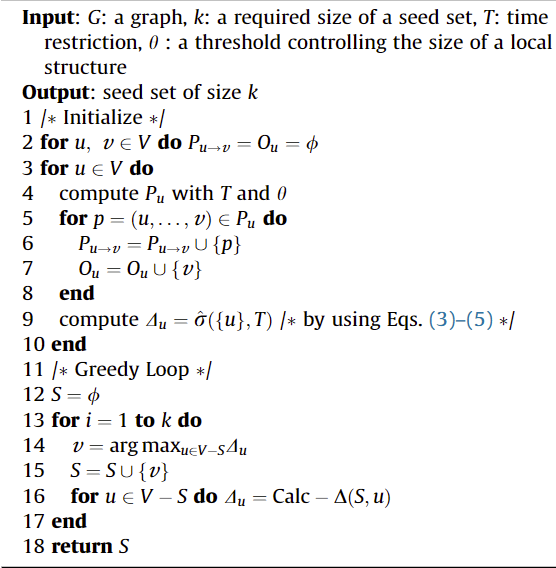
#### 5.2.3.1 CT-IPA算法

现有的实验中，可以看出IC模型中的贪婪算法是无法适应于CT-IC模型的，所以作者通过改进IC模型中的IPA算法并且提出了连续激活和时间受限的影响路径算法CT-IPA。

IPA通过将独立的影响路径作为影响扩散评估的基本单位来评估种子节点的影响扩散。IPA通过控制影响路径的数量来扩大影响扩散评估，这相当于丢弃传播可能性小于预定阈值的可忽略的影响路径。通过改变影响路径的影响传播定义，可以无缝地完成从IPA到CT-IPA的扩展。在IC模型中，影响路径的影响扩散是通过对路径中每条边的实值传播概率进行多次迭代来获得的。在CT-IC模型中，影响路径的影响传播为infp(,),涉及了一些矩阵乘法。

作者从节点u开始定义重要路径，pu= ，其中SPu是所有从u开始的简单路径。从节点u到v的重要路径集合为，这代表着u可以从这个集合中的一条路径影响v，最终，节点u的影响区域可以被定义为

CT-IPA算法如下：

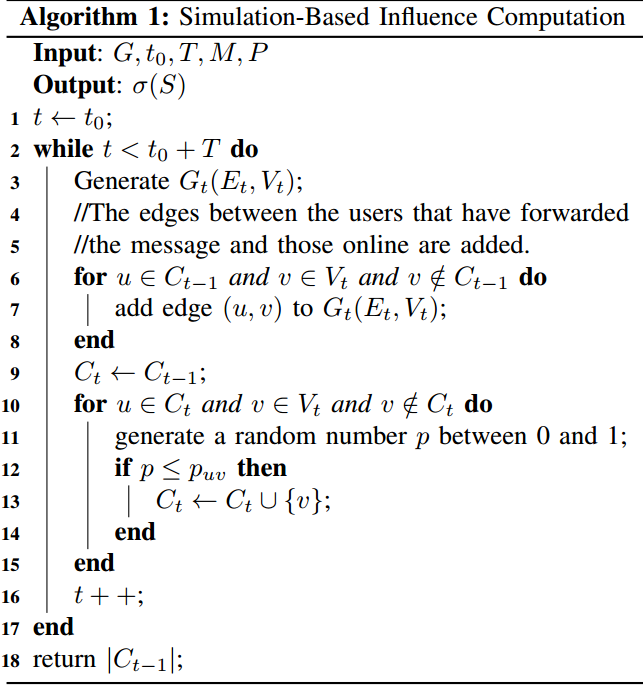


### 5.2.4 文献[12]中的算法

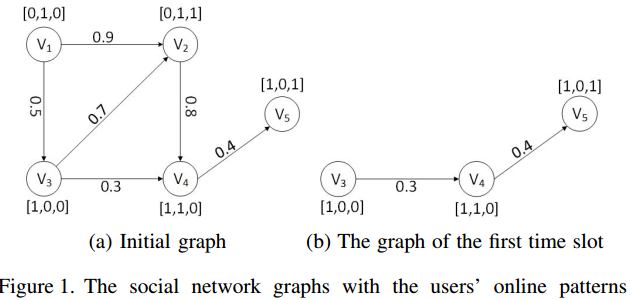
#### 5.2.4.1 基于模拟的影响计算

当作者考虑用户的在线模式时，在考虑时间段t时就只需要考虑在时间段t中在线的用户。作者使用表示在时间段t内在线的用户。用Ct表示在时间段t之后已经传递消息的节点的集合。作者的算法运行在t0到t0 + T的所有时间段。对于任意一条边都有一个概率puv代表v转发了u的消息。在算法中，作者随机生成一个数p，如果puv大于等于p，那么节点v就会转发u给他的消息。接收和转发消息的节点会在时间序列上反复影响它们的在线邻居。如果没有不受影响的节点或到达时间T，则此过程结束。

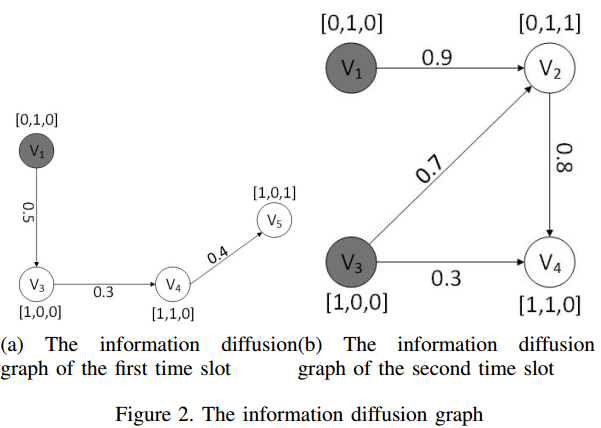
如算法1所示，作者使用仿真来计算节点在信息扩散过程中是否受到影响。



生成的如下所示：



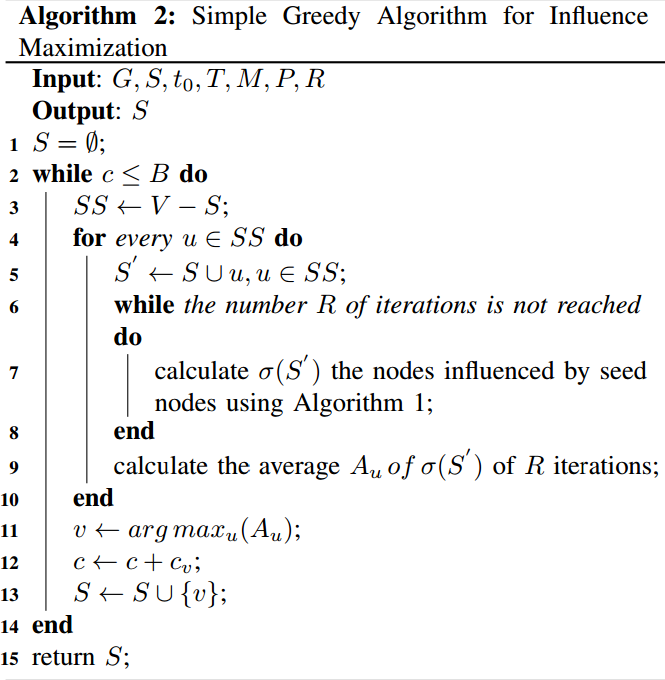
节点附近括号中的数字是用户的在线模式，比如说V1[0,1,0]代表这节点在第二个时间段在线。第一个时间段的图显示，只有V3、V4和V5是在线的，包含离线节点的边被修剪。



从图2中，我们可以看到V1发布了一条消息(假设用户在当前时间段离线，但是作为一个源节点的话可以认为他是在线的)，这时候只有V3、V4和V5是在线的。V3转发V1消息的概率为0.5，此时产生一个随机数（假设为0.2）因为0.5大于0.2，所以V3转发了消息。现在V4也在线，我们假设产生一个随机数为0.5，大于V4的转发概率（0.3）所以V4不会转发消息。信息扩散过程在第一个时间段终止。然后在第二个时隙中，图中的新边是第二个时隙中在线用户之间的边。在第一个时间段内成功转发消息的结点在第二个时间段内要继续向邻居转发，同时删除在第一个时间段内已经成功转发的两个节点之间的边。如果没有不受影响的节点或到达时间T，则进程终止。当进程终止时，返回受影响节点的数量。

#### 5.2.4.2 对于影响力最大化问题的简单贪心算法

作者提出了算法2来模拟时间敏感的影响沿时间序列扩散，并求解时间敏感的影响力最大化问题。请注意,算法2与传统的不考虑时间因素的影响最大化问题不同。TCIO模型必须保证有单调性和子模性。算法2基于爬山贪婪算法，为了减小误差，作者使用R轮取平均的做法来选择种子。

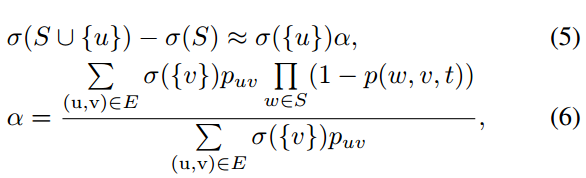


#### 5.2.4.3 Greedy on Maximal Added Influence

前两个算法在最坏的情况下，它没有性能保证。进一步设计了一种基于最大影响的改进算法。

算法3 (GMAI算法)利用影响权重计算信息扩散过程中的附加影响。因为影响函数是单调的、子模的算法反复添加对种子集S影响最大的节点。我们可以根据种子节点的近似影响增益来选择种子节点。

是种子节点集可以影响到的节点数量。作者通过 来计算增加影响，次我们选择一个种子节点，我们计算当前的增加影响。在大型社交网络中，这需要花费大量的时间和空间。作者使用模拟结果来近似增加的影响。



K是种子节点集合的大小，Puv是节点v转发节点u的概率。P(w,v,t)是节点v在时间段t之前转发种子节点w消息的概率。这个可以在算法1中被计算。

每个节点的影响范围都可以通过算法1去计算得到。在每个节点的仿真中，作者逐一逼近一个节点的附加影响。在此基础上，提出了算法3。算法3首先利用算法1计算出每个节点的影响，并全部记录下来。然后选择影响最大的第一个种子节点作为第一个种子节点。然后，利用上述公式依次选择种子节点，直到得到总预算。为了在预算范围内获得性能最佳、性价比最高的种子节点，我们进行了一些优化，选择了对每个节点成本增加影响的最大比例。

### 5.2.5 文献[13]中的算法

#### 5.2.5.1 MIA-M算法

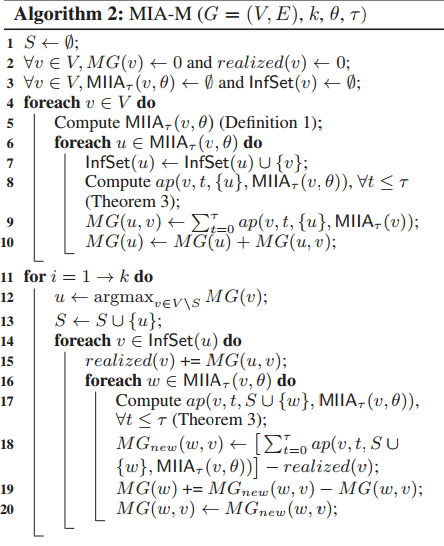
在描述算法之前，先介绍一些必要的符号。对于一组节点u,v，P(u,v)代表在图G中从u到v的所有路径的集合。给定一条u到v的路径，他的传播概率为：。随后作者定义了从u到v的最大影响路径MIP(u,v)为所有路径中概率最大的那个。如果u，v之间没有路径，则。此外，对于每个u,v最多需要一个MIP(u,v)。为了计算MIP，如果将传播概率P(u,v)转化为变权重logp(u,v)。计算MIP(u,v)就可以等价为找两点之间的最短路径，可以使用迪杰斯特拉算法。

对于MIA-M。我们介绍了路径P的augmented长度LA(P), 考虑相遇事件和截止日期的约束。考虑一条边（ui,uj）属于P，由于随机的相遇事件，ui在第t步激活后，其影响不会在t + 1时刻传播到uj。相反，传播可能需要多个步骤，这些步骤的数量是一个随机变量Xi,j，也可以解释为ui激活后ui与uj第一次相遇所需要的伯努利试验次数。显然，Xi,j属于几何分布，传播成功的概率为，期望为，标准差为。这里我们提出估算Xi,j的值为，并且对路径P的增强路径长度LA(P)定义为

**构建树状结构**：对于G中的任意节点v，我们利用v的最大树形影响(MIIA)来近似所有结点u（不包含v）到v的影响。为了让结点v成为根节点，作者首先在所有到v的最大影响路径上取并集。之后，将执行两个修剪步骤。首先,删除传播概率低于预定义的影响阈值的路径。其次,考虑到期限的影响,作者消除长度大于τ的路径。

**定义1：Maximum Influence In-Arborescence**：给定一个影响阈值，一个最后期限，对于任意结点v的最大影响入树为：

MIA-M算法如下所示：



#### 5.2.5.2 MIA-C算法

MIA-C算法包含两个步骤。首先，对于任意一条边(u,v)，我们估算一个转换传播概率，它包含相遇概率m，影响概率p，和截止时间T。旨在模拟原IC模型中IC- m模型下的影响扩散。二、取得全部pc(u,v)，作者将这些转换后的概率作为IC模型的参数，运行为IC选择k个种子的MIA算法。pc(u,v)的计算为：

经过概率转换步骤后，我们利用MIA(算法4，(Chen, Wang, and Wang 2010))来寻找种子集，使得MIA- c能够非常高效地利用节点边际收益的更新。

### 5.2.6 文献[14]中的算法

作者在文献[14]中提出了PROTEUS-GENIE 算法，PROTEUS-GENIE算法的目标是利用FFM预测底层网络从t0到tr的拓扑结构，贪婪地选择边际期望影响最大的节点。

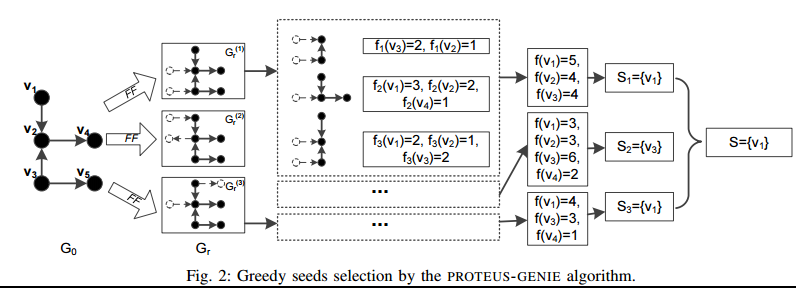
在PROTEUS-GENIE中种子的选择过程如下：

第一：在t0给定一个网络G0，估算所有v属于V0的在时刻tr，在社交网络Gr中的边际影响，Gr是通过在G0上运行FFM模型来得出的。通过拟合选择正向燃烧概率和反向燃烧比。

第二：选择期望影响最大的节点作为第一个节点，并将其从Gr中删除。

第三：对前两个步骤执行k轮迭代，选择k个种子作为S1。注意，在前面的步骤中，我们使用FFM生成一个目标网络Gr，这将导致tr时刻的确定性网络。然而，使用FFM进行t0到tr的网络演化是一个随机过程，单轮仿真无法准确描述。因此，前三个步骤分别执行I轮，导致I个不同的Gr实例，表示为 ，因此，种子集S1，…， SI是在I轮之后生成的。最后，将这些种子的排序进行汇总，选择总体排序最高的前k个节点作为最终的种子集S。

总的流程如下所示：



# 6.对齐网络中的影响力问题

由于本章中的问题不太好进行分类总结，所以就按照文献来总结了。

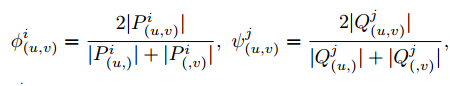
## 6.1 Influence maximization Across Partially Aligned Heterogenous Social Networks

在文献[15]中，作者认为在现实世界中，社交网络通常包含异构信息，例如各种类型的节点和复杂的链接，用户通过这些节点广泛连接，并有多个渠道相互影响。所以作者提出了对齐异构网络中的影响力最大化，并给出定义如下：给定两个部分对齐的网络G(1)，G(2)以及一个在二者之间的无向锚链接。G(1)，G(2)的用户集合可以表示为U(1)，U(2)。，其中Z属于两个网络用户集合的并集，这代表影响函数，也就是用户种子集合Z可以影响到的人数。AHI问题的目标就是寻找一个最优的包含d个用户的种子集Z\*去在网络中影响到最多的人。

随后作者又定义了多对齐多关系网络：给定两个部分对齐的网络G(1)，G(2)，我们给出MMN的定义为M=(U,E,R)，其中U是U(1)，U(2)的并集，代表在网络中所有结点的集合。E是网络中所有节点之间链接的集合，它的元素e可以用（u，v，r）表示，代表了（u，v）之间有一个链接关系r。R是链接种类集合。

在传统LT模型中，给一个同构网络G=（V，E），一个节点uk可以被激活，是根据uk与它的入邻居之间的权重之和是否大于激活它所需要的阈值来定义的。

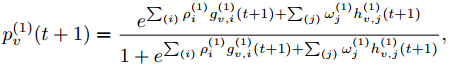
基于MMNs M = (U, E, R)，通过pathsim估计每对具有不同扩散关系的用户的权重。其中，用户u与v之间的网内(网间)扩散权重与关系i(j)定义为:



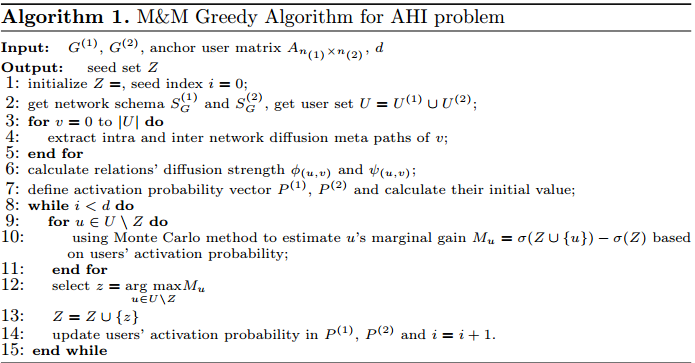
其中u、v分别作为起始用户和结束用户的元路径实例数。

在传统LT模型的基础上，影响在网络中以离散的步骤传播。在步骤t中，所有激活用户都保持活动状态，如果接收到的影响超过其阈值，则可以激活非激活用户。只有在步骤t时被激活的用户才会影响到其在步骤t+1时的邻居，且在一个网络中(如G(1))用户v在网络内关系i和网络间关系j下的激活概率分别为。

通过对网络内部和网络间的各种关系进行聚合，可以得到v(1)的整体激活概率



随后作者给出了基于贪心思想的解决AHI问题的算法：



## 6.2 Discover Tipping Users For Cross Network Influence

在文献[16]中，作者认为如今的用户通常同时参与多个在线社交网络，那些加入Facebook的用户也在使用其他网络，比如Twitter，Foursquare和Instagram。如果想要研究Facebook网络中影响力问题的话，也要研究影响力从上述其他网络中的传入。为了更好的描述作者的想法，作者首先给出了如下定义：

reduced network：给一个异构网络G=(V,E)和一组锚用户Z，缩减网络为：也就是把锚用户删掉再把与锚用户相连的边删掉。

Reduced Partially Aligned Networks：给一对部分对其的异构网络H=(G(s),G(t),L)，一个用户集合Z，缩减的部分对其网络为 其中是缩减网络，LZ是与Z中用户相连的锚用户的集合。

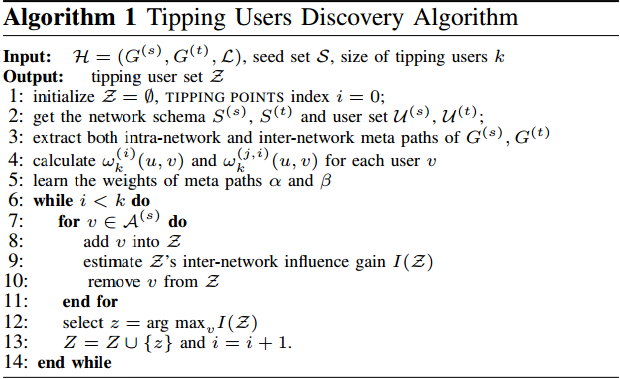
influence function：给一对部分对其的异构网络H=(G(s),G(t),L)和一个种子用户集合，用 表示影响函数，它表示从源网络中选的种子可以在目标网络中影响到的用户数。

Cross Network Influence Gain：给一对部分对其的异构网络H=(G(s),G(t),L)和一个种子用户集合S，一组锚用户Z，是在G(t)中可以影响到的用户数目，定义用户集Z的跨网络影响增益为二者的差。

Tipping users：给一对部分对其的异构网络H=(G(s),G(t),L)，一个种子用户集S，tipping users就是一组锚用户，它可以使得在源网络中的跨网络影响增益最大。

The TURN problem：(Tipping Users for cRoss Network influencing):给一对部分对其的异构网络H=(G(s),G(t),L)，一个种子用户集S，一个预先定义好的数k，TURN问题的目标是在源网络中找到k个Tipping users，可以使得cross-network influence gain最大。

这篇文献中作者是把种子集合作为输入，把tipping user作为一个输出，其算法流程如下：



## 6.3 local experts finding Across Multiple social networks

在文献[17]中，作者认为本地的专家可以在很多应用中发挥重要作用，如解决地方信息查询、社会事件安排等。事实上，雅虎最近的一项调查显示，人们更喜欢向当地的专家学习，因为他们熟悉社区，有第一手的经验。

作者对部分对齐异构网络中的本地专家发现问题做出来如下定义：给定一个部分对齐的异构社交网络，给定一个查询q=<t(q),l(q),k>，PAHLEF的目标是找到一个由k个候选人组成的集合，这些候选人在查询主题t(q)和地理位置l(q)中有很高的权威值。

为了解决这个问题，作者提出了KTMSNs方法。KTMSNs由数据准备阶段、社会拓扑感知反向索引构建、知识衰减转移和权衡讨论组成。

# 7. 参考文献

1. Li, Y., Fan, J., Wang, Y., & Tan, K.-L. (2018). Influence Maximization on Social Graphs: A Survey. IEEE Transactions on Knowledge and Data Engineering, 1–1. doi:10.1109/tkde.2018.2807843
2. Zhu, W., Yang, W., Xuan, S., Man, D., Wang, W., & Du, X. (2018). Location-Aware Influence Blocking Maximization in Social Networks. IEEE Access, 1–1. doi:10.1109/access.2018.2876141
3. Liqing, Q., Jinfeng, Y., Xin, F., Wei, J., & Wenwen, G. (2019). Analysis of influence maximization in temporal social networks. IEEE Access, 1–1. doi:10.1109/access.2019.2894155
4. Zhan, Q., Zhang, J., Wang, S., Yu, P. S., & Xie, J. (2015). Influence Maximization Across Partially Aligned Heterogenous Social Networks. Lecture Notes in Computer Science, 58–69. doi:10.1007/978-3-319-18038-0\_5
5. Chen, W., Wang, C., & Wang, Y. (2010). Scalable influence maximization for prevalent viral marketing in large-scale social networks. Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD ’10. doi:10.1145/1835804.1835934
6. Zhou, C., Zhang, P., Zang, W., & Guo, L. (2015). On the Upper Bounds of Spread for Greedy Algorithms in Social Network Influence Maximization. IEEE Transactions on Knowledge and Data Engineering, 27(10), 2770–2783. doi:10.1109/tkde.2015.2419659
7. Goyal, Amit, Wei Lu, and Laks VS Lakshmanan. "Simpath: An efficient algorithm for influence maximization under the linear threshold model." 2011 IEEE 11th international conference on data mining. IEEE, 2011.
8. Tang, Y., Xiao, X., & Shi, Y. (2014). Influence maximization. Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data - SIGMOD ’14. doi:10.1145/2588555.2593670
9. Liu, Bo, et al. "Time constrained influence maximization in social networks." 2012 IEEE 12th international conference on data mining. IEEE, 2012.
10. Kim, Jinha, Wonyeol Lee, and Hwanjo Yu. "CT-IC: Continuously activated and time-restricted independent cascade model for viral marketing." Knowledge-Based Systems 62 (2014): 57-68.
11. Gomez-Rodriguez, Manuel, et al. "Influence estimation and maximization in continuous-time diffusion networks." ACM Transactions on Information Systems (TOIS) 34.2 (2016): 9.
12. Mohammadi A, Saraee M, Mirzaei A. Time-sensitive influence maximization in social networks[J]. Journal of Information Science, 2015, 41(6): 765-778.
13. Chen, Wei, Wei Lu, and Ning Zhang. "Time-critical influence maximization in social networks with time-delayed diffusion process." Twenty-Sixth AAAI Conference on Artificial Intelligence. 2012.
14. Li H, Bhowmick S S, Cui J, et al. Time is what prevents everything from happening at once: Propagation time-conscious influence maximization[J]. arXiv preprint arXiv:1705.10977, 2017.
15. Zhan Q, Zhang J, Wang S, et al. Influence maximization across partially aligned heterogenous social networks[C]//Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, 2015: 58-69.
16. Zhan Q, Zhang J, Philip S Y, et al. Discover tipping users for cross network influencing[C]//2016 IEEE 17th International Conference on Information Reuse and Integration (IRI). IEEE, 2016: 67-76.
17. Ma Y, Yuan Y, Wang G, et al. Local Experts Finding Across Multiple Social Networks[C]//International Conference on Database Systems for Advanced Applications. Springer, Cham, 2019: 536-554.