**Time-Critical Influence Maximization in Social Networks with Time-Delayed Diffusion Process**

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.

**1.introduction**

最近，Facebook、Twitter和谷歌+等在线社交网站的迅速流行为大规模病毒式营销活动提供了巨大的机会。

然而，大多研究都没有考虑一个很重要的因素那就是时间。

首先，影响从一个人传播到另一个人可能会导致一定的时间延迟，这从最近统计物理学的研究中可以明显看出。第二，影响力的传播对时间是有严格约束的。

作者通过一个时间限制来实现对传播时间的约束，作者还提出了一个新的模型Independent Cascade model with Meeting events（IC-M）。除此之外，作者在现有的MIA算法之上又改进了两种启发式算法MIA-M和MIA-C。

**2. Influence Maximization with Deadline and Meeting Events and Its Properties**

**2.1 model and problem definition**

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。

对于相遇事件可以考虑为一下几种情形。例如，Facebook上的一个用户u在她朋友v的墙上发了一条消息，可以被认为是一个相遇事件，不同对的朋友在彼此的墙上交流信息的频率可能不同，这可以通过见面概率来反映。

可以看出传统的IC模型是IC-M模型中的一个特例，将IC-M中的相遇概率m都置1，就是IC模型。需要注意的是，如果我们不考虑时间的限制，那么相遇概率就毫无意义。一旦考虑了时间限制，那么相遇概率就变得更加重要了。

作者使用来表示时间限制。带时间限制的时序严格的IM问题的目标是寻找一个包含k个节点的种子集合S使得激活的种子数目最多。

**2.2 IC-M模型的性质**

IC-M模型下的影响函数是单调的和子模的

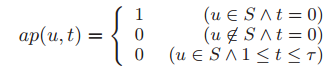
**3. Computing Influence in Arborescences**

在本节中，作者提出了一种动态规划算法，它可以精确地计算树结构中的影响扩散。in-arborescence是一种有向树，所有的边都指向根。给一个图G=(V,E)，一个影响函数p，一个遇见函数m，考虑一个in-arborescence为A=(VA,EA),根节点在v。作者认为能影响到v的节点只能是在A中。作者还认为一定有一个种子节点是在A中，否则的话A中的节点都不能被激活。

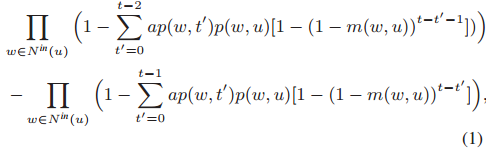
作者用ap(u,t)表示节点u在第t步时的激活概率。由于u在不同步骤被激活的事件是互斥的，u在步骤结束时变得活跃的概率为，影响传播公式为。

定理3：在树A中的任意节点u，在任意时刻，u的激活概率ap(u,t)可以递归的计算为

当u属于S或者t=0时



当u不属于S并且t属于[1,,,,t]时，



是节点u在A中的入邻居的集合。

由公式1给出的递归可以通过动态规划来实现，从叶子到根遍历。

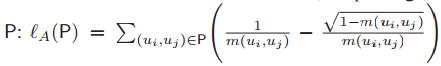
为了减少计算量，可以在实现中应用一些优化。作者使用Path(u)表示在树A中种子集合中的节点到节点u的最短路径，所以我们可以只计算，在这之前u是不可能被影响到的，也就是说ap(u,t)=0。

**4** **MIA Algorithms for IC-M**

贪心近似算法由于缺乏有效计算一般图中影响扩散的方法(第2节)，在实际应用中效率低下。为了克服这种低效率，我们提出了两种基于mia的启发式算法。第一个算法是MIA-M，利用定理3中的动态规划计算种子的精确影响。第二个是MIA-C算法，此算法首先将相遇概率融合到传播概率中。两种算法都首先为图中的每个节点构造一个最大影响树序(MIIA)，计算通过这些MIIA传播的影响，以近似原始网络中的影响。

**4.1 MIA-M算法**

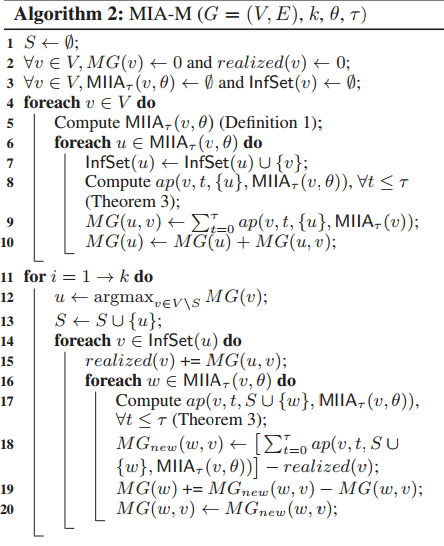
在描述算法之前，我们先介绍一些必要的符号。对于一组节点u,v，P(u,v)代表在图G中从u到v的所有路径的集合。给定一条u到v的路径，他的传播概率为：。随后作者定义了从u到v的最大影响路径MIP(u,v)为所有路径中概率最大的那个。如果u，v之间没有路径，则MIP（u，v）=空。此外，对于每个u,v最多需要一个MIP(u,v)。为了计算MIP，如果将传播概率P(u,v)转化为变权重logp(u,v)。计算MIP(u,v)就可以等价为找两点之间的最短路径，可以使用迪杰斯特拉算法。

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：



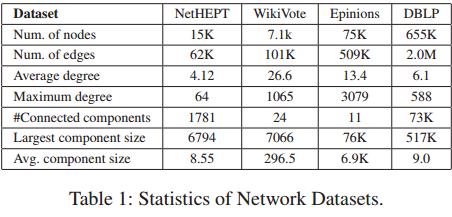
4.2 MIA-C

现在来讨论第二个算法，具有转换传播概率的MIA（MIA-C）他包含两个步骤。首先，对于任意一条边(u,v)，我们估算一个转换传播概率pc(u,v)，它包含相遇概率m，影响概率p，和截止时间。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.实验**

**数据集：**



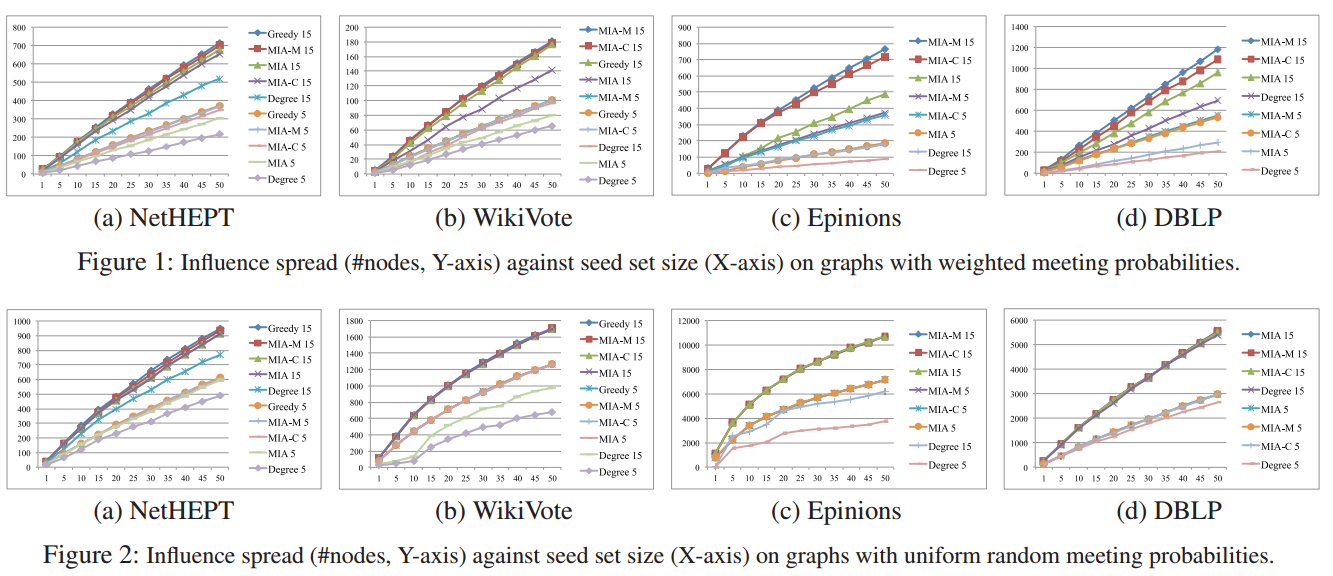
**参数**：p(u,v)的概率为入度分之一。

这里c是一个平滑常数，我们选择它为5。

对比算法：MIA-M,MIA-C,CELF,MC,Degree,MIA。

**5.1实验结果：**

种子集合的质量也就是影响传播范围



运行时间：

