



Computational Intelligence in Electrical Engineering
Vol. 13, No. 3, 2022
Research Paper

An approximate algorithm for maximizing modularity by estimating the domain of influence

Seyfollah Soleimani ¹, Rouhollah Javadpour Boroujeni ²

¹ Department of Computer Engineering, Faculty of Engineering, Arak University, Arak, Iran

² Department of Computer Engineering, Faculty of Engineering, Arak University, Arak, Iran

Abstract:

As social networks grow, they become more and more complex, and analyzing them becomes complicated. One way to reduce this complexity is to divide the network into subnets, which are also called communities. Dividing social networks into desirable communities can help analysts and experts to understand the behavior and function of the networks. Community detection in networks is a challenging topic in network science and various methods have been proposed for that. Modularity maximization is one of the state-of-the-art methods suggested for community detection. Modularity maximization is an NP-hard problem meaning that no polynomial-time algorithm exists that could solve the problem optimally unless $P=NP$. A group of approaches that could solve such problems is approximate algorithms. Identifying the influential nodes has many important applications in social networks. This technique could also be used in community detection. To maximize modularity, in this paper, we propose approximate algorithms based on identifying the influential nodes and their influence domain. We used the concept of scale-free networks to prove the approximate factor. Experiments on real-world networks show that the proposed algorithm can compete with the state-of-the-art methods of community detection algorithms.

Keywords: Approximate algorithm, Community detection, Reverse Influence Sampling (RIS) Framework, Social networks, Influential nodes, Modularity.



This is an open access article under the CC BY-NC-ND/4.0/ License (<https://creativecommons.org/licenses/by-nc-nd/4.0/>).



<http://dx.doi.org/10.22108/isee.2021.120798.1315>

یک الگوریتم تقریب برای بیشینه‌سازی ماژولاریتی به وسیله تخمین حوزه نفوذ

سیف‌اله سلیمانی^{*}، روح‌الله جوادپور بروجنی^۲

۱- استادیار، گروه مهندسی کامپیوتر- دانشکده مهندسی - دانشگاه اراک- اراک- ایران

s-soleimani@araku.ac.ir

۲- دانشجوی دکتری، گروه مهندسی کامپیوتر- دانشکده مهندسی- دانشگاه اراک- اراک- ایران

r-javadpour@phd.araku.ac.ir

چکیده: با رشد شبکه‌های اجتماعی، این شبکه‌ها هر روز بزرگ و بزرگ‌تر می‌شوند و تحلیل آنها به مراتب پیچیده‌تر می‌شود. برای سادگی تحلیل شبکه‌های اجتماعی می‌توان آنها را به مجموعه‌ای از اجتماعات مختلف تقسیم کرد. این کار، تحلیلگران و کارشناسان را در درک رفتار و عملکرد اینگونه شبکه‌ها یاری می‌دهد. روش‌های مختلفی برای تشخیص اجتماعات در شبکه‌ها ارائه شده‌اند. بیشینه‌سازی ماژولاریتی، یکی از روش‌های مدرن و مناسب برای تشخیص اجتماع است. بیشینه‌سازی ماژولاریتی یک مسئله NP-hard است؛ به این معنی که هیچ الگوریتم چندجمله‌ای برای حل این مسئله وجود ندارد؛ مگر اینکه $P=NP$ باشد. یک دسته از روش‌ها برای حل اینگونه مسائل، الگوریتم‌های تقریب است. شناسایی گره‌های پرنفوذ، کاربردهای زیادی در شبکه‌های اجتماعی دارد. این روش می‌تواند برای تشخیص اجتماع نیز به کار رود. در این مقاله، الگوریتم‌های تقریبی برای بیشینه‌سازی ماژولاریتی براساس شناسایی گره‌های پرنفوذ و دامنه نفوذشان پیشنهاد می‌شود. همچنین، از مفاهیم شبکه‌های مستقل از مقیاس برای اثبات نرخ تقریب استفاده می‌شود. آزمایش‌ها روی شبکه‌های واقعی نشان می‌دهند الگوریتم پیشنهادی قابل رقابت با روش‌های مدرن تشخیص اجتماع است.

واژه‌های کلیدی: الگوریتم تقریب، تشخیص اجتماع، چارچوب نمونه‌گیری نفوذ معکوس (RIS)، شبکه‌های اجتماعی،

گره‌های پرنفوذ، ماژولاریتی

۱- مقدمه

روزافزون شبکه‌های اجتماعی و گسترده‌تر شدنشان، تحلیل آنها نیز به مراتب پیچیده‌تر می‌شود. یک راه‌حل مناسب برای غلبه بر پیچیدگی یک مسئله، تقسیم آن به زیرمسئله‌های کوچک‌تر است. تحلیل شبکه‌ها و به‌طور خاص، تحلیل شبکه‌های اجتماعی نیز از همین قاعده پیروی می‌کند و برای غلبه بر پیچیدگی، می‌توان آن را به زیرشبکه‌های کوچک‌تر به نام اجتماع تقسیم کرد. تشخیص اجتماعات در شبکه‌ها یک موضوع چالشی در یک دهه گذشته بوده و مطالعات زیادی درباره این موضوع انجام شده است. مطالعات جامعه‌شناختی نشان می‌دهد بسیاری از رفتارها و تصمیمات اشخاص، از افراد بانفوذی تأثیر می‌گیرد که شخص به نحوی از طریق اجتماعات مختلف با آنها ارتباط دارد. در دنیای مجازی نیز بسیاری از رفتارها و تصمیمات اشخاص، از

به تازگی با سهولت هرچه بیشتر دسترسی به اینترنت و گسترش شبکه‌های اجتماعی مجازی، افراد بیشتری در حال پیوستن به این گونه شبکه‌ها هستند. در این شبکه‌ها اطلاعات زیادی که انعکاسی از دنیای واقعی‌اند، بین افراد ردوبدل می‌شوند. با بررسی فرآیند انتشار این اطلاعات، اطلاعات پنهان زیادی استخراج می‌شود [۹]. با رشد

^۱ تاریخ ارسال مقاله: ۱۳۹۸/۱۰/۰۸

تاریخ پذیرش مقاله: ۱۴۰۰/۰۲/۰۵

نام نویسنده مسئول: سیف‌اله سلیمانی

نشانی نویسنده مسئول: گروه مهندسی کامپیوتر- دانشکده

مهندسی- دانشگاه اراک - اراک- ایران

می‌شوند. به‌طور کلی زمانی که با کلان داده‌ها روبه‌رو هستیم، دو رویکرد برای مسئله تشخیص اجتماعات وجود دارد؛ یک رویکرد، مدیریت تشخیص سطح به سطح (تشخیص اجتماع چند سطحی) و دیگری رویکرد کاهش گراف است [۱۱].

هدف اصلی در الگوریتم‌های مبتنی بر مفهوم چندسطحی، ایجاد سلسله‌مراتبی از مسئله است. درحقیقت اندازه گراف اولیه، سطح به سطح با درهم‌آمیختن گره‌ها و یال‌ها کاهش پیدا می‌کند. اینگونه الگوریتم‌ها شامل سه مرحله درشت‌سازی، تفکیک، بازگشت به حالت اولیه و پالایش‌اند. الگوریتم‌های زیادی بر مبنای این رویکرد پیشنهاد شده‌اند که به [3, 12-16] اشاره می‌شود.

در رویکرد کاهش گراف، ایده اصلی، کاهش اندازه گراف بدون از بین رفتن کیفیت راه‌حل یا تغییر ساختار گراف است. به این ترتیب، هزینه محاسبات روی گراف کاهش می‌یابد. یکی از روش‌های کاهش گراف، نمونه‌گیری^۳ است. در این روش‌ها یک زیرمجموعه از گره‌ها یا یال‌ها انتخاب می‌شود. این روش‌ها به سه دسته تقسیم می‌شوند: (الف) نمونه‌گیری گره مانند [۱۷] (ب) نمونه‌گیری یالی مانند [۱۸] (ج) نمونه‌گیری براساس پیمایش مانند [۱۹].

روش‌های تشخیص اجتماع براساس معیار کیفیت سنجی نیز دسته‌بندی می‌شوند. در منابع، آنها را به دو دسته الگوریتم‌ها براساس ماژولاریتی^۴ و بدون ماژولاریتی^۵ دسته‌بندی می‌کنند.

یک نوع از الگوریتم‌های بدون ماژولاریتی، خوشه‌بندی طیفی^۶ هستند (به‌طور مثال، [۲۰]). این الگوریتم‌ها براساس مفهوم تقسیم‌بندی گراف در زیرشبکه‌هایی به نام برش‌ها^۷ کار می‌کنند. هدف، مینیمم‌سازی تعداد برش‌های تولیدشده است. نوع دیگری از الگوریتم‌های بدون ماژولاریتی، الگوریتم‌های پیداکردن اجتماعات هم‌پوشان هستند؛ برای نمونه، یکی از الگوریتم‌های رایج COPRA [۲۱] است که از این تکنیک استفاده می‌کند.

همان‌طور که گفته شد الگوریتم‌های براساس ماژولاریتی، از یک معیار به نام ماژولاریتی برای ارزیابی

افراد پرنفوذ تأثیر می‌گیرد که به نحوی در اجتماعات مجازی مختلف با آنها در تعامل‌اند؛ درنتیجه، افراد پرنفوذ نقش مهمی در راستای تغییر رفتار دیگران دارند؛ بنابراین، شناسایی افراد پرنفوذ و تشخیص اجتماعات آنان در شبکه‌های اجتماعی اهمیت زیادی دارد.

شبکه‌های اجتماعی همانند سایر شبکه‌های پیچیده، ویژگی‌های ماژولاریتی سلسله‌مراتبی^۱ و مستقل از مقیاس^۲ دارند [۱, ۲]. ماژولاریتی به این معناست که شبکه‌ها عموماً شامل اجتماعات (یا ماژول‌ها) با ارتباطات داخلی چگال و ارتباطات خارجی خلوت‌اند. سلسله‌مراتبی به این معناست که درون هر یک از اجتماعات، اجتماعات دیگری وجود دارند. بیشتر روش‌های تشخیص اجتماع مشهور شامل نوعی بهینه‌سازی یک تابع کیفیت است. یکی از مشهورترین توابع کیفیت، Newman-Girvan modularity [۳] است که با وجود برخی اشکالات [۴, ۵] به‌عنوان تابع هدف برای تخمین کیفیت استحکام اجتماعات استفاده می‌شود. این تابع پراستفاده‌ترین و مشهورترین تابع کیفیت برای تشخیص اجتماع است [۶].

بیشینه‌سازی ماژولاریتی، یکی از روش‌های مدرن و مناسب برای تشخیص اجتماع است؛ البته بیشینه‌سازی ماژولاریتی یک مسئله NP-hard است [۷]؛ مگر اینکه $P=NP$ باشد؛ بدین معنی که هیچ الگوریتم شناخته‌شده‌ای وجود ندارد که در زمان چندجمله‌ای مسئله را حل کند. یک رویکرد برای حل مسائل NP-hard الگوریتم‌های تقریب است که کیفیت راه‌حل را ضمانت می‌کند. پیچیدگی زمانی یک الگوریتم تقریب لزوماً چندجمله‌ای است و آن را با میزان خطای بدترین حالت ممکن روی همه نمونه‌های مسئله ارزیابی می‌کنند [۸]. به یک الگوریتم برای حل یک مسئله بیشینه‌سازی، الگوریتم α -approximation گویند. اگر برای هر نمونه از مسئله، جواب، حداقل $1/\alpha$ مقدار بهینه باشد ($\alpha < 1$)، به α نرخ تقریب گفته می‌شود.

با رشد سریع شبکه‌های اجتماعی، داده‌های حجیم با سرعت زیادی تولید می‌شوند؛ از همین رو مسائل مرتبط با اینگونه شبکه‌ها با چالشی به نام کلان داده نیز روبه‌رو

بوده‌اند با استفاده از نمونه‌برداری، حداکثر انتشار نفوذ را با یک میزان خطای مشخص تقریب بزنند. دسته دوم، روش‌های مکاشفه‌ای هستند (مانند [۳۰]) که هرچند در بیشتر مواقع مقیاس‌پذیرند، قادر به ارائه هیچ کرانی برای بدترین حالت برای انتشار نفوذ نیستند. دسته سوم، راه‌حل‌های فرامکاشفه‌ای هستند که براساس تکنیک‌های محاسبات تکاملی کار می‌کنند (مانند روش [۳۱]). این الگوریتم‌ها نیز هیچ کرانی برای بدترین حالت انتشار نفوذ ارائه نمی‌دهند. دسته چهارم، راه‌حل‌ها براساس اجتماعات هستند (مانند [۳۲]) که الگوریتم‌های این دسته از شناسایی اجتماعات در شبکه اجتماعی به‌عنوان یک اقدام واسط برای کاهش اندازه مسئله در سطح اجتماع و بهبود مقیاس‌پذیری عمل می‌کنند. دسته آخر مواردی‌اند که در هیچ‌یک از دسته‌بندی‌های فوق قرار نمی‌گیرند.

در این مقاله، با استفاده از شناسایی گره‌های پرنفوذ، یک الگوریتم تقریب برای مسئله بیشینه‌سازی ماژولاریتی ارائه شده است. نرخ تقریب الگوریتم نیز با استفاده از مفاهیم شبکه‌های مستقل از مقیاس اثبات شده است. در شناسایی افراد پرنفوذ از رویکرد نمونه‌گیری استفاده می‌شود که این کار معمولاً با برخی اقدامات کیفی هدایت‌پذیر است و می‌توان از یکسری اطلاعات اضافی مانند ویژگی‌های گره‌ها بهره‌برد [۱۰]. پس رویکرد ما علاوه بر مقیاس‌پذیری، انعطاف‌پذیر نیز خواهد بود؛ به این معنا که می‌توان تقسیم‌بندی یک شبکه اجتماعی را براساس یکسری ویژگی‌های مربوط به گره‌ها هدایت کرد.

در بخش ۲ مفاهیم اولیه بیان شده‌اند. در بخش ۳، الگوریتم پیشنهادی برای شبکه‌های مستقل از مقیاس بدون جهت مطرح شده و سپس در بخش ۴ الگوریتم برای شبکه‌های جهت‌دار تعمیم داده شده است. در بخش ۵، نتایج آزمایشات روی چندین شبکه کوچک و شبکه‌های اجتماعی بزرگ نشان داده می‌شوند. در انتها در بخش ۶ نتیجه‌گیری شده است.

کیفیت تقسیم‌بندی استفاده می‌کنند. این معیار معمولاً با Q نشان داده می‌شود و الگوریتم به دنبال پیدا کردن یک تقسیم‌بندی است که Q را بیشینه کند. تکنیک حریمانه، یکی از روش‌ها براساس ماژولاریتی است که در [۱۳] ارائه شده است. در [۲۲] و [۲۳]، الگوریتم‌هایی براساس مدل‌های ریاضی برای بیشینه‌سازی ماژولاریتی با کیفیت بالا ارائه شده است که البته مقیاس‌پذیر نیستند. در چندین مقاله نیز از تکنیک تقریب برای حل مسئله استفاده شده است؛ به‌طور مثال، در [۲۵] نویسندگان دو الگوریتم تقریب برای شناسایی اجتماعات در شبکه‌های اجتماعی پویا پیشنهاد داده‌اند. Dinh و همکاران در [۶] الگوریتم‌هایی برای مسئله بیشینه‌سازی ماژولاریتی در شبکه‌های مستقل از مقیاس ارائه کرده‌اند. این الگوریتم‌ها براساس درجه گره‌ها کار می‌کنند و هر گره را براساس برخی قواعد به عضو، سرگروه^۱ و مدارگرد^۱ برچسب‌گذاری می‌کنند. سپس هر سرگروه، یک اجتماع تشکیل می‌دهد و دنبال‌کننده‌هایش یعنی عضوها و مدارگردها را به آن اجتماع انتساب می‌دهند. برخی نیز روش‌هایی پیشنهاد کرده‌اند که عملکرد روش‌های گذشته را بهبود می‌بخشد؛ مانند [۲۳، ۲۴] که در آنها از تکنیک وزن‌دهی مجدد برای بهبود تشخیص اجتماع براساس بیشینه‌سازی ماژولاریتی استفاده شده است.

همچنین، درباره مسئله بیشینه‌سازی نفوذ نیز کارهای زیادی صورت پذیرفته است. Banerge و همکاران در [۲۴] این روش‌ها را به ۵ دسته تقسیم کرده‌اند؛ دسته نخست، روش‌های تقریبی‌اند که یک کران برای بدترین حالت انتشار نفوذ ارائه می‌دهند. Kempe و همکاران [۲۵] نخستین کسانی بودند که یک الگوریتم حریمانه ساده با کران $(\epsilon - \frac{1}{e})$ پیشنهاد کردند و کارهای [۲۶] و [۲۷] در راستای بهبود آن گام برداشتند؛ البته بسیاری از این روش‌ها از مقیاس‌پذیر نبودن رنج می‌برند؛ یعنی با افزایش اندازه شبکه، زمان اجرا به‌طور فوق‌العاده‌ای زیاد می‌شود. یک رویکرد برای غلبه بر این چالش، نمونه‌گیری از شبکه است که در [۲۸] و [۲۹] استفاده شده است. این رویکرد براساس مجموعه‌های در دسترس معکوس تصادفی^{۱۱} (RR) قادر

۲- تعریف مفاهیم اولیه

در این بخش، ابتدا مسئله بیشینه‌سازی ماژولاریتی، معرفی و سپس تابع ماژولاریتی توصیف می‌شود و پس از آن، چارچوب RIS^{۱۳} و نحوه شناسایی گره‌های پرنفوذ به‌عنوان هسته اجتماع معرفی می‌شود. در انتهای این بخش نیز شبکه‌های مستقل از مقیاس معرفی می‌شوند.

برخی نمادهای استفاده‌شده در این مقاله در جدول (۱) آمده‌اند.

جدول (۱) خلاصه نمادها

نماد	مفهوم
G	گراف (شبکه) اجتماعی
V(E)	مجموعه گره‌ها (یال‌ها)
n(m)	تعداد گره‌ها (یال‌ها)
R _i	یک مجموعه معکوس نفوذ (RR)
R	مجموعه از مجموعه‌های معکوس نفوذ (مجموعه از RRها)
d _i ⁱⁿ d _i ^{out}	درجه‌های ورودی و خروجی گره i
C	مجموعه‌ای از اجتماعات
E _l	تعداد گره‌هایی که دو انتها آنها در اجتماع l است
D _l	مجموع درجات گره‌ها در اجتماع l
δ	حد آستانه پیمایش گراف
β	نرخ توزیع درجات
Q	ماژولاریتی

این مجموعه با $C = \{C_1, C_2, \dots, C_t\}$ نشان داده می‌شود. به هر یک از این زیرمجموعه‌ها (یعنی به هر یک از $C_i \subseteq V$) یک اجتماع گفته می‌شود. ماژولاریتی (Q) نسبت تعداد یال‌های هم‌نوع (یعنی یال‌های درون یک اجتماع) به کل یال‌های شبکه منهای امید همین نسبت در شبکه‌ای با تقسیم‌بندی یکسان متناهی با اتصالات تصادفی بین گره‌ها است. هدف از بیشینه‌سازی ماژولاریتی این است که یال‌های بیشتری در درون اجتماعات نسبت به زمانی باشد که یکسری اجتماعات تصادفی از گره‌های گراف وجود دارد. اگر تعداد گره‌های درون - اجتماع، بهتر از یک حالت تصادفی نباشد، مقدار Q نزدیک به صفر خواهد بود؛ در حالی که مقدار نزدیک به یک نشان‌دهنده شبکه‌هایی با ساختار اجتماع قوی است. به‌طور رسمی، ماژولاریتی مطابق رابطه‌ی (۱) تعریف می‌شود [۶]:

$$Q(C) = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{d_i d_j}{2m}) \varphi_{ij} \quad (1)$$

جایی که $\varphi_{ij} = 1$ است، اگر i و j در یک اجتماع باشد و در غیر این صورت، $\varphi_{ij} = 0$ است. هدف مسئله بیشینه‌سازی ماژولاریتی، یافتن یک تقسیم‌بندی از گره‌های گراف است که ماژولاریتی را بیشینه کند. ماژولاریتی مطابق رابطه‌ی (۲) تعریف می‌شود [۶]:

$$Q(C) = \sum_{l=1}^t \left(\frac{E_l}{m} - \frac{D_l^2}{4m^2} \right) \quad (2)$$

جایی که E_l تعداد یال‌هایی است که هر دو انتهای آن در اجتماع l است و D_l برابر حجم C_l یعنی مجموع درجات تمام گره‌ها در C_l است.

۲-۲- چارچوب نمونه‌گیری معکوس نفوذ

(Reverse Influence Sampling)

در این زیربخش، ابتدا چارچوب RIS برای پیدا کردن گره‌های پرنفوذ ارائه می‌شود. به همین منظور، در ابتدا یکی از مهم‌ترین مدل‌های انتشار یعنی مدل انتشار آبخاری - که در این پژوهش لحاظ شده است - معرفی و سپس اصول چارچوب RIS تعریف می‌شود.

۲-۱- تعریف بیشینه‌سازی ماژولاریتی

یک شبکه با گراف $G = (V, E)$ نشان داده می‌شود. تعداد گره‌ها n ($|V| = n$) و تعداد یال‌ها m است ($|E| = m$). همچنین، برای G یک ماتریس هم‌جواری به‌صورت $A = (A_{i,j})$ تعریف می‌شود؛ به طوری که اگر بین i و j یک یال وجود داشته باشد، $A_{i,j} = 1$ است؛ در غیر این صورت، $A_{i,j} = 0$ است. ضمناً درجه گره i با d_i نشان داده شد.

مفهوم ماژولاریتی طبق تعریف موجود در [۳]، یک معیار برای اندازه‌گیری کیفیت تقسیم‌بندی گره‌های یک گراف (V) به یک مجموعه از زیرمجموعه‌های مجزا از گره‌ها است که اجتماع این زیرمجموعه‌ها همان V است.

۱-۲-۲ مدل انتشار

میزان خطای پذیرفته شده ε و احتمال خطا $(1 - \delta)$ است که به تقریب (ε, δ) معروف است و با $\theta(\varepsilon, \delta)$ نمایش داده می‌شود.

تعریف ۲: یک الگوریتم تصادفی یک تقریب (ε, δ) برای مقدار V ارائه می‌دهد. اگر X که خروجی الگوریتم است، در رابطه $\Pr(|X - V| \leq \varepsilon V) \geq 1 - \delta$ صدق کند، تعداد نمونه‌هایی که این شرط را اکتفا می‌کند، حداقل برابر است با $\frac{\ln(\frac{2}{\delta})}{\varepsilon^2}$.

۳-۲- شبکه‌های مستقل از مقیاس

برای تحلیل کارایی الگوریتم، ابتدا لازم است گراف به صورت رسمی تعریف شود. با توجه به اینکه شبکه‌های اجتماعی مستقل از مقیاس‌اند، چنین شبکه‌هایی از توالی درجه توان پایه پیروی می‌کنند. به همین منظور، یک مدل از گراف تصادفی در نظر گرفته می‌شود که از توزیع درجه وابسته به دو مقدار مفروض α و β پیروی می‌کند.

فرض کنید h گره با درجه $d > 0$ وجود دارد، به طوری که رابطه $\log h = \alpha - \beta \log d$ را اکتفا کند. به عبارت دیگر، $h = \frac{e^\alpha}{k^\beta}$ ، $|\{v | \text{degree}(v) = d\}| = h$ به طوری که α لگاریتم تعداد گره‌ها از درجه یک است و β نرخ توزیع درجات است. واقعیت‌های زیر از این مدل استنتاج می‌شود:

(۱) بیشترین درجه گراف $e^{\frac{\alpha}{\beta}}$ است.

تعداد گره‌ها و یال‌های گراف از روابط (۳) و (۴) به دست می‌آید:

$$n = \sum_{k=1}^{\frac{\alpha}{\beta}} \frac{e^\alpha}{d^\beta} \approx \begin{cases} \zeta(\beta)e^\alpha & \text{if } \beta > 1 \\ \alpha e^\alpha & \text{if } \beta = 1 \\ \frac{e^\alpha}{1-\beta} & \text{if } \beta < 1 \end{cases} \quad (۳)$$

$$m = \frac{1}{2} \sum_{k=1}^{\frac{\alpha}{\beta}} d \frac{e^\alpha}{d^\beta} \approx \begin{cases} \frac{1}{2} \zeta(\beta - 1) e^\alpha & \text{if } \beta > 2 \\ \frac{1}{4} \alpha e^\alpha & \text{if } \beta = 2 \\ \frac{1}{2} \frac{e^{\frac{2\alpha}{\beta}}}{2 - \beta} & \text{if } \beta < 2 \end{cases} \quad (۴)$$

مدل‌های زیادی برای شبیه‌سازی انتشار نفوذ در گراف اجتماعی وجود دارد. در این مقاله، تنها بر مدل آبشار مستقل تمرکز شده است. همچنین، فرض می‌شود تنها گره‌های پرنفوذ در دوره صفر فعال‌اند و سعی در فعال کردن گره‌های همسایه در دوره‌های بعدی را دارند. فرآیند انتشار مدل به صورت زیر توصیف می‌شود.

مدل آبشار مستقل^۳ (IC) زمانی که یک گره u فعال می‌شود، در هر دوره به استثنای دوره صفر، تنها یک شانسی دارد که هر یک از همسایه‌های غیرفعال خود (به فرض v) را با احتمال موفقیت $p(u, v)$ فعال کند که نسبتی از وزن یال $w(u, v)$ است. درضمن، هر گره فعال تا انتهای فرآیند انتشار، فعال باقی می‌ماند.

۲-۲-۲ اصول RIS

Borgs و همکارانش در [۳۳] روش نمونه‌برداری معکوس نفوذ (RIS) را ارائه کرده‌اند. این روش، یک چشم‌انداز از نفوذ در گراف G را با ایجاد یک کلکسیون \mathcal{R} از مجموعه‌های دسترس‌پذیر معکوس تصادفی (به نام RR) به دست می‌آورد.

تعریف ۱: مجموعه‌های RR یا مجموعه‌های دسترس‌پذیر معکوس [۲۹]: فرض کنید گراف $G = (V, E, w)$ داده شده است، یک مجموعه تصادفی RR (که با R_j نشان داده می‌شود)، به این صورت از G ایجاد می‌شود: (۱) یک گره تصادفی (v) از V انتخاب می‌شود؛ (۲) یک گراف نمونه (g) از G تولید می‌شود و (۳) ایجاد مجموعه R_j به‌عنوان یک مجموعه از گره‌های g که می‌توانند به v دسترسی پیدا کنند.

بنابراین R_j مجموعه گره‌هایی است که می‌توانند بر v تأثیر بگذارند. اگر چندین مجموعه تصادفی RR تولید شود، احتمالاً گره‌های پرنفوذ به تناوب در این مجموعه‌ها ظاهر خواهند شد.

یک موضوع مهم در این چارچوب این است که مینیمم اندازه مجموعه \mathcal{R} یا به عبارتی، تعداد نمونه‌ها (یعنی تعداد R_j ها) چیست؟ پاسخ به این سؤال مستلزم دانستن کران

شناسایی می‌شود که جزء هیچ اجتماعی نیست و از آن به‌عنوان گره شروع‌کننده پیمایش گراف استفاده می‌شود. سپس گراف، حداکثر به فاصله ۲ از هسته اجتماع (گره پرنفوذ) پیمایش می‌شود (الگوریتم اول سطح در بخش ۳-۲ را ببینید). این دسته از گره‌های پیمایش شده با همان شماره اجتماع گره پرنفوذ هسته برچسب‌گذاری می‌شوند. این کار تا جایی ادامه می‌پذیرد که هیچ گره‌ای بدون برچسب شماره اجتماع وجود نداشته باشد.

در آخر الگوریتم یک، بدون خدشه وارد کردن به ضمانت تقریب یک بهینه‌سازی روی اجتماعات به دست آمده انجام می‌پذیرد؛ به این صورت که برای هر یک از اجتماعات به دست آمده یک گره انتزاعی در نظر گرفته می‌شود که درجه آن برابر با مجموع درجات تمام گره‌های درون آن اجتماع است و الگوریتم تا زمانی که بهبود در ماژولاریتی داشته باشد، به صورت بازگشتی اجرا می‌شود. سرانجام نیز یک متد جستجوی محلی برای افزایش ماژولاریتی کلی صورت می‌پذیرد.

Algorithm 1: Graph Traversal based on the Probability of Influence (GTPI)

input : Graph $G, n, 0 < \epsilon, \delta < 1$

output: list of communities ($ListCom$)

```

for  $i = 1$  to  $\max(n, \theta(\epsilon, \delta))$  {
  create an RR set ( $R_i$ ) for node  $i$ ;
   $\mathcal{R} = \mathcal{R} \cup R_i$ 
  for each node  $v$  in  $R_i$ 
     $list(v) = list(v) \cup \#R_i$ ;
}
for each  $v_i, v_j$  that be neighbors
   $w_{i,j} = \frac{list(i) \cap list(j)}{|list(i)| + |list(j)|}$ ;  $\mu = \text{Median of } (W)$ ;
 $C = \emptyset$ ;
 $F = V$ ;
 $comNo = 0$ ;
while ( $F \neq \emptyset$ )
{
   $i = \text{findInfluentialNode}(V, F, \mathcal{R})$ ;
  ListCom [ $i$ ] =  $comNo$ ;
   $C = C \cup i$ ;
  Neig =  $BFS(G, i, \mu, W)$ ;
   $C = C \cup \text{Neig}$ ;
  for each  $v$  in Neigh
    ListCom [ $v$ ] =  $comNo$ ;
     $F = F \setminus C$ ;
   $comNo ++$ ;
}
Perform post optimization on ListCom;
return ListCom;
```

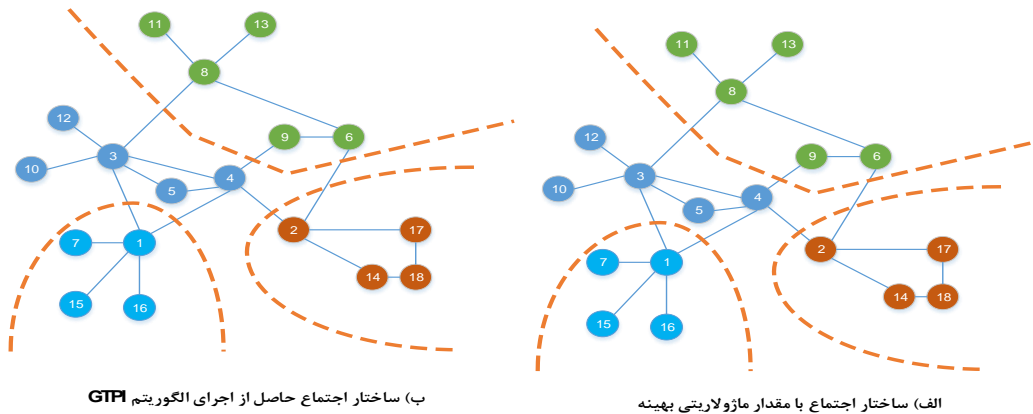
جایی که تابع ریمان زتا به صورت $\zeta(\beta) = \sum_{i=1}^{\infty} \frac{1}{i^\beta}$ است که برای $\beta > 1$ همگرا و برای $\beta \leq 1$ واگرا است. درخور ذکر است اگر از اعداد حقیقی به جای اعداد صحیح گردشده استفاده شود، ممکن است باعث خطا شود؛ اما این خطا در اثبات این مقاله، به اندازه کافی کوچک است.

۳- الگوریتم تقریب برای بیشینه‌سازی ماژولاریتی

در این بخش، یک الگوریتم با ضریب تقریب ثابت برای مسئله بیشینه‌سازی نفوذ برای شبکه‌های مستقل از مقیاس ارائه می‌شود. ایده اصلی این روش این است که اجتماعات اطراف گره‌های پرنفوذ شکل می‌گیرند؛ بنابراین، اگر بتوان گره‌های پرنفوذ را پیدا کرد و وزن یال‌های گراف را براساس تمایل هر گره برای ارتباط با همسایه‌هایش تقریب زد (یعنی احتمال نفوذ)، می‌توان بیشینه‌سازی ماژولاریتی را با پیمایش گراف اطراف گره‌های پرنفوذ تقریب زد.

۳-۱- پیمایش گراف براساس احتمال نفوذ

الگوریتم پیمایش گراف براساس احتمال نفوذ (GTPI)، همان‌گونه که در الگوریتم ۱ نشان داده شده، به این صورت است که ابتدا برای هر گره یک یا چند مجموعه دسترس‌پذیر معکوس (RR) ایجاد می‌کند، به طوری که تعداد زنجیره‌های RR حداقل $\theta(\epsilon, \delta)$ باشد. سپس براساس تخمین تمایل هر گره برای ارتباط با همسایه‌هایش (یعنی احتمال نفوذ) یال‌های گراف وزن‌دهی می‌شوند. میانگین وزن یال‌های گراف، همان‌گونه که در بخش ۳-۲ گفته شده است، به‌عنوان حد آستانه گراف برای پیمایش محاسبه می‌شود. همچنین، کلکسیون از مجموعه‌های RR، یعنی \mathcal{R} نیز برای شناسایی گره‌های پرنفوذی استفاده می‌شود که هنوز متعلق به هیچ اجتماعی نیستند. این گره‌های پرنفوذ به‌عنوان هسته اجتماعات استفاده می‌شوند. گره‌های پرنفوذ، بیشتر از دیگر گره‌ها در مجموعه \mathcal{R} تکرار می‌شوند. از این گره‌ها شروع به پیمایش گراف به منظور مشخص کردن حوزه نفوذ استفاده می‌شود؛ بنابراین، هر بار تأثیر‌گذارترین گره‌ای



شکل (۱): الف) یک ساختار اجتماع با مقدار ماژولاریتی بهینه که با CPLEX به دست آمده است. ب) یک ساختار اجتماع که با الگوریتم GTPI به دست آمده است (بهترین نتیجه حاصل از ده بار اجرای الگوریتم)

۳-۲- الگوریتم پیمایش اول سطح (BFS)

الگوریتم پیمایش اول سطح به این صورت است که ابتدا گره v به عنوان شروع کننده پیمایش و مقدار آستانه نفوذ δ را می گیرد و با توجه به ماتریس همجواری و ماتریس وزن دار W سعی می کند گرههایی را مشخص سازد که با گره v در یک اجتماع قرار می گیرند، به این صورت که ابتدا همسایه ها گره v را مشخص می کنند و سپس برای هر گره u در همسایگی v اگر وزن یال $w_{v,u}$ از حد آستانه δ بیشتر بود، گره u نیز به همان اجتماع گره v می پیوندد و همین روند برای همسایه های گره u تکرار می شود؛ یعنی در صورت برقراری شرط (وزن یال بیشتر از حد آستانه باشد)، همسایه های گره u نیز به همان اجتماع گره v می پیوندند.

Algorithm 2: BFS

input: Graph G , node v , threshold δ , adjacency matrix
output: list of node in community

```

 $N = neighbors(v);$ 
 $A = \emptyset;$ 
for each  $u$  in  $N$ 
{
    if  $w_{v,u} > \delta$  then
         $A = A \cup u;$ 
}
for each  $u$  in  $A$ 
{
     $N = neighbors(u);$ 
     $B = \emptyset;$ 
    for each  $x$  in  $N$ 
        if  $w_{u,x} > \delta$  then
             $B = B \cup x;$ 
}
return  $A \cup B;$ 
}
    
```

۳-۳- الگوریتم پیدا کردن گره پرنفوذ

پیدا کردن گره پرنفوذ یک الگوریتم حریصانه است. الگوریتم هر بار گره v را با بیشترین سود حاشیه ای، یعنی گره ای که بیشترین حضور را در مجموعه های RR دارد که قبلاً متعلق به هیچ اجتماعی نیستند را به عنوان گره پرنفوذ انتخاب می کند.

Algorithm 3: findingInfluentialNode

input: RR sets (R) , V , F

output: influential node

```

 $v = \arg \max_{\{v \in V\}} (Cov_R(V) - Cov_R(F))$ 
return  $v$ 
    
```

یک مثال از الگوریتم GTPI در شکل ۱ نشان داده شده است. الگوریتم ابتدا وزن هر یال را محاسبه می کند و به آن انتساب می دهد. وزن هر یال (i, j) به صورت $w(i, j) = \frac{1}{d_{in}(i)}$ محاسبه می شود؛ جایی که $d_{in}(i)$ نشان دهنده درجه ورودی است. همچنین، مقادیر $\epsilon = 0.1$ و $\delta = 1/n$ تنظیم شده اند. بر اساس این، حداقل تعداد نمونه ها $\frac{\ln(\frac{2}{\delta})}{\epsilon^2}$ است که هر یک از ۱۸ گره گراف به ترتیب و تناوب به عنوان شروع کننده زنجیره نفوذ (RR) انتخاب می شوند. با ایجاد هر نمونه RR، شماره آن (یعنی #RR) در لیست تک تک گره های موجود در مجموعه RR، اضافه می شود. این مجموعه از لیست ها برای وزندهی یال های گراف استفاده می شود. وزن هر یال (i, j) برابر است با

لیست هیچ گره همجواری نبوده‌اند و آنها را به اجتماع اضافه می‌کند.

با توجه به معادله ۵، با توجه به اینکه حد آستانه کمی کمتر از میانه در نظر گرفته شده است، وزن نصف یال‌ها بیشتر از این آستانه خواهد بود و عملاً دو انتهای بیش از نصف یال‌ها در یک اجتماع‌اند:

$$E_L \geq \sum_{i,j} e_{i,j} \mid w_{i,j} > \delta$$

$$E_L = |e_{i,j} \mid w_{i,j} > \delta \quad \text{if } \delta = \text{Median}(W) \quad (5)$$

$$\rightarrow E_L > \frac{1}{2} - \epsilon$$

حال نوبت آن رسیده است که یک کران بالا برای حجم درجه گررها محاسبه شود. فرض کنید اولین گره پرنفوذ i با درجه d_i باشد، با توجه به اینکه پیمایش به آستانه δ و درجه d_i بستگی دارد، تعداد کمی از همسایه‌های پیمایش شده به اجتماع اضافه می‌شوند. پس میانگین تعداد همسایه‌های هر گره برابر است با $\frac{2E}{n}$ و طبق معادله ۳ و ۴ با جای‌گذاری E و n داریم $\frac{\zeta(\beta-1)}{\zeta(\beta)} = B$. در واقع یک جستجوی اول سطح تا فاصله ۲ از گره پرنفوذ انجام می‌پذیرد، پس کران بالا برای حجم درجه گره برابر است با:

$$d_i + d_i \delta B + d_i \delta^2 B^2 = d_i (1 + \delta B + \delta^2 B^2)$$

$$\tau = 1 + \delta B + \delta^2 B^2 \quad (6)$$

$$\frac{d_i^2}{4m^2} \Rightarrow \frac{(\tau d_i)^2}{4m^2} \Rightarrow \frac{\tau d_i}{2m} = O(1)$$

و در نتیجه، با استفاده از معادلات ۵ و ۶ خواهیم داشت:

$$Q(C) > \frac{1}{2} - \epsilon - O(1) \quad (7)$$

۳-۴- پیچیدگی و تعیین حد آستانه

وزن دهی مجدد یال‌های گراف نقش مهمی در الگوریتم GTPI دارد. به همین منظور، لازم است نخست، مجموعه‌ای از RRها تولید شود و سپس اشتراک لیست (حضور) هر دو گره همسایه محاسبه شود؛ بنابراین، اگر متوسط اندازه هر مجموعه RR، c در نظر گرفته شود و $n = |V|$ باشد،

نسبت تعداد اشتراک listهای هر دو گره i و j (یعنی هر دو در چند زنجیره نفوذ به‌طور مشترک حضور دارند) به مجموع اندازه‌های لیست‌های $list(i)$ و $list(j)$ بعد از این کار، میانه وزن یال‌ها به‌عنوان آستانه پیمایش استفاده می‌شود. اکنون تا زمانی که همه یال‌ها برچسب اجتماع نخورده‌اند، گره‌ای با بیشترین نفوذ انتخاب می‌شود که جزء هیچ اجتماعی نیست (برای مثال، گره شماره ۲) و سپس به‌صورت پیمایش اول سطح تا فاصله دو از گره منبع (گره با نفوذ)، به شرطی که وزن یال بین دو گره بیشتر از حد آستانه باشد، پیمایش می‌شود. هر گره‌ای که پیمایش شد، با همان برچسب اجتماع گره منبع برچسب‌گذاری می‌شود.

به سبب آنکه الگوریتم برای گراف شکل ۱ به‌صورت تصادفی اجرا می‌شود، الگوریتم ۱۰ بار اجرا شده که بهترین ماژولاریتی ۰٫۴۳۶۶، برابر با ماژولاریتی بهینه است و بدترین حالت ۰٫۲۲۴۳ است که بیش از نیمی از ماژولاریتی بهینه است. همان‌طور که در شکل ۱ ب مشاهده می‌شود، نتیجه تقسیم‌بندی گراف برابر با مقدار بهینه است.

تئوری ۱: برای شبکه مستقل از مقیاس با $\beta > 1$ ، ماژولاریتی ساختار اجتماع که با الگوریتم GTPI به دست می‌آید، حداقل $\frac{1}{2} - \epsilon$ خواهد بود؛ جایی که $\epsilon > 0$ یک ثابت کوچک دلخواه باشد.

اثبات: بر طبق تعریف ماژولاریتی، ابتدا باید یک کران پایین برای قسمت اول و یک کران بالا برای قسمت دوم معادله (۲) به دست آید. به همین منظور، یک کران پایین برای تعداد یال‌هایی که هر دو انتهای آن در یک اجتماع است و با E_L نشان داده می‌شود و یک کران بالا برای مجموع درجه گررها داخل هر اجتماع t ارائه می‌شود.

نخست برای یک کران پایین E_L لازم است به دنبال یال‌هایی در اجتماعات بود که هر دو انتهای آن در یک اجتماع باشند. الگوریتم GTPI به این صورت است که برای هر اجتماع، گره پرنفوذی به‌عنوان هسته اجتماع پیدا می‌شود که قبلاً متعلق به هیچ اجتماعی نباشد. سپس گراف به‌صورت اول سطح با شروع از آن گره پرنفوذ به شرط آنکه وزن یال از δ بیشتر باشد، شروع به پیمایش می‌کند، سپس به سمت گرهای همجواری می‌رود که تا کنون متعلق به

اصلاح می‌شود. اکنون در گراف جهت‌دار احتمال وجود یک یال از گره i به گره j برابر با $\frac{d_j^{in} d_i^{out}}{m}$ خواهد بود. بدین ترتیب، معادله ۱ به صورت معادله ۱۰ اصلاح می‌شود:

$$Q(C) = \frac{1}{m} \sum_{i,j} \left(A_{i,j} - \frac{d_j^{in} d_i^{out}}{m} \right) \varphi_{i,j} \quad (10)$$

الگوریتم تقریب بیشینه‌سازی ماژولاریتی برای گراف‌های جهت‌دار به نام DGTPPI اساساً شبیه به الگوریتم GTPI است، با این تفاوت مهم که بین $w_{j,i}$ و $w_{i,j}$ تمایز در نظر گرفته می‌شود.

بنابراین، ماژولاریتی تقسیم‌بندی گراف به صورت معادله ۱۱ محاسبه می‌شود:

$$Q_F = \sum_{u \in F} \left(\frac{E_u}{2m} - \frac{d_u^{in} d_u^{out}}{4m^2} \right) \quad (11)$$

جایی که E_u برابر است با تعداد یال‌های جهت‌دار که دو انتهای آنها در C_u است. برای اثبات ضمانت تقریب، نخست، یک کران پایین برای $\sum E_u$ ارائه می‌شود. با توجه به اینکه هر یال براساس δ پیمایش می‌شود و تعیین این مقدار آستانه در قسمت ۳-۵ توضیح داده شد، برای آنکه بیشترین مقدار را داشته باشیم، این مقدار آستانه برابر $\frac{1}{2}$ در نظر گرفته می‌شود و بنابراین، $E_u = \frac{1}{2}m$ و بنابراین، کران پایین برای آن برابر است با $\frac{1}{4} - \varepsilon$.
اما برای کران بالا برای قسمت دوم معادله خواهیم داشت:

$$\sum_{u \in F} \frac{d_u^{in} d_u^{out}}{4m^2} \leq d_u^{out} + d_u^{in} \delta B + d_u^{in} \delta^2 B^2$$

$$\Delta = \max_{i=1..n} \{d_i^{out}, d_i^{in}\} \text{ and } \tau = 1 + \delta B + \delta^2 B^2 \quad (12)$$

$$d_u^{out} < \Delta(1 + \delta B + \delta^2 B^2) = \Delta \tau$$

$$\frac{\Delta \tau \cdot d_u^{in}}{4m^2} = \frac{\Delta \tau}{2m} \cdot \frac{d_u^{in}}{2m} = O(1)$$

در نتیجه:

$$\frac{1}{4} - \varepsilon - O(1). \quad (13)$$

پیچیدگی الگوریتم $O(cn)$ برای فاز وزن‌دهی و $O(|E|)$ برای فاز پیمایش خواهد بود.

۳-۵- تعیین حد آستانه

آستانه δ نقش مهمی در الگوریتم GTPI دارد و طبق تعریف ماژولاریتی در معادله ۲، E_l و d_l رابطه مستقیمی با حد آستانه دارند.

برای تعیین حد آستانه، فرض کنید پیمایش گراف از گره i شروع می‌شود و طبق الگوریتم یال‌هایی که وزن آنها از δ بیشتر است، پیمایش می‌شوند. پس تعداد δm از یال‌ها دو انتهای آنها در یک اجتماع قرار می‌گیرد. همچنین، C_l دو برابر یال‌هایی است که داخل اجتماعات قرار دارند؛ بنابراین، $D_l = 2\delta m$ ؛ در نتیجه، از معادله ۲ خواهیم داشت:

$$\sum \frac{\delta m}{m} - \frac{4\delta^2 m^2}{4m^2} = \sum \delta - \delta^2 \quad (8)$$

رابطه ۸ زمانی بیشینه است که مشتق آن برابر صفر شود؛ بنابراین، خواهیم داشت:

$$1 - 2\delta = 0 \rightarrow \delta = \frac{1}{2}. \quad (9)$$

۴- بیشینه‌سازی ماژولاریتی در شبکه‌های

جهت‌دار

بسیاری از شبکه‌های پیچیده ذات جهت‌دار دارند، جهت یال‌ها اطلاعات دقیق‌تری از روابط بین موجودیت‌های شبکه ارائه می‌دهد. به همین منظور، لازم است جهت یال‌ها در تشخیص اجتماع در نظر گرفته شود.

در اینجا الگوریتم GTPI برای شبکه‌های جهت‌دار تعمیم داده شده است. برای انجام این کار، ابتدا برخی تعاریف ارائه می‌شوند. برای یک گراف جهت‌دار مفروض، درجه ورودی و درجه خروجی گره i به ترتیب با d_i^{in} و d_i^{out} نشان داده می‌شود. همچنین، مجموعه همسایه‌های ورودی با $N^-(i) = \{j | (j, i) \in E\}$ و مجموعه همسایه‌های خروجی با $N^+(i) = \{j | (j, i) \in E\}$ نشان داده می‌شود. با تعریف $|N^-(i)| = d_i^{in}$ و $|N^+(i)| = d_i^{out}$ برای گراف‌های جهت‌دار

۵- نتایج آزمایش‌ها

در این قسمت، کارایی الگوریتم GTPI برای چندین شبکه پیچیده کوچک و بزرگ ارزیابی شده است.

۵-۱- شبکه‌های کوچک

نخست، الگوریتم GTPI با چندین روش بیشینه‌سازی ماژولاریتی روی چندین مورد شبکه مشهور با اندازه کوچک برای تشخیص اجتماع مقایسه می‌شود. نام دیتاست‌ها همراه با اندازه آنها در جدول ۲ آورده شده است. الگوریتم پیشنهادی با سه الگوریتم دیگر مقایسه شده که الگوریتم LDF [۶] به دلیل استفاده از رویکرد مشابه، یعنی تقریب‌زدن جواب بهینه و الگوریتم MCCA [۱۲] به لحاظ استفاده از رویکرد مدیریت تشخیص سطح به سطح برای غلبه بر چالش کلان داده (مقیاس‌پذیری) و الگوریتم (MILP) [۲۲] انتخاب شده‌اند که توانایی محاسبه ماژولاریتی بهینه را دارد.

تمام آزمایش‌های انجام شده روی PC با سیستم عامل لینوکس و با پردازشگر Intel Core i7, 7700 و 16GB RAM انجام شده است. همچنین، پیاده‌سازی الگوریتم‌ها با زبان C++ صورت پذیرفته است. مقادیر ماژولاریتی به‌دست آمده با الگوریتم‌های مختلف در جدول ۳ نشان داده شده‌اند.

۵-۲- شبکه‌های اجتماعی واقعی

آزمایش‌های بیشتر برای ارزیابی الگوریتم روی دید لحظه‌ای^۴ از چهار شبکه اجتماعی Foursq, Facebook و Twitter و Flickr انجام شده‌اند.

اندازه شبکه‌های اجتماعی به همراه ماژولاریتی الگوریتم‌ها MCCA, LDF and GTPI در جدول ۴ نشان داده شده است. درخور ذکر است الگوریتم MILP به لحاظ ناتوانی اجرا روی شبکه‌های بزرگ در نظر گرفته نشده است. هر سه الگوریتم مقیاس‌پذیر بوده و در زمان معقولی روی این دیتاست‌ها اجرا شده‌اند. الگوریتم GTPI کارایی بهتری روی شبکه‌های اجتماعی با پایه کمتر یعنی شبکه‌های Foursq and Twitter داشته است. هرچه پایه توان (β) کمتر باشد، گراف چگال‌تر خواهد بود. هرچه گراف چگال‌تر باشد، واریانس اندازه زنجیره‌های RR کمتر است؛ از این رو، دقت در تشخیص گره‌های بانفوذ بیشتر می‌شود. با تشخیص بهتر گره‌های بانفوذ، هسته اجتماعات بهتر تشخیص داده می‌شود و ماژولاریتی بهتری به دست خواهد آمد.

۶- نتیجه‌گیری

در این مقاله، الگوریتم‌های تقریبی برای بیشینه‌سازی ماژولاریتی براساس این ایده ارائه شده است که گره‌های پرنفوذ در هسته اجتماعات‌اند و با تعیین دامنه نفوذ آنها اجتماعات، تشخیص داده می‌شود

جدول ۲: ویژگی‌های دیتاست‌ها

شناسه	نام	تعداد گرهها (n)	تعداد یال‌ها (m)
1	Zachary's karate club	34	78
2	Dotphin's social network	62	159
3	Les Miserables	77	254
4	Books about US politics	105	441
5	American College Footbal	115	613
6	Electronic Circuit(s838)	512	819

جدول ۳: مقادیر ماژولاریتی به دست آمده از الگوریتم‌های [12] MCCA ، [6] LDF ، GTPI (بهترین و بدترین حالت) و مقادیر ماژولاریتی بهینه (MILP) [22] برای شبکه‌های کوچک

شناسه	n	LDF	MCCA	GTPI(worst)	GTPI(best)	ماژولاریتی بهینه (MILP)
1	34	0.4198	0.4198	0.2571	0.4198	0.4198
2	62	0.5179	0.5274	0.3814	0.5285	0.5285
3	77	0.5600	0.5532	0.4982	1.5600	0.5600
4	105	0.5257	0.5265	0.3694	0.5272	0.5272
5	115	0.6028	0.6014	0.5516	0.6046	0.6046
6	512	0.8159	0.8168	0.6120	0.8179	0.8194

جدول ۴: مقادیر ماژولاریتی به دست آمده از الگوریتم‌های [6] LDF [12] MCCA و GTPI

نام شبکه اجتماعی	تعداد گرهها	تعداد یالها	β	MCCA	LDF	GTPI
Foursq	45k	1664k	1.64	0.4498	0.4502	0.4512
Facebook	64k	906k	2.27	0.6467	0.6414	0.6218
Twitter	88k	2364	1.69	0.5491	0.5470	0.5507
Flickr	81k	5900k	2.21	0.5236	0.5215	0.5211

مراجع

- [1] A. L. Barabasi, Z. Dezso, E. Regan, S. H. Yook, Z. Oltvai, "Scale Free and Hierarchical Structures in Complex Networks," *Modeling of Complex Systems*, Vol. 661, pp. 1-16, April 2003.
- [2] S. M. Shekatkar, G. Ambika, "Complex networks with scale-free nature and hierarchical modularity," *The European Physical Journal B*, Vol. 88, No. 9, pp. 227, September 2015.
- [3] M. E. J. Newman, M. Girvan, "Finding and evaluating community structure in networks," *Phys Rev E*, Vol. 69, No.2, pp. 026113, March 2004.
- [4] B. H. Good, Y.A. de Montjoye, A Clauset, "Performance of modularity maximization in practical contexts," *Physical Review E*, Vol. 81, No. 4, pp. 046106, April 2010.
- [5] S. Fortunato, M. Barthélemy, "Resolution limit in community detection," *Proceedings of the National Academy of Sciences*, Vol. 104, No. 1, pp. 36, January 2007.
- [6] T. N. Dinh, M. T. Thai, "Community Detection in Scale-Free Networks: Approximation Algorithms for Maximizing Modularity," *IEEE Journal on Selected Areas in Communications*, Vol. 31, No. 6, pp. 997-1006, June 2013.
- [7] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hofer, Z. Nikoloski, et al., "On Modularity Clustering," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 20, No. 2, pp. 172-188, February 2008.
- [8] E. Coffman, M. R. Garey, D. Johnson, "Approximation Algorithms for NP-Hard Problems," *SIGACT News*, Vol. 28, No.2,

در این مقاله، از ماژولاریتی به عنوان معیار اندازه‌گیری کیفیت تقسیم‌بندی اجتماعات، استفاده و سعی شده است الگوریتم‌هایی به همراه ضمانت کارایی شان برای شبکه‌های مستقل از مقیاس ارائه شود. به این ترتیب که برای شبکه‌های مستقل از مقیاس بدون جهت با $B > 1$ الگوریتم تقریبی با ضریب تقریب $\epsilon - \frac{1}{2}$ و برای شبکه‌های جهت‌دار با $\epsilon - \frac{1}{4}$ ارائه شده است. مزیت روش پیشنهادی این است که چون از رویکرد نمونه‌گیری برای شناسایی گره‌های پرنفوذ به عنوان هسته اجتماع استفاده شده است و به دلیل اینکه در فرآیند نمونه‌گیری از اطلاعات اضافی مانند ویژگی‌های گره‌ها بهره برده می‌شود، پس رویکرد ما علاوه بر کارایی روی دیتاست‌های بزرگ، انعطاف‌پذیر نیز هست؛ به این معنا که قادر خواهد بود در تقسیم‌بندی شبکه اجتماعی از یکسری ویژگی‌های گره استفاده کند که این امر در کاربردهای مختلف و تحلیل‌های گوناگون کارایی دارد. برای کارهای آینده پیشنهاد می‌شود در کنار تقسیم‌بندی ساختار گراف بر تقسیم‌بندی براساس محتوای گره‌ها نیز تمرکز شود؛ زیرا الگوریتم‌های موجود که گراف را براساس محتوا تقسیم‌بندی می‌کنند، بیشتر با مشکل مقیاس‌ناپذیری روبه‌رو هستند و این مشکل با این راهکار تا حدی مرتفع می‌شود.

- 2015.
- [20] L. Hagen, A. B. Kahng, "New spectral methods for ratio cut partitioning and clustering," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 11, No. 9, pp. 1074-1085, September 1992.
- [21] S. Gregory, "An Algorithm to Find Overlapping Community Structure in Networks," *Knowledge Discovery in Databases*, pp. 91-102, September 2007.
- [22] E. Alinezhad, B. Teimourpour, M. M. Sepehri, M. Kargari, "Community detection in attributed networks considering both structural and attribute similarities: two mathematical programming approaches," *Neural Computing and Applications*, Vol. 32, pp. 3203-3220, February 2019.
- [23] G. Agarwal, D. Kempe, "Modularity-maximizing graph communities via mathematical programming," *The European Physical Journal B*, Vol. 66, No.3, pp.409-418, November 2008.
- [24] S. Banerjee, M. Jenamani, D.K. Pratihar, "A survey on influence maximization in a social network," *Knowledge and Information Systems*, Vol. 62, No. 9, pp. 3417-3455, March 2020.
- [25] D. Kempe, J. Kleinberg, É. Tardos, "Maximizing the Spread of Influence through a Social Network," *Proceedings of the ACM SIGKDD*, pp. 137-146, August 2003.
- [26] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. Vanbriesen, N. Glance, "Cost-effective outbreak detection in networks," *Proceedings of the ACM SIGKDD*, pp. 420-429, August 2007.
- [27] A. Goyal, W. Lu, L. Lakshmanan, "CELF++: Optimizing the greedy algorithm for influence maximization in social networks," *Proceedings of the 20th international conference companion on World wide web*, pp. 47-48, March 2011.
- [28] Y. Tang, X. Xiao, Y. Shi, "Influence Maximization: Near-Optimal Time Complexity Meets Practical Efficiency," *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 75-86, June 2014.
- [29] H. T. Nguyen, M. T. Thai, T. N. Dinh, "Stop-and-Stare: Optimal Sampling Algorithms for Viral Marketing in Billion-scale Networks", *Proceedings of the 2016 International Conference on Management of Data*, pp. 695-710, June 2016.
- [30] A. Goyal, W. Lu, L. Lakshmanan, "SIMPACT: An Efficient Algorithm for Influence Maximization under the Linear pp. 46-93, January 1997.
- [9] M. Li, X. Wang, K. Gao, S. Zhang, "A Survey on Information Diffusion in Online Social Networks: Models and Methods," *Information*, Vol. 8, No. 4, pp. 118, September 2017.
- [10] F. Menczer, S. Fortunato, C.A. Davis, "A First Course in Network Science," Cambridge University Press, Cambridge, February 2020.
- [11] M. Azaouzi, D. Rhouma, L. Romdhane, "Community detection in large-scale social networks: state-of-the-art and future directions, " *Social Network Analysis and Mining*, Vol. 9, May 2019.
- [12] D. Rhouma, L. Romdhane, "An efficient multilevel scheme for coarsening large scale social networks," *Applied Intelligence*, Vol. 48, March 2018.
- [13] V.D. Blondel, J.L. Guillaume, R. Lambiotte, E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, Vol. 2008, No. 10, pp. P10008, October 2008.
- [14] V. Satuluri, S. Parthasarathy, "Scalable graph clustering using stochastic flows: applications to community discovery," *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.737-746, June 2009.
- [15] L. Waltman, N. Eck, "A smart local moving algorithm for large-scale modularity-based community detection," *The European Physical Journal B*, Vol. 86, pp. 1-14, November 2013.
- [16] D. LaSalle, G. Karypis, "Multi-threaded modularity based graph clustering using the multilevel paradigm," *Journal of Parallel and Distributed Computing*, Vol. 76, pp. 66-80, February 2015.
- [17] J. Leskovec, C. Faloutsos, "Sampling from large graphs," *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 631-636, August 2006.
- [18] V. Krishnamurthy, M. Faloutsos, M. Chrobak, L. Lao, J-H. Cui, A. Percus, "Reducing Large Internet Topologies for Faster Simulations," *Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, pp. 328-341, May 2005.
- [19] Y. Ruan, D. Fuhry, J. Liang, Y. Wang, S. Parthasarathy, "Community Discovery: Simple and Scalable Approaches", *User Community Discovery*, pp. 23-54, October

- August 2016.
- [33] C. Borgs, M. Brautbar, J. Chayes, B. Lucier, "Maximizing social influence in nearly optimal time", Society for Industrial and Applied Mathematics, Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms, pp. 946-957, January 2014.
- [34] D. Aloise, S. Cafieri, G. Caporossi, P. Hansen, S. Perron, L. Liberti, " Column generation algorithms for exact modularity maximization in networks," Phys Rev E Stat Nonlin Soft Matter Phys , Vol. 82, pp. 046112, October 2010.
- Threshold Model, " 2011 IEEE 11th International Conference on Data Mining , pp. 211-220, January 2011.
- [31] M. Gong, C. Song, C. Duan, M. Lijia, B. Shen, "An Efficient Memetic Algorithm for Influence Maximization in Social Networks, " IEEE Computational Intelligence Magazine, Vol. 11, No. 3, pp.22-33, July 2016.
- [32] E. Bagheri, G. Dastghaibifard, A. Hamzeh, "An efficient and fast influence maximization algorithm based on community detection," 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), pp. 1636-1641,

-
- ¹ Hierarchical
 - ² Free scale
 - ³ Sampling
 - ⁴ Modularity based
 - ⁵ Non modularity based
 - ⁶ Spectral clustering
 - ⁷ Cuts
 - ⁸ member
 - ⁹ eader
 - ¹⁰ orbiter
 - ¹¹ random reverse reachable
 - ¹² reverse influence sampling
 - ¹³ ndependent Cascade
 - ¹⁴ Satisfy
 - ¹⁵ Snapshots

