مقاله پژوهشی

ارائه الگوریتم موازی و کارا به‌منظور شناسایی انجمن‌های همپوشان در شبکه‌های اجتماعی

مصطفی سبزه‌کار، شیما برادران‌نژاد، مهدی خزاعی‌پور و مهدی خرد

چکیده: شبکه‌های اجتماعی نه‌تنها به عنوان ابزاری برای ارتباطات، بلکه یکی از پتانسیل‌های مهم در کسب‌وکار و تجارت می‌باشند. یکی از مهم‌ترین مسائل تعریف‌شده در این حوزه، خوشه‌بندی گره‌ها و استخراج الگوهای مؤثر و مفید از آنهاست که به کشف انجمن معروف است. از چالش‌های مهم شناسایی انجمن در شبکه‌های اجتماعی می‌توان به حجم بسیار زیاد گره‌ها اشاره نمود که هر گونه تحلیلی بر روی آن را با مشکل مواجه می‌سازد. از دیگر چالش‌های پیش رو، اشتراک برخی از اعضای خوشه‌ها با یکدیگر می‌باشد که از آن به‌عنوان همپوشانی انجمن‌ها نام برده می‌شود. در چنین شبکه‌هایی هر گره می‌تواند به چند گروه تعلق پیدا کند. در نظر گرفتن همپوشانی بین انجمن‌ها به خصوص در شبکه‌های بزرگ، تشخیص و شناسایی انجمن را با مشکلات زیادی روبه‌رو می‌نمایند؛ از این رو در بیشتر پژوهش‌ها این مسئله نادیده گرفته می‌شود. در این مقاله، رویکردی به منظور رفع این مشکلات ارائه می‌شود. مرحله یافتن گره‌های تأثیرگذار شبکه که زمان‌برترین مرحله در الگوریتم پیشنهادی است، به‌صورت موازی انجام می‌شود و همچنین همپوشانی بین انجمن‌ها در نظر گرفته شده و تحلیل می‌گردد. نتایج حاصل از ارزیابی روش پیشنهادی در قیاس با روش‌های مورد مقایسه، حاکی از برتری آن در یکنواختی انجمن‌های کشف شده است.

کلیدواژه: **شبکه‌های اجتماعی، موازی‌سازی، کشف انجمن، انجمن‌های هم‌پوشان.**

۱- مقدمه

امروزه تقریباً می‌توان هر پدیده طبیعی را با تعریف مجموعه‌ای از موجودیت‌ها و در نظر گرفتن روابط بین آنها به‌صورت شبکه در نظر گرفت [۱]. شبکه‌های اجتماعی نمونه شناخته‌شده‌ای از یک شبکه هستند که عبارت است از ساختاری اجتماعی از مجموعه‌ای از گره‌ها به عنوان بازیگران اجتماعی که توسط یک یا چند نوع خاص از وابستگی متقابل به هم مرتبط هستند [۲]. شبکه‌های اجتماعی شبکه‌های دنیای واقعی هستند که ماهیتی فراگیر دارند. برخی از نمونه‌های معمولی از شبکه‌های اجتماعی عبارتند از شبکه‌های همکاری گروهی تحقیقاتی، شبکه‌های ارتباطی شرکت، شبکه‌های دوستی آنلاین، شبکه‌های مبتنی بر رفتار حیوانات و غیره. استفاده از شبکه‌های اجتماعی به دلیل کاربرد گسترده آن در دنیای واقعی، توجه تحقیقاتی قابل توجهی را به خود معطوف کرده است. با گسترش خدمات مبتنی بر اینترنت و افزایش حجم داده‌ها، شبکه‌های اجتماعی نیز به طرز چشم‌گیری در حال افزایش حجم و اندازه هستند [۳] و [۴]. به علاوه این شبکه‌ها امروزه به علت کمک به برقراری ارتباط بین افراد، بی‌گمان به محبوب‌ترین ابزار مبتنی بر اینترنت تبدیل شده‌اند [۵].

از مهم‌ترین مشخصه‌های شبکه‌های اجتماعی، ساختار انجمن‌های درون آنهاست. شناسایی ساختار انجمن یکی از موضوعات مهم و چالش‌برانگیز در شبکه‌های اجتماعی می‌باشد که هدف از آن، یافتن زیرگراف‌هایی است که از نظر داخلی متراکم بوده، اما از نظر خارجی به‌طور پراکنده با هم در ارتباط هستند [۶]. شناسایی انجمن‌ها اصولاً به عنوان مسائل خوشه‌بندی و گراف‌کاوی تعریف می‌شوند [۷]. انجمن‌ها اغلب نشان‌دهنده گروه‌های سازمان‌یافته از افراد با ویژگی‌ها، علایق یا روابط شخصی مشابه هستند و در نتیجه، شناسایی ساختار انجمن‌ها در شبکه‌های اجتماعی بسیار ضروری است [۸].

تشخیص انجمن‌ها اگرچه مسئله جدیدی نبوده و به عنوان یک مسئله خوشه‌بندی تعریف می‌شود، اما با چالش‌های متعددی نیز همراه است که به شرح ذیل می‌باشد:

- یکی از این چالش‌ها همپوشانی موجود بین انجمن‌هاست. بسیار طبیعی است که در دنیای واقعی هر گره از شبکه بتواند به چندین گروه تعلق پیدا کند. در نظر گرفتن این مسئله موجب تحمیل‌شدن حجم زیادی از محاسبات و پیچیده‌شدن راه‌حل‌های ارائه‌شده می‌گردد؛ به نحوی که در بسیاری از پژوهش‌ها از همپوشانی بین انجمن‌ها صرف نظر می‌شود [۹].
- شبکه اجتماعی ماهیتی پویا دارد که در طی زمان تغییر کرده و تکامل می‌یابد. ماهیت این تغییرات شامل چهار نوع عملیات مختلف یعنی ایجاد و حذف گره و ایجاد و حذف لبه تعریف می‌گردد. در پژوهش‌های گذشته، تشخیص انجمن بیشتر برای شبکه‌های ایستا مورد مطالعه قرار گرفته؛ اما امروزه به دلیل رشد اندازه شبکه و افزایش کاربران آنها، مطالعات بسیاری بر روی شبکه‌های پویا انجام می‌شود [۱۰].
- در شبکه‌های اجتماعی نظر افراد مختلف با یکدیگر یکسان نیست و این مسئله بایستی در تشخیص انجمن‌ها مد نظر قرار بگیرد. به طور مثال، چنانچه یک نظر توسط افراد مشهور یا بسیار نزدیک به فرد مورد نظر ارائه شده باشد، در تصمیم‌گیری در خصوص مسئله خاص تأثیر بیشتری می‌گذارد [۱۱]. بنابراین در پژوهش‌های اخیر بر روی

این مقاله در تاریخ ۱۹ مرداد ماه ۱۴۰۳ دریافت و در تاریخ ۲۸ بهمن ماه ۱۴۰۲ بازنگری شد.

مصطفی سبزه‌کار (نویسنده مسئول)، گروه مهندسی کامپیوتر، دانشگاه صنعتی بیرجند، بیرجند، ایران، (email: sabzekar@birjandut.ac.ir).

شیما برادران‌نژاد، گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد بیرجند، بیرجند، ایران، (email: shima.baradaran2005@gmail.com).

مهدی خزاعی‌پور، گروه مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد بیرجند، بیرجند، ایران، (email: mkhazaeipoor@iaubir.ac.ir).

مهدی خرد، گروه مهندسی کامپیوتر، دانشگاه قم، قم، ایران، (email: m.kherad@stu.qom.ac.ir).

یافتن و شناسایی گره‌های تأثیرگذار و دخالت‌دادن آنها بر تشخیص انجمن‌ها تأکید گردیده است؛ اگرچه در پاسخ به این چالش، مطالعات چندانی صورت نگرفته است.

- مسئله تشخیص انجمن در شبکه‌های بزرگ در زمره مسائل NP-hard قرار می‌گیرد. روش‌هایی برای شناسایی سریع انجمن‌ها پیشنهاد شده که اکثر آنها بر اساس بهینه‌سازی توابع هدف هستند. تاکنون روشی برای موازی‌سازی الگوریتم‌های پیشنهادی برای حل مشکل ارائه نگردیده است.

در این مقاله، روشی برای حل تمامی چالش‌های ذکرشده ارائه گردیده است. هدف از این مقاله، ارائه یک الگوریتم کشف انجمن پویاست که از تأثیر و نفوذ گره‌ها برای ایجاد انجمن استفاده می‌کند. الگوریتم پیشنهادی در مرحله اول، تأثیر گره‌ها در شبکه را محاسبه می‌کند و k گره بالاتر به‌عنوان گره مرکزی[1] انتخاب شده و سپس انجمن‌ها بر اساس این گره‌های مرکزی انتخاب می‌شوند. گره‌هایی که یال مستقیم به یک گره مرکزی داشته باشند، به آن انجمن اختصاص می‌یابند. سایر گره‌ها که فاقد یال مستقیم به یکی از گره‌های مرکزی هستند، بر اساس میزان شباهت و همسایگان مشترک به انجمن‌ها ملحق می‌شوند. در صورتی که برای یک گره هیچ یک از همسایگی‌ها به انجمن خاصی تعلق نداشته باشند، خودشان تشکیل انجمن می‌دهند؛ بدین ترتیب تعداد انجمن می‌تواند بیش از k باشد. در مرحله بعد، انجمن‌ها بررسی شده و بر اساس یک معیار جدید، میزان همپوشانی انجمن‌ها محاسبه و در نهایت انجمن‌ها با هم ادغام می‌شوند. در ادامه فرایند کشف انجمن با دریافت یک بازه زمانی دیگر، بعد از شناسایی تغییرات در شبکه، انجمن‌ها به‌روز می‌شوند. فرایند به‌روزشدن انجمن‌ها بر اساس ایجاد انجمن جدید، حذف انجمن موجود و یا ادغام انجمن‌های موجود انجام می‌شود. پیش از این در [۴] روشی برای تشخیص انجمن در شبکه‌های اجتماعی با در نظر گرفتن تأثیر گره‌های مهم ارائه نمودیم و در این مقاله توسعه آن برای حالت موازی با در نظر گرفتن انجمن‌های همپوشان ارائه می‌گردد.

به طور خلاصه نوآوری‌ها و دستاوردهای این مقاله شامل موارد زیر هستند:

- ارائه یک روش تشخیص انجمن در شبکه پویا بر اساس تأثیر گره‌ها بر اساس اطلاعات محلی و سراسری و توجه به امکان همپوشانی انجمن‌ها
- ارائه روشی جهت شناسایی تغییرات در شبکه پویا و میزان تأثیرگذاری هر گره و اعمال این تغییرات در انجمن‌های موجود
- مقایسه الگوریتم پیشنهادی با روش‌های پیشرفته اخیر در زمینه تشخیص انجمن در شبکه‌های پویا شامل Dynamic Louvain، DynaMo، IncNSA، ECD، DPC-DLP، DynCRep و CD-SACS بر روی سه مجموعه داده استاندارد Cit-Hep Ph، sx-mathoverflow و CollegeMsg با معیارهای ارزیابی مدولاریتی نیومن، مدولاریتی با جریمه تقسیم، مدولاریتی چگالی و زمان اجرا
- مجهزکردن روش پیشنهادی به اجرای موازی برای یافتن انجمن‌ها در زمان کوتاه که این روش را برای شبکه‌های بزرگ بسیار کارا می‌نماید.

در ادامه این مقاله در بخش ۲، مفاهیم پایه پژوهش بیان شده است. بخش ۳ به پیشینه تحقیق با مقایسه کارهای انجام‌شده در حوزه حل مسئله کشف انجمن با استفاده از گره‌های تأثیرگذار پرداخته است. اجزا و فازهای روش پیشنهادی در بخش ۴ و پیاده‌سازی و ارزیابی روش مذکور در بخش ۵ ارائه شده است. نهایتاً خلاصه و نتیجه‌گیری رویکرد پیشنهادی در بخش ۶ جمع‌بندی شده است.

۲- مفاهیم پایه

۲-۱ شبکه‌های اجتماعی

همان طور که در فصل پیش عنوان شد، در یک تعریف ساده، مجموعه‌ای از گره‌ها و ارتباط بین آنها را می‌توان شبکه نامید. شبکه، یک ساختار به‌هم‌پیوسته از گره‌هایی است که با هم ارتباطاتی را شکل می‌دهند. حال چنانچه این شبکه برای بیان تعاملات اجتماعی بین مجموعه‌ای از کاربران به کار رفته باشد، آن را شبکه اجتماعی می‌نامیم. اغلب برای نمایش شبکه‌های اجتماعی از ماتریس مجاورت استفاده می‌شود. در این ماتریس چنانچه n گره از کاربران را داشته باشیم، وجود رابطه بین هر زوج از کاربران را با یک و عدم وجود چنین ارتباطی را با صفر نمایش می‌دهیم. لذا ماتریس مجاورت یک شبکه، ماتریسی $n\times n$ متشکل از صفر و یک خواهد بود. بدیهی است ساختار چنین شبکه‌هایی کاملاً پویا بوده و با اضافه یا کم شدن کاربران (رئوس گراف) و ایجاد یا حذف ارتباط بین کاربران (یال‌های گراف) در هر لحظه تصویر گراف با لحظه قبل متفاوت خواهد بود. در نتیجه، انجمن‌های موجود در شبکه نیز به‌صورت پویا در حال تغییر می‌باشند [۱۲].

۲-۲ تشخیص انجمن

در علم شبکه برای تشخیص انجمن که خوشه‌بندی گراف نیز نامیده می‌شود، تعریف مشخص، دقیق و مورد پذیرش توسط عامه‌ای وجود ندارد [۱۳]. در [۱۴] انجمن به‌صورت گروهی از گره‌ها در یک گراف تعریف شده که یال‌هایی که آنها را به یکدیگر متصل می‌کند، بسیار بیشتر از یال‌هایی است که آن انجمن را به بقیه گراف متصل می‌نماید. نیومن نیز انجمن را مجموعه رئوس یک گراف در نظر می‌گیرد که تعداد یال‌های بین آنها بیشتر از میانگین می‌باشد [۱۵]. لذا منظور از تشخیص انجمن، یافتن انجمن‌هایی (زیرگراف‌هایی) است که با تعریف فوق برای انجمن هم‌خوانی داشته باشند. چنانچه شبکه مورد نظر با زمان تغییر نکند، مسئله تشخیص انجمن به‌صورت خوشه‌بندی گراف تعریف شده که یک مسئله کاملاً مطالعه‌شده است.

با معرفی و رواج شبکه‌های پویا، مفهوم انجمن‌های پویا نیز مطرح شد. انجمن‌های پویا انجمن‌هایی هستند که می‌توانند در طول زمان تغییر یا تکامل پیدا کنند. هدف از تشخیص انجمن پویا شناسایی مجموعه‌ای از تمام انجمن‌های موجود در شبکه پویاست که پارتیشن‌های توصیف‌شده توسط آن می‌توانند همپوشانی داشته باشند. مسئله تشخیص انجمن در شبکه‌های مقیاس بزرگ در زمره مسائل NP-hard قرار می‌گیرد.

راه‌حل‌های زیادی برای حل مسئله شناسایی انجمن‌ها در شبکه‌های اجتماعی پیشنهاد شده است. بزرگ‌بودن مقیاس شبکه‌ها از یک سو و همپوشانی بین انجمن‌ها از سوی دیگر، مسئله تشخیص انجمن‌ها را با دشواری روبه‌رو می‌نماید. به‌منظور سنجش و ارزیابی کیفیت انجمن‌ها معیارهای مختلفی ارائه شده است. یکی از مهم‌ترین معیارها، ماژولاریتی است که به‌طور گسترده برای ارزیابی کیفیت ساختار انجمن شبکه مورد استفاده قرار می‌گیرد. ساختارهای انجمن با ماژولاریتی بالا دارای اتصالات متراکم‌تری در بین رئوس در انجمن‌های مشابه هستند؛ اما

1. Seed

اتصالات پراکنده‌تری در بین رئوس سایر انجمن‌ها دارند. در فصل آینده مرور مختصری بر روش‌های موجود برای حل مسئله می‌پردازیم.

۳- پیشینه تحقیق

در این فصل مروری بر مهم‌ترین پژوهش‌های انجام‌شده برای حل مسئله تشخیص انجمن‌ها در شبکه‌های اجتماعی ارائه می‌گردد. هدف از این مرور صرفاً ذکر ایده‌هایی است که برای حل مسئله استفاده شده و از هر گونه تقسیم‌بندی پرهیز شده است.

در دسته‌ای از روش‌های ارائه‌شده، رویکرد اتخاذشده شامل یک استراتژی بهینه‌سازی چندهدفه است؛ مثلاً [۱۶] تکنیک ترکیب احتمالاتی را به کار می‌گیرد و از دو رویکرد متمایز، یعنی تنوع همسایه و جمعیت همسایه استفاده می‌کند. این رویکردها تشکیل سریع و دقیق اجتماعات مناسب را تسهیل می‌کنند. همچنین استفاده از یک معیار پیشرفت به نویسندگان اجازه می‌دهد تا شباهت‌های بین انجمن‌های تشکیل‌شده در دو تصویر لحظه‌ای متوالی را شناسایی کنند.

در [۱۷]، روشی برای تشخیص انجمن در شبکه‌های پویا بر اساس نفوذ ارائه شده است. این روش، مسئله را به عنوان یک مسئله بهینه‌سازی ترکیبی فرموله می‌کند که هدف آن، پارتیشن‌بندی یک شبکه اجتماعی معین به m انجمن مجزاست. هدف، بیشینه‌کردن مجموع انتشار نفوذ در شبکه اجتماعی با بیشینه‌کردن آن درون هر انجمن می‌باشد.

در [۱۸]، روشی برای شناسایی انجمن‌ها در شبکه‌های پویا ارائه شد که بر اساس ردیابی ستون فقرات‌ها و پل‌ها عمل می‌کند. آنها از «ستون‌های فقرات» برای نشان‌دادن یال‌های حیاتی انجمن‌ها و از یال‌های «پل» برای توصیف ارتباطات کلیدی بین انجمن‌ها استفاده کردند.

محققان در [۱۹] به بررسی مدل جدیدی برای شبکه‌های پویا به نام DynCRep می‌پردازند که در کنار ساختار انجمن‌ها از متغیرهای پنهان مبنی بر متقارن‌بودن روابط نیز برای تولید شبکه استفاده می‌کند. این مدل بر اساس یک فرایند مارکوف بنا شده که در آن، احتمال وجود یال بین دو گره در زمان t وابسته به زمان قبلی $t-1$ است. روش پیشنهادی برای تشخیص انجمن، مدل تولیدی احتمالی با متغیرهای پنهان است که به طور همزمان از تقارن و عضویت در انجمن به عنوان اطلاعات ساختاری شبکه استفاده می‌کند. این مدل فرض می‌کند که وجود یک لینک بین دو رأس به طور شرطی وابسته به عضویت آن دو رأس در انجمن‌ها و همچنین وجود لینک متقارن در قدم زمانی قبل است. این مدل بر روی داده‌های مصنوعی و واقعی آزمایش شد و نتایج نشان داد که درک بهتری از متقارن‌بودن روابط دارد و عملکرد بهتری در پیش‌بینی پیوندهای آینده شبکه دارد.

یک روش جدید تشخیص انجمن با استفاده از شبکه‌های عصبی خودرمزنگار عمیق و الگوریتم جستجوی کلاغ در [۲۰] با نام CD-SACS پیشنهاد شده است. در این روش، ابتدا یک ماتریس مدولاریته برای گراف ورودی تولید می‌شود؛ سپس از طریق یک سری خودرمزنگارهای پشته‌ای، بعد ماتریس مدولاریته کاهش داده می‌شود. این کاهش بعد باعث می‌شود تا توپولوژی شبکه حفظ شده و همچنین زمان محاسباتی بهبود یابد. سپس ماتریس کاهش‌یافته به عنوان ورودی به الگوریتم خوشه‌بندی $k-\text{means}$ تغییریافته با استفاده از الگوریتم بهینه‌سازی جستجوی کلاغ داده می‌شود. این الگوریتم به جای تولید تصادفی مراکز اولیه خوشه‌ها از جستجوی کلاغ استفاده می‌کند تا مراکز خوشه‌ها را به‌صورت بهینه تولید کند.

نویسندگان در [۲۱] ابتدا گره‌های تأثیرگذار سراسری و محلی در شبکه را شناسایی کرده‌اند و سپس سعی نموده‌اند تا از بین گره‌های تأثیرگذار موجود، گره‌هایی را که فاقد همپوشانی هستند، شناسایی نموده و اقدام به شکل‌دادن انجمن‌ها نمایند.

نویسندگان [۲۲] یک الگوریتم انتشار برچسب جدید را برای تشخیص انجمن در شبکه‌های اجتماعی در مقیاس بزرگ بر اساس اهمیت گره و تأثیر برچسب ارائه نموده‌اند. این پژوهش از یک روش جدید اهمیت گره با استفاده از شبکه بیزی برای ارزیابی اهمیت گره استفاده می‌کند. نتایج به‌دست‌آمده از این روش روی شبکه‌های دنیای واقعی و مصنوعی نشان داده که می‌تواند به‌طور قابل توجهی، کیفیت تشخیص انجمن را بهبود بخشد و همچنین در موارد پیچیدگی مشابه از دقت و پایداری بهتری برخوردار است.

محققان در [۲۳] مسئله تشخیص انجمن را از زاویه انتشار اطلاعات پویا بررسی نموده‌اند. این روش از یک مدل جدید به نام نفوذ گره[1] برای ارزیابی تأثیر بین گره‌ها و انجمن‌ها میانی پیشنهاد نموده است. همچنین یک الگوریتم جدید به نام تجمع تودرتو از تأثیرات گره برای ادغام گره‌ها یا انجمن‌های میانی در مسیر صعودی پیشنهاد شده است. این روش از چهار معیار برای تعیین تأثیر گره و شش معیار شباهت برای دستیابی به معیار تأثیرپذیری استفاده نموده است.

در ادامه به بررسی پژوهش‌های ارائه‌شده برای کشف انجمن در شبکه‌های پویا می‌پردازیم. در اغلب مطالعات، جهت کشف انجمن پویا از همان الگوریتم‌های ایستا استفاده می‌شود؛ در حالی که در هر بازه زمانی معین، یک تصویر لحظه‌ای از شبکه گرفته شده و الگوریتم ایستا روی آن اعمال می‌شود. در [۲۴] الگوریتمی به‌منظور تشخیص انجمن پویا مبتنی بر ماژولاریته با هدف شناسایی انجمن در شبکه‌های پویا توسط استفاده مکرر از الگوریتم‌های استاتیک، اما به روشی کارآمدتر ارائه شده است. این روش یک الگوریتم تطبیقی و افزایشی است که برای به حداکثر رساندن افزایشی ماژولاریته در حین به‌روزرسانی ساختار انجمن شبکه‌های پویا طراحی شده است. در این مقاله برای به‌روزرسانی ساختارهای انجمن، شبکه پویا به‌عنوان دنباله‌ای از تغییرات تدریجی شبکه مدل شده و برای هر تغییر تدریجی شبکه، عملیاتی برای به حداکثر رساندن ماژولاریتی طراحی شده است.

یک روش تشخیص انجمن همپوشانی احتمالی در [۱۲] ارائه شده است. این روش، وظیفه تشخیص انجمن را به‌عنوان یک مسئله فاکتورسازی ماتریس غیرمنفی در نظر می‌گیرد. روش پیشنهادی از یک مدل احتمالی برای کنترل پویایی ساختار انجمن و از روش مختصات بلوکی برای حل تابع هدف مدل فاکتورسازی ماتریس استفاده می‌نماید. این راهکار، عامل نهفته غیرمنفی را برای سرعت‌بخشیدن به محاسبه گرادیان‌ها تخمین می‌زند. نتایج به‌دست‌آمده نشان می‌دهند که روش پیشنهادی از نظر معیارهای ارزیابی شناخته‌شده از الگوریتم‌های قبلی در شبکه‌های در حال تکامل بهتر عمل می‌کند.

محققین در [۲۵] روشی را به نام DPC-DLP ارائه نمودند که از خوشه‌بندی قله تراکم برای یافتن مراکز انجمن و از انتشار برچسب برای تخصیص‌دادن برچسب‌ها به گره‌ها و تشکیل انجمن‌های نهایی استفاده نموده و قادر است به‌طور مؤثرتری برچسب‌های واقعی را به نمونه‌های داده‌ای که در مناطق مرزی و همپوشانی قرار دارند، اختصاص دهد. نتایج حاصل از آزمایش این روش، نشان از مناسب‌بودن عملکرد آن نسبت به سایر روش‌ها دارد.

1. Node Influence

یک روش جدید برای تشخیص انجمن با استفاده از الگوریتم ژنتیک در [۲۶] پیشنهاد شده و در آن در کنار عملگرهای ژنتیکی کلاسیک و سنتی، یک عملگر ژنتیکی مهاجرتی جدید ارائه شده است. این عملگر بر یک چارچوب تجزیه تکیه دارد و سعی می‌کند تا تشخیص انجمن تکاملی را به‌عنوان یک مسئله بهینه‌سازی چندهدفه فرموله کند. همچنین روشی سریع برای محاسبه معیار ماژولاریتی پیشنهاد شده که منجر به جستجوی سریع‌تر فضای مسئله به‌منظور یافتن راه‌حل بهینه می‌شود.

پژوهشگران در [۲۷]، یک روش تشخیص انجمن تکاملی را پیشنهاد نموده‌اند که می‌تواند ساختارهای اجتماعی با کیفیت بالا را از شبکه‌های در حال تکامل در بازه‌های زمانی مختلف تشخیص دهد. هنگامی که شبکه تکامل می‌یابد، روش پیشنهادی فقط وابستگی‌های اجتماعی گره‌های جزئی را به‌طور مؤثر در نظر می‌گیرد که یا گره‌های تازه متولدشده یا برخی گره‌های فعال در زمان قبل هستند. سپس زیرگراف‌هایی برای این گره‌ها ساخته می‌شوند تا انجمن‌های اولیه به دست آیند. در نهایت نتیجه نهایی از تشخیص انجمن از طریق بهینه‌سازی انجمن‌های اولیه به دست می‌آید.

نویسندگان در [۲۸] یک روش بهینه‌سازی مدولاریتی محلی را فراهم می‌کنند که طی آن، تنها انجمن‌هایی که دچار تغییر شدند بررسی شده و سایر انجمن‌ها دست‌نخورده باقی می‌مانند. طبق ادعای مقاله، الگوریتم دارای اثربخشی مناسب‌تری نسبت به سایر الگوریتم‌های مشابه است.

با مرور مطالعات انجام‌شده می‌توان مسائل باز پژوهشی بسیاری را تعریف نمود. شناسایی انجمن یک مسئله NP-hard است. ذات پویابودن شبکه و وجود تصاویر مختلف از شبکه، نیازمند انجام محاسبات مجدد برای به‌روزکردن و یا ایجاد انجمن‌هاست؛ بنابراین به روش‌هایی نیاز است که تا حد امکان کم‌هزینه و سریع و به لحاظ محاسباتی ساده باشد. همچنین برای کشف انجمن در شبکه‌های پویا لازم است دو تصویر لحظه‌ای قبلی و فعلی با هم مورد مقایسه قرار گیرند. این مقایسه، اغلب زمان‌بر بوده و نیاز به محاسبات متوسط تا بالایی دارد. داشتن قوانین و یا روش‌هایی جهت تسریع این فرایند می‌تواند کلید موفقیت در کاهش سرعت الگوریتم‌های کشف انجمن در شبکه‌های پویا باشد. یکی از راهکارهای مناسب برای حل این مسئله موازی‌سازی است. موازی‌کردن محاسبات سبب افزایش سرعت الگوریتم می‌شود. به‌علاوه، اهمیت‌دادن به گره‌های رهبر و یا به عبارتی تأثیرگذار می‌تواند بستر تولید بسیاری از انجمن‌ها باشد. این موضوع در بسیاری از تحقیقات نادیده گرفته شده است. نهایتاً اغلب تحقیقات، موضوع همپوشانی انجمن‌ها را نادیده گرفته‌اند؛ بنابراین روش کارا برای حل مسئله بایستی به تمامی جوانب مسئله توجه نماید. در فصل آینده روشی کارا برای حل مسئله ارائه خواهد شد که به تمامی چالش‌های ذکرشده پاسخ خواهد داد.

٤- روش پیشنهادی

روش پیشنهادی دارای شش فاز اصلی می‌باشد که در ادامه هر فاز را شرح می‌دهیم.

٤-١ دریافت تصویر لحظه‌ای از شبکه

در این مرحله، از شبکه چندین تصویر لحظه‌ای در زمان‌های مختلف گرفته می‌شود. هر تصویر نشان‌دهنده مجموعه‌ای از گره‌ها و لینک بین آنهاست که بر اساس بازه زمانی موجود تقسیم شده‌اند. هر تصویر به‌صورت مستقل به الگوریتم داده شده تا انجمن‌های آن شناسایی شود. تصویر اول از شبکه برای تشخیص انجمن و تصاویر بعدی برای به‌روزرسانی انجمن‌ها استفاده خواهد شد. علاوه بر این، گره‌ها و یال‌های موجود در تصویر جاری از شبکه، استخراج و کدگذاری می‌شوند. برای این کار، دو مجموعه شامل مجموعه گره‌ها و مجموعه یال‌ها خواهیم داشت. فرض بر این است که هر گره، حاوی یک شناسه منحصربه‌فرد است و شبکه با استفاده از این شناسه می‌تواند گره را شناسایی کند؛ بنابراین با توجه به همین شناسه، اقدام به تولید مجموعه گره‌ها و یال‌ها می‌شود.

مجموعه گره‌های موجود در تصویر جاری از شبکه در آرایه $Current_Nodes$ ذخیره می‌شوند. برای ذخیره‌سازی یال‌ها نیز یک ماتریس $m\times 2$ به نام $Current_Edges$ خواهیم داشت که m برابر با تعداد یال‌ها و هر سطر شامل شناسه گره‌های تشکیل‌دهنده یال است.

٤-٢ انتخاب گره‌های تأثیرگذار بر اساس اطلاعات محلی و سراسری

در این مرحله در سه گام، بر اساس اطلاعات محلی و سراسری هر گره، اقدام به یافتن گره‌های تأثیرگذار می‌شود. در گام نخست به‌منظور شناسایی اطلاعات سراسری گره، ابتدا از الگوریتم تجزیه شبکه $k-shell$ استفاده می‌شود. به‌منظور محاسبه اهمیت گره‌ها در شبکه، معیارهای مختلفی وجود دارد، اما فقط درجه و ضریب خوشه‌بندی گره‌ها می‌تواند اطلاعات محلی شبکه‌ها را مشخص کند. $k-shell$ یک زیرگراف متصل حداکثر از گراف G است که در آن درجه هر رأس حداقل برابر با k است. مقدار $k-shell$ برای گره i که با $Ks(i)$ نشان داده می‌شود، نشان می‌دهد که گره i به پوسته k تعلق دارد اما به هیچ $k+1$ پوسته دیگر تعلق ندارد. روش تجزیه $k-shell$، اغلب برای شناسایی گره هسته و گره‌های حاشیه‌ای شبکه استفاده می‌شود. این روش با حذف تمام گره‌هایی که تنها یک پیوند دارند، شروع می‌شود تا زمانی که هیچ گرهی باقی نماند و آن‌ها را به پوسته ۱ اختصاص می‌دهد. به همین ترتیب به‌صورت بازگشتی تمام گره‌های با درجه ۲ (کمتر) را حذف می‌کند و پوسته ۲ را ایجاد می‌کند. این روند تا زمانی که تمام گره‌های شبکه به یک پوسته اختصاص داده شوند، ادامه پیدا می‌کند. پوسته‌هایی با شاخص‌های بالاتر در هسته یا مرکز شبکه قرار دارند. روش تجزیه $k-shell$ را می‌توان به‌طور مؤثر با پیچیدگی زمانی خطی $O(m)$ که در آن m تعداد یال‌ها در شبکه است، پیاده‌سازی کرد.

در گام دوم، اطلاعات سراسری و اطلاعات محلی هر گره به تفکیک محاسبه می‌شود. اطلاعات سراسری، وضعیت گره را در کل شبکه نشان می‌دهد. گرهی شانس مرکزیت یک خوشه را دارد که میزان $k-shell$ را افزایش دهد. اطلاعات سراسری گرهی مانند i که با GI_i نشان داده می‌شود، شدت وابستگی سایر گره‌های موجود در شبکه به گره i را نشان می‌دهد. به عبارت بهتر، GI_i بر اساس میانگین $shell$های موجود در همسایگی گره‌ها به دست می‌آید؛ پس برای یک گره مانند گره i، طبق (۱) محاسبه می‌شود. در این رابطه $NumShell$ تعداد لایه‌های ایجادشده توسط تجزیه $k-shell$ و $neighbor(i,j)$ تعداد همسایگی‌های گره i است که دارای لایه j از تجزیه $k-shell$ باشند

$$GI_i = \frac{\sum_{j\in NumShell} \left|neighbor(i,j)\right|\times j}{NumShell} \qquad (۱)$$

پس از به دست آوردن اطلاعات سراسری گره‌ها، اقدام به اندازه‌گیری اطلاعات محلی می‌شود. برای اندازه‌گیری اطلاعات محلی نیز از تعداد همسایگی‌های هر گره استفاده می‌شود؛ بنابراین بر اساس (۱) مقدار LI_i که به معنای اطلاعات محلی گره iام می‌باشد، به دست می‌آید

```
1.  initialize V = all nodes in G
2.  for i = 1: n
3.  | compute k-shell by k-shell decomposition algorithm
4.  end for
5.  compute k-shell for each node
6.  calculate the number of neighbors of each node
7.  for each node like i:
8.  | calculate GI(i) according to formula (2)
9.  | calculate LI (i) according to formula (3)
10. | calculate influence(i) using GI(i) and LI(i) and (4)
11. end for
```

شکل ۱: محاسبه تأثیر گره‌ها.

$$LI_i = \left|neighbor(i)\right| \qquad (۲)$$

در گام نهایی به‌منظور محاسبه نفوذ گره در شبکه، اطلاعات سراسری و محلی گره مطابق با (۳) با هم ادغام می‌شود. در این رابطه a و b ضریب اطلاعات سراسری و اطلاعات محلی است

$$influence(i) = \alpha(GI_i) + \beta(LI_i) \qquad (۳)$$

گره‌هایی که میزان تأثیرگذاری بیشتری دارند، به‌عنوان مرکز انجمن در نظر گرفته می‌شوند. شبه‌کد مربوط به الگوریتم این فاز در شکل ۱ آمده است.

۳-٤ راه‌اندازی انجمن‌های اولیه

با توجه به فاز دوم، فهرست تأثیر گره‌ها به‌دست‌آمده و گره‌های موجود در لیست رتبه‌بندی به ترتیب نزولی مرتب می‌شوند. در گام بعد، k گره دارای نفوذ بالاتر به‌عنوان مراکز انجمن‌ها در نظر گرفته می‌شوند. از بین گره‌های باقی‌مانده در شبکه، گره‌هایی که یک لینک مستقیم به گره‌های مرکز خوشه دارند، انجمن را تشکیل می‌دهند. در صورتی که دو مرکز انجمن دارای یال مستقیم به هم باشند، از این یال صرف نظر می‌شود و این دو در انجمن‌های یکدیگری قرار نمی‌گیرند.

٤-٤ توسعه انجمن‌ها

در این فاز گره‌هایی که هنوز انجمن ندارند، به انجمن‌های موجود ملحق می‌شوند و برای این کار به همسایگی‌ها توجه می‌شود. در حقیقت گره به خوشه‌ای ملحق می‌شود که اکثر همسایگانش به آن تعلق داشته باشند و اگر تعداد همسایگی‌ها برای بیش از یک خوشه، یکسان باشد، گره به همه خوشه‌ها ملحق می‌شود. مقدار وابستگی به انجمن با استفاده از (۴) محاسبه می‌شود

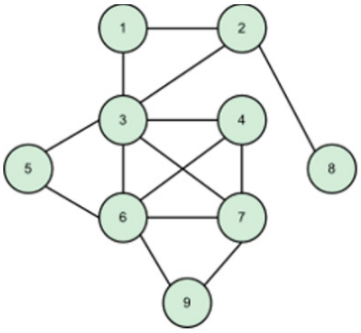
$$dependency(i) = \max_{j \in ClusterHeads} (\left|cluster(neighbor(i), j)\right|) \qquad (۴)$$

که $\left|cluster(neighbor(i), j)\right|$ تعداد همسایگان گره i برای خوشه j ام را نشان می‌دهد. در صورتی که گره حاوی همسایگانی باشد که هنوز خوشه به آنها اختصاص داده نشده باشد، آن همسایگان در رابطه فوق نادیده گرفته می‌شوند. شکل ۲ مراحل مربوط به ایجاد و توسعه انجمن‌ها را بیان می‌نماید. در خط ۱ از شکل ۲، گره‌ها بر اساس میزان نفوذشان مرتب می‌شوند. در خط ۲، k گره که دارای نفوذ بیشتر است به‌عنوان سرخوشه انتخاب و در یک لیست به نام *ClusterHead* که وظیفه نگهداری مراکز انجمن را دارد، ذخیره می‌شوند. در خط ۳ به‌منظور مدیریت بهتر گره‌ها برای توسعه انجمن، یک لیست به نام *Candidate* ایجاد می‌شود. این لیست وظیفه نگهداری گره‌هایی را بر عهده دارد که حداقل یکی از همسایگی‌های آن دارای انجمن باشد. بدین ترتیب مطمئن می‌شویم که گره‌هایی برای تعمیم انجمن انتخاب می‌شود که حتماً دارای

```
1.  Sort nodes in a descending order
2.  HeadCluster← select k top nodes
3.  Candidate←Neighbors(HeadCluster)
4.  assign_Nodes←assign_Nodes union HeadCluster
5.  for j in Candidate
6.  | flag←false
7.  | for HeadCluster like h
8.  | | if j is neighbors h
9.  | | | Communities (h) ←j
10. | | | flag ← true
11. | | end if
12. | end for
13. | if (flag == false)
14. | | maxCom ← calculate dependency(j) using (5)
15. | | for each community in maxCom
16. | | | Communities (community) ←j
17. | | end for
18. | end if
19. | for each node in Neighbors(j) like t
20. | | if t dos not exist in assign_Nodes
21. | | | Candidate←Candidate union t
22. | | end if
23. | end for
24. | assign_Nodes←assign_Nodes union j
25. end for
26. while (there is node in Graph that does not exist in
        assign_Nodes like j)
27. | neighbor← Neighbors(j) union j
28. | HeadCluster←maxinfluence(neighbor)
29. | for each node in neighbor like t
30. | | Communities (maxinfluence(neighbor)) ←t
31. | | assign_Nodes←assign_Nodes union t
32. | end for
33. end while
```

شکل ۲: الگوریتم ایجاد و توسعه انجمن.

یک همسایگی دارای خوشه باشند و در غیر این صورت گره به هیچ انجمنی تعلق نخواهد داشت. لذا در خط ۳، همه همسایگان مراکز خوشه به *Candidate* ملحق می‌شوند. سپس یک لیست دیگر نیز با نام *assign_Nodes* تشکیل شده که وظیفه آن نگهداری گره‌هایی است که به عضویت حداقل یک انجمن درآمده باشند (خط ۴). در ادامه، تمام نقاطی که به‌عنوان سرخوشه انتخاب شدند، به این اضافه می‌شوند تا دوباره در فرایند خوشه‌بندی ظاهر نشوند. در خط ۵ به ترتیب قرارگیری گره‌ها، اولین گره مانند j انتخاب می‌شود. در خط ۶ یک متغیر به نام *flag* مقداردهی می‌شود تا در صورتی که گره، همسایه یکی از مراکز انجمن باشد، شناسایی شود. در خط ۷ برای هر مرکز انجمن، چک می‌شود که آیا گره j همسایگی آن است یا خیر. در صورتی که گره j همسایه آن مرکز انجمن باشد، گره j به آن انجمن ملحق می‌شود و مقدار *flag* نیز *true* می‌شود (خط ۷ تا ۱۲). در صورتی که گره j همسایه هیچ یک از مراکز انجمن نباشد، با استفاده از (۴)، مقدار وابستگی به انجمن محاسبه شده و نهایتاً به انجمن‌های ممکن ملحق می‌شود (خط ۱۳ تا ۱۸). سپس تمام همسایگی‌های گره j در صورتی که فاقد انجمن باشند، به لیست *Candidate* اضافه می‌شوند و گره j نیز به *assign_Nodes* اضافه می‌شود تا دیگر در فرایند ایجاد انجمن ظاهر نگردد (خط ۱۹ تا ۲۴). نهایتاً این روند تا زمانی که لیست *Candidate* دارای یک عضو باشد ادامه می‌یابد. در خط ۲۶ تا ۳۰ نیز وجود پراکندگی



شکل ۳: گراف مثال.

در انجمن بررسی شده و در صورت وجود گره‌های فاقد انجمن، بانفوذترین گره در بین آن گره و همسایگانش به‌عنوان گره مرکز خوشه انتخاب گردیده و سپس سایر گره‌ها به‌عنوان عضو انجمن به آن ملحق می‌شوند.

۴-۵ ارزیابی و ادغام انجمن‌ها

در فرایند انتخاب گره‌های تأثیرگذار، k گره با تأثیرگذاری بالا و همسایه‌های آنها از انجمن‌های اولیه تشکیل شده‌اند؛ بنابراین اگر یک گره به چندین مرکز انجمن متصل باشد، می‌توان آن را به انجمن‌های مختلف اختصاص داد. در فرایند توسعه انجمن، اگر یک گره با اولویت یکسان به چندین انجمن تعلق داشته باشد، گره به این انجمن‌ها اختصاص داده می‌شود؛ بنابراین بسیاری از گره‌ها دارای همپوشانی هستند. با این حال، اگر همپوشانی بین انجمن‌ها خیلی زیاد باشد، پیچیدگی الگوریتم را افزایش می‌دهد. در نتیجه، کنترل همپوشانی انجمن‌ها برای بهینه‌سازی ساختار انجمن ضروری است. به‌منظور کنترل همپوشانی از نرخ همپوشانی استفاده می‌شود که بر اساس (۵) محاسبه می‌گردد

$$\delta = \frac{\left|C_i \cap C_j\right|}{\min\{C_i, C_j\}} \qquad (۵)$$

هرچه δ بزرگ‌تر باشد، گره‌های بیشتری در بین انجمن‌های دارای همپوشانی دارند و در نتیجه دو انجمن مستعد ادغام‌شدن هستند. با این حال در دنیای واقعی این امکان وجود دارد که دو انجمن، دارای همپوشانی بالا باشند؛ اما به دلایلی همچون موضوع انجمن، امکان ادغام آن دو موجود نداشته باشد؛ بنابراین در این مقاله علاوه بر معیار فوق از معیار دیگری به نام برازندگی انجمن نیز استفاده می‌شود. معیار برازندگی انجمن به اهمیت انجمن در شبکه اشاره دارد و طبق (۶) محاسبه می‌شود

$$fitness(C_i) = \frac{density_i + unique_i}{۲} \qquad (۶)$$

$$density_i = \frac{\left|C_i\right|}{totalNodes} \qquad (۷)$$

$$unique_i = \frac{unique_member_i}{member} \qquad (۸)$$

که $density_i$ تراکم انجمن را مشخص می‌کند و بر اساس نسبت اعضای انجمن به کل گره‌های موجود در شبکه محاسبه می‌شود. $totalNodes$ نشان‌دهنده تمام گره‌های موجود در شبکه است و همچنین $unique_i$ نسبت گره‌هایی است که تنها در انجمن i وجود دارند (در انجمن‌های دیگر عضو نیستند) به کل گره‌های موجود در انجمن i. بدین ترتیب انجمن‌هایی که بتوانند مقدار برازندگی بیش از حد آستانه را کسب کنند، ماندگار می‌شوند؛ در غیر این صورت با انجمن دیگر ادغام می‌شود. به عبارت بهتر در مورد معیار برازندگی، هرچه این مقدار کمتر باشد، احتمال

جدول ۱: نمایش میزان $k-shell$ برای هر لایه.

گره	Shell	لیست همسایگان	اطلاعات سراسری	اطلاعات محلی
۱	۲	۲ و ۳	۱٫۶۷	۲
۲	۲	۱، ۳ و ۸	۲	۳
۳	۳	۱، ۲، ۴، ۵، ۶ و ۷	۵	۶
۴	۳	۳، ۶ و ۷	۳	۳
۵	۲	۳ و ۶	۲	۲
۶	۳	۳، ۴، ۵، ۷ و ۹	۵٫۳۳۳۳	۵
۷	۳	۳، ۴، ۶ و ۹	۷	۴
۸	۱	۲	۰٫۶۷	۱
۹	۲	۶ و ۷	۲	۲

ادغام‌شدن خوشه بیشتر است. به‌منظور ادغام دو خوشه، گرهی که میزان نفوذ بیشتری داشته به‌عنوان سرخوشه انتخاب شده و تمام اعضای هر دو خوشه به همراه مرکز خوشه دیگر به‌عنوان اعضای آن شناخته می‌شوند.

۴-۶ به‌روزرسانی انجمن‌ها

این فاز خود شامل چهار گام است که عبارتند از دریافت تصویر جدید و شناسایی تغییرات، تغییرات در انجمن‌ها، بررسی انجمن‌ها، تعیین انجمن برای گره‌های فاقد انجمن و ارزیابی و ادغام انجمن‌ها. در ادامه هر یک از گام‌ها تشریح می‌شوند.

در گام اول، یک تصویر لحظه‌ای دیگر از شبکه دریافت می‌شود و بر اساس انجمن‌های موجود اقدام به به‌روزکردن انجمن‌ها می‌شود. برای به‌روزرسانی انجمن‌ها با در نظر گرفتن اینکه t زمان جاری شبکه باشد، باید تغییرات صورت‌گرفته در شبکه در زمان t نسبت به زمان $t-۱$ (شبکه قبلی) را مشخص کرد. تغییرات شبکه می‌تواند شامل اضافه‌شدن یا حذف یک گره به/ از شبکه و یا اضافه/ حذف شدن یال به/ از شبکه باشد. فرض کنید گره‌های تعیین‌شده در شبکه قبل در لیست $Current_Node$ و لبه‌های مربوط به آن در لیست $current_Edge$ قرار گرفته باشد.

در گام دوم، بعد از شناسایی تغییرات، اقدام به اعمال آن به روی شبکه و انجمن‌های موجود از تصویر قبلی شبکه می‌شود. شبکه پنج نوع تغییر را می‌تواند فراهم کند که این تغییرات شامل حذف و اضافه شدن گره یا حذف و اضافه شدن یک یال است. در هر یک از این سناریوها میزان نفوذ گره‌ها و عضویت گره‌ها در انجمن‌ها ممکن است دستخوش تغییراتی شوند که بایستی در خصوص آن تصمیم‌گیری شود.

بعد از انجام تغییرات در گام قبل، اقدام به بررسی مجدد انجمن‌ها می‌شود. در این مرحله، تمامی انجمن‌هایی که دستخوش حداقل یک تغییر شده باشند، نفوذ مرکز انجمن و گره‌های موجود در آن بررسی شده و در صورت نیاز انجمن به گره دیگر واگذار می‌شود. همچنین در صورتی که انجمن نتواند حد آستانه نفوذ را برآورده کند، آن انجمن منحل شده و درباره اعضای آن، میزان $dependency(i)$ محاسبه شده و در خصوص گره‌های فاقد انجمن تصمیم‌گیری می‌شود.

در گام پایانی نیز عملیات ارزیابی میزان همپوشانی و برازندگی گره‌ها تعیین و مشخص می‌شود و در صورت نیاز انجمن‌ها با هم ادغام می‌گردند. در فصل آینده در خصوص کارایی روش پیشنهادی به‌صورت مفصل بحث خواهیم نمود.

۴-۷ مثال از روش پیشنهادی

برای درک روش پیشنهادی مثال ساده‌ای ارائه می‌نماییم. گراف موجود در شکل ۳ را در نظر بگیرید. در جدول ۱ با توجه به روابط ذکرشده در

شناسه گره	۱	
	شناسه	Shell
همسایگان	۲	۲
	۳	۳

$$\frac{|neighbor(1,1)|\times 1+|neighbor(1,2)|\times 2+|neighbor(1,3)|\times 3}{3}=$$

$$\frac{0\times 1+1\times 2+1\times 3}{3}=\frac{5}{3}=1/67$$

(الف)

شناسه گره	۱	
	شناسه	Shell
همسایگان	۲	۲
	۳	۳

$$\frac{|neighbor(2,1)|\times 1+|neighbor(2,2)|\times 2+|neighbor(2,3)|\times 3}{3}=$$

$$\frac{1\times 1+1\times 2+1\times 3}{3}=\frac{6}{3}=2$$

(ب)

شکل ۴: محاسبات انجام‌شده برای تعیین اطلاعات سراسری برای گره‌های (الف) ۱ و (ب) ۲ گراف مثال شکل ۳. (در متن ارجاع ندارد)

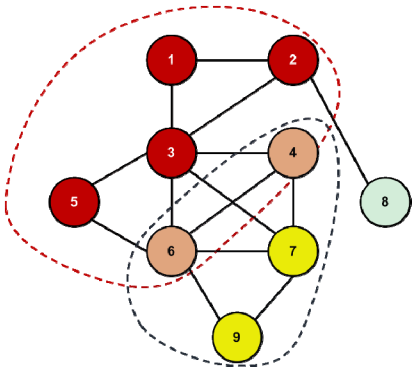
جدول ۲: نمایش میزان نفوذ برای گره‌های گراف مثال شکل ۴.

نام گره	میزان نفوذ
۳	۵/۵
۷	۵/۵
۶	۵/۱۶۶۵
۴	۳
۲	۲/۵
۵	۲
۹	۲
۱	۱/۸۳۵
۸	۰/۸۳۵

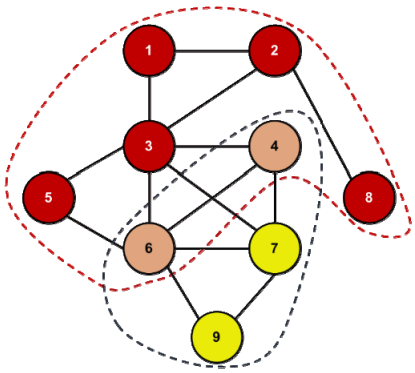
بخش ۴-۲، اطلاعات سراسری و محلی هر گره و در نتیجه میزان نفوذ گره مشخص می‌شود.

در شکل ۵ محاسبات مربوط به GI_i برای گره‌های ۱ و ۲ آمده و برای سایر گره‌ها نیز به همین صورت محاسبه انجام گرفته است. همان طور که بیان شد، اطلاعات محلی LI_i نیز برابر با تعداد همسایگان گره i است. در ادامه، میزان نفوذ هر یک از گره‌ها با توجه به اطلاعات موجود در جدول ۲ و (۴) با مقدار $a=\beta=0/5$ محاسبه می‌شود. بعد از مرتب‌سازی گره‌ها بر اساس میزان نفوذ با فرض اینکه مقدار $k=2$ باشد، گره‌های ۷ و ۳ به عنوان مرکز خوشه انتخاب می‌شوند. جدول ۲ ترتیب گره‌ها بر اساس میزان نفوذ را نشان می‌دهد.

شکل ۵، انجمن‌های متشکل از این مراکز خوشه را نشان می‌دهد. در این شکل گره‌های ۱، ۲ و ۵ به خوشه حاصل از گره ۳ و گره ۹ به مرکز خوشه ۷ تعلق دارد. دو گره ۴ و ۶ هم به هر دو خوشه تعلق دارند. تنها گرهی که هنوز بدون خوشه است، گره ۸ است. تنها همسایه گره ۸، گره ۲ است که قبلاً به خوشه حاصل از گره ۳ ملحق شده است؛ لذا گره ۸ نیز به این خوشه ملحق می‌شود. شکل ۶ انجمن‌های نهایی برای گراف مثال را نشان می‌دهد.



شکل ۵: انجمن‌های اولیه برای گراف مثال شکل ۳.



شکل ۶: انجمن‌های نهایی برای گراف مثال شکل ۳.

با توجه به گراف مثال و انجمن‌های تولیدشده، میزان نرخ همپوشانی δ برای دو انجمن با توجه به (۵) برابر خواهد بود با

$$\delta=\frac{|\{1,2,3,4,5,6,8\}\cap\{4,6,7,9\}|}{\min\{7,4\}}=\frac{|\{4,6\}|}{\min\{7,4\}}=\frac{2}{4}=0/5 \quad (9)$$

بر اساس (۶) تا (۸) مقدار معیار برازندگی برای دو انجمن به‌صورت زیر محاسبه شده است

$$density_i=\frac{|C_i|}{totalNodes}=\frac{|\{1,2,3,4,5,6,8\}|}{9}=\frac{7}{9} \quad (10)$$

برای خوشه حاصل از گره ۳

$$unique_i=\frac{unique_member_i}{member_i}=\frac{5}{7}=0/71 \quad (11)$$

$$fitness(C_i)=\frac{density_i+unique_i}{2}=\frac{0/78+0/71}{2}\approx 0/75 \quad (12)$$

برای خوشه حاصل از گره ۷

$$density_i=\frac{|C_i|}{totalNodes}=\frac{|\{1,2,3,4,5,6,8\}|}{9}=\frac{7}{9} \quad (13)$$

$$unique_i=\frac{unique_member_i}{member_i}=\frac{5}{7}=0/71 \quad (14)$$

$$fitness(C_i)=\frac{density_i+unique_i}{2}=\frac{0/78+0/71}{2}\approx 0/75 \quad (15)$$

با توجه به این نتایج، میزان همپوشانی برای دو خوشه ۰/۵ به دست آمده که با در نظر گرفتن مقدار آستانه ۰/۷۵ برآورده نمی‌شود. اگر مقدار به‌دست‌آمده، بیشتر از آستانه بود، دو خوشه می‌توانستند با هم ادغام گردند. علاوه بر این، مقدار برازندگی برای خوشه حاصل از گره ۳، ۰/۷۵ و برای

جدول ۳: مجموعه داده مورد استفاده در پیاده‌سازی.

نام مجموعه داده	تعداد رئوس	تعداد یال‌ها	نوع روابط	تعداد تصاویر لحظه‌ای
Cit-Hep Ph [۲۹]	۳۴۵۴۶	۴۲۱۵۷۸	ارجاع به هم	۱۱
sx-mathoverflow [۳۰]	۲۴۸۱۸	۵۰۶۵۵۰	نظردادن روی سؤال و روی پاسخ	۸
CollegeMsg [۳۱]	۱۸۹۹	۵۹۸۳۵	ارسال پیام خصوصی به هم	۷

خوشه حاصل از گره ۷، مقدار ۰/۴۵ بوده است. اگر مقدار آستانه را ۰/۵ در نظر بگیریم، خوشه حاصل از گره ۷ مقدار برازندگی کمتر از ۰/۵ را به دست آورده است؛ پس امکان ادغام دو خوشه با هم وجود دارد. اما با توجه به اینکه میزان همپوشانی نتوانسته است آستانه ۰/۷۵ را برآورده کند، فرایند ادغام منتفی می‌شود.

۴-۸ قابلیت موازی‌سازی

بخش عمده پیچیدگی زمانی روش پیشنهادی در فاز دوم برای محاسبه گره‌های تأثیرگذار بر اساس اطلاعات سراسری و محلی است. در این فاز، الگوریتم $k-shell$ که خود دارای پیچیدگی زمانی و محاسباتی زیادی است، مورد استفاده قرار می‌گیرد. در نتیجه با موازی‌سازی این الگوریتم می‌توان زمان روش پیشنهادی را به‌طور قابل توجهی کاهش داد. الگوریتم $k-shell$ را می‌توان به‌صورت موازی اجرا کرد. چندین روش برای موازی‌سازی این الگوریتم وجود دارد:

- موازی‌سازی مرحله حذف گره: در این روش، گره‌ها به طور همزمان در چندین پردازنده حذف می‌شوند. این روش می‌تواند به طور قابل توجهی زمان اجرای الگوریتم را برای شبکه‌های بزرگ کاهش دهد.
- موازی‌سازی محاسبه درجه گره: در این روش، درجه گره‌ها به طور همزمان در چندین پردازنده محاسبه می‌شود.
- موازی‌سازی مراحل تکرار: در این روش، مراحل تکرار الگوریتم $k-shell$ به طور همزمان در چندین پردازنده اجرا می‌شوند.

۵- پیاده‌سازی و ارزیابی

در این بخش به پیاده‌سازی و ارزیابی الگوریتم پیشنهادی پرداخته شده و برای پیاده‌سازی از زبان برنامه‌نویسی پایتون استفاده گردیده است. برای ارزیابی عملکرد الگوریتم پیشنهادی، آن را با پنج روش اخیراً ارائه‌شده در [۲۸]، [۲۷]، [۲۵]، [۱۹] و [۲۰] مقایسه نموده‌ایم. در ادامه، مجموعه داده‌های استفاده‌شده بررسی می‌گردد و سپس معیارهای ارزیابی معرفی شده و نتایج حاصل از این معیارها برای روش پیشنهادی و الگوریتم‌های مورد مقایسه بر روی مجموعه داده‌های معرفی‌شده ارائه می‌شود.

۱-۵ معرفی مجموعه داده‌ها

جهت اجرا، ابتدا مجموعه داده‌های مرتبط فراخوانی می‌شوند. مجموعه داده‌های مورد استفاده در این مقاله در جدول ۳ آمده است.

مجموعه داده Cit-Hep Ph دارای ۳۰۵۰۱ رأس و ۳۴۶۷۴۲ یال است. در این مقاله بر اساس سال، تصاویر لحظه‌ای استخراج شدند و از این رو ۱۱ تصویر لحظه‌ای استخراج شده است. هر تصویر لحظه‌ای شامل گره‌های همان سال به همراه گره‌های مربوط به سال قبل از آن است. مثلاً برای سال ۱۹۹۹، گره‌ها و یال‌های مربوط به سال ۱۹۹۹ و ۱۹۹۸ استخراج و استفاده می‌شوند. بدین ترتیب هم حذف و هم اضافه شده یال‌ها مورد بررسی قرار می‌گیرد.

مجموعه داده sx-mathoverflow شامل شبکه‌ای موقتی از تعاملات در وب‌سایت تبادل اطلاعات MathOverflow است. سه نوع مختلف از تعاملات شامل پاسخ‌دادن، نظردادن روی سؤال و نظردادن روی پاسخ وجود دارد. این مجموعه داده ۲۴۸۱۸ گره و ۵۰۶۵۵۰ یال دارد که طی ۲۳۵۰ روز جمع‌آوری شده است. در این پژوهش، این مجموعه داده بر اساس سال از هم جدا شده و نهایتاً شامل ۸ بازه زمانی مختلف است که در هر بازه زمانی، اطلاعات مربوط به سال جاری و سال قبل آمده است.

مجموعه CollegeMsg شامل پیام‌های خصوصی ارسال‌شده در یک شبکه اجتماعی آنلاین در دانشگاه کالیفرنیا است و ۱۸۹۹ گره و ۵۹۸۳۵ یال دارد. در این مقاله، سال به عنوان جداکننده در نظر گرفته شده و بدین ترتیب ۷ بازه زمانی مختلف مشخص شده است.

۲-۵ معیارهای ارزیابی

معیارهای ارزیابی مورد استفاده در این مقاله عبارتند از مدولاریتی نیومن[1]، مدولاریتی با جریمه تقسیم[2] و مدولاریتی چگالی[3]. مدولاریتی نیومن برای شبکه‌های بدون وزن و بدون جهت، این نوع مدولاریتی به‌عنوان نسبت تفاوت بین تعداد واقعی و مورد انتظار یال‌ها در انجمن تعریف می‌شود که در (۱۶) نشان داده شده است

$$Q=\sum_{c_i\in C}[\frac{\left|E_{c_i}^{in}\right|}{\left|E\right|}-(\frac{۲\left|E_{c_i}^{in}\right|+\left|E_{c_i}^{out}\right|}{۲\left|E\right|})^{۲}] \qquad (۱۶)$$

در این رابطه C مجموعه همه انجمن‌ها، c_i انجمن iام، $\left|E_{c_i}^{in}\right|$ تعداد یال‌های درون انجمن، $\left|E_{c_i}^{out}\right|$ تعداد یال‌های خارج از انجمن و $\left|E\right|$ نیز نشان‌دهنده کل یال‌های موجود در شبکه است. هرچه مقدار این معیار بیشتر باشد، به معنای مناسب‌بودن انجمن‌هاست. بایستی توجه داشت که در شبکه‌های اجتماعی به دلیل پیچیده‌بودن ارتباطات و امکان وجود ارتباط بین هر دو یال، مقدار مناسب مدولاریتی حدود ۰/۵ در نظر گرفته می‌شود. یعنی در صورتی که مقدار این معیار حدود ۰/۵ باشد، به معنای مناسب‌بودن وضعیت انجمن‌ها در شبکه اجتماعی است.

معیار مدولاریتی با جریمه تقسیم، یکی دیگر از معیارهای ارزیابی انجمن است و برای بررسی کیفیت ساختار انجمن استفاده می‌شود. مدولاریتی با جریمه تقسیم (Qs) بر اساس (۱۷) به دست می‌آید. در این رابطه، Q میزان مدولاریتی و SP تعداد لبه‌های مشترک بین انجمن‌ها است و بر اساس (۱۸) محاسبه می‌گردد

$$Q_s=Q-SP \qquad (۱۷)$$

$$SP=\sum_{c_i\in C}[\sum_{\substack{c_j\in C\\ c_j\neq c_i}}\frac{\left|E_{c_i,c_j}\right|}{۲\left|E\right|}] \qquad (۱۸)$$

که در (۱۶)، $\left|E_{c_i,c_j}\right|$ تعداد لبه‌هایی است که از انجمن c_i به c_j امتداد یافتند. در این معیار نیز هرچه مقدار به‌دست‌آمده بیشتر باشد، نشان‌دهنده مناسب‌تربودن ساختار انجمن‌هاست.

1. Newman's Modularity
2. Modularity with Split Penalty
3. Modularity Density

جدول ۴: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی نیومن برای مجموعه داده CIT-HEP PH.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۵۷۷۷	۰/۵۳۸۹	۰/۶۲۸۹	۰/۶۰۵۸	۰/۵۹۲۶	**۰/۶۵۵۰**
۲	۰/۰۵۹۰	۰/۰۸۷۰	۰/۰۸۳۹	۰/۰۸۱۹	۰/۰۷۱۲	**۰/۰۸۸۲**
۳	۰/۰۶۱۹	۰/۰۳۴۰	۰/۰۳۶۶	۰/۰۴۸۰	**۰/۰۶۸۱**	۰/۰۴۰۴
۴	۰/۰۱۶۸	۰/۰۱۵۵	۰/۰۱۷۳	۰/۰۱۴۸	۰/۰۱۹۲	**۰/۰۲۲۹**
۵	۰/۰۱۵۳	۰/۰۱۰۸	۰/۰۰۵۸	۰/۰۱۱۹	**۰/۰۱۳۶**	۰/۰۱۲۴
۶	۰/۰۱۰۷	۰/۰۱۱۵	۰/۰۰۵۱	۰/۰۱۰۸	**۰/۰۱۲۲**	۰/۰۱۱۸
۷	۰/۱۱۸۱	۰/۱۲۳۱	۰/۱۱۱۵	۰/۱۴۷۵	۰/۱۷۴۲	**۰/۱۹۹۴**
۸	۰/۲۰۹۱	۰/۲۱۱۵	۰/۲۰۱۵	۰/۲۰۴۸	۰/۲۱۷۲	**۰/۲۱۸۵**
۹	۰/۲۹۲۵	۰/۲۳۸۰	۰/۲۲۳۶	۰/۲۸۴۳	۰/۳۲۷۱	**۰/۳۷۴۸**
۱۰	۰/۳۵۱۹	۰/۳۶۵۳	۰/۱۰۳۷	۰/۲۹۴۲	۰/۳۷۸۲	**۰/۴۰۵۹**
۱۱	۰/۲۵۲۳	۰/۳۰۳۵	۰/۲۲۵۵	۰/۲۷۹۱	۰/۳۲۵۰	**۰/۴۱۲۱**
میانگین	۰/۱۷۸۵	۰/۱۷۶۲	۰/۱۴۹۴	۰/۱۸۰۳	۰/۱۹۹۹	**۰/۲۲۱۹**

جدول ۵: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی نیومن برای مجموعه داده COLLEGEMSG.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۳۵۴۰	۰/۲۷۸۶	۰/۳۲۹۸	۰/۳۴۴۶	۰/۳۳۷۵	**۰/۳۸۰۵**
۲	۰/۱۰۸۶	۰/۱۲۹۲	**۰/۱۴۹۰**	۰/۱۲۵۵	۰/۱۳۹۰	۰/۱۴۶۲
۳	۰/۳۵۹۰	۰/۳۷۸۵	۰/۴۷۷۴	۰/۴۵۷۶	۰/۴۸۱۲	**۰/۴۹۴۷**
۴	۰/۲۱۰۳	۰/۲۴۵۳	۰/۲۴۴۹	۰/۲۳۹۷	۰/۲۴۸۱	**۰/۲۵۳۳**
۵	۰/۱۰۸۱	۰/۱۰۴۴	۰/۱۱۷۹	۰/۱۱۹۲	۰/۱۲۰۸	**۰/۱۳۷۹**
۶	۰/۱۵۴۹	۰/۱۷۱۲	۰/۱۵۰۵	۰/۱۵۰۲	۰/۱۵۱۹	**۰/۱۶۷۵**
۷	۰/۲۳۹۰	۰/۲۳۸۷	۰/۲۰۶۶	۰/۲۲۴۸	۰/۲۷۵۴	**۰/۲۸۸۶**
میانگین	۰/۲۱۹۱	۰/۲۲۰۸	۰/۲۳۹۴	۰/۲۳۷۴	۰/۲۵۰۶	**۰/۲۶۷۰**

جدول ۶: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی نیومن برای مجموعه داده SX-MATHOVERFLOW.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۱۵۱۲	۰/۲۰۹۰	۰/۲۱۴۳	۰/۲۰۱۹	۰/۲۲۱۲	**۰/۲۴۲۷**
۲	۰/۰۵۷۲	۰/۰۰۱۰	**۰/۰۶۸۴**	۰/۰۰۸۴	۰/۰۱۵۷	۰/۰۰۰۲
۳	۰/۱۶۰۱	۰/۲۱۳۵	**۰/۲۱۸۱**	۰/۱۸۹۲	۰/۱۷۴۱	۰/۰۰۰۵
۴	۰/۱۱۴۴	۰/۰۸۴۹	۰/۰۰۴۴	**۰/۱۲۸۶**	۰/۰۱۶۲	۰/۰۰۱۸
۵	۰/۰۶۲۱	۰/۰۴۹۳	۰/۰۵۴۱	۰/۰۶۲۷	۰/۰۸۴۶	**۰/۱۰۰۱**
۶	۰/۱۰۴۲	۰/۱۰۶۶	۰/۰۹۶۰	۰/۰۹۷۵	۰/۱۰۱۲	**۰/۱۱۸۱**
۷	۰/۰۶۱۹	۰/۱۱۱۶	۰/۰۷۱۷	۰/۰۹۱۶	۰/۱۱۰۸	**۰/۱۲۴۹**
۸	۰/۱۰۲۰	۰/۱۰۸۳	۰/۰۱۶۴	۰/۰۸۹۱	۰/۰۹۴۲	**۰/۱۱۰۹**
میانگین	۰/۱۰۱۶	۰/۱۱۰۵	۰/۰۹۲۹	۰/۱۰۸۶	۰/۱۰۲۳	۰/۰۸۷۴

هم مدولاریت نیومن (Q) و هم مدولاریت با جریمه تقسیم (Qs) مستقل از تعداد گره‌ها در انجمن‌ها هستند. مدولاریتی چگالی، گره‌ها و تراکم آنها در انجمن‌ها را نیز مورد بررسی قرار می‌دهد. برای شبکه‌های بدون جهت، چگالی مدولاریته (Qs) به‌صورت (۱۹) تعریف می‌شود

$$Q_{ds} = \sum_{c_i \in C}\left[\frac{\left|E_{c_i}^{in}\right|}{2\left|E\right|} d_{c_j} - \left(\frac{2\left|E_{c_i}^{in}\right| + \left|E_{c_i}^{out}\right|}{2\left|E\right|}\right)^2 - \sum_{\substack{c_j \in C \\ c_j \neq c_i}} \frac{\left|E_{c_i,c_j}\right|}{2\left|E\right|} d_{c_i,c_j}\right] \quad (۱۹)$$

که d_{c_i} تراکم داخلی خوشه c_i و d_{c_i,c_j} تراکم بین دو انجمن c_i و c_j بوده و با استفاده از (۲۰) محاسبه می‌شود

$$d_{c_i} = \frac{2\left|E_{c_i}^{in}\right|}{\left|c_i\right|\left(\left|c_i\right|-1\right)} \;,\; d_{c_i,d_{cj}} = \frac{\left|E_{c_i,c_j}\right|}{\left|c_i\right|\left|c_j\right|} \quad (۲۰)$$

مشابه با معیارهای قبل، در این معیار نیز بیشتربودن مقدار به‌دست‌آمده به معنای مناسب‌تربودن ساختار انجمن‌هاست. معیار زمان به بررسی مدت زمان مورد نیاز بر حسب ثانیه برای اجرای روش‌های تشخیص انجمن بر روی یک سیستم می‌پردازد. به‌طور طبیعی هرچه این مقدار کمتر باشد، بهتر است.

۳-۵ نتایج آزمایش‌ها

جداول ۴ تا ۶ نشان‌دهنده نتایج حاصل از معیار مدولاریتی نیومن برای روش‌های مورد مقایسه روی مجموعه داده‌های مختلف می‌باشد.

طبق نتایج جدول ۴، روش پیشنهادی در مجموعه داده cit-Hep Ph، در ۸ بازه زمانی از ۱۱ بازه زمانی و همچنین به طور میانگین بهترین مقدار معیار مدولاریتی نیومن را نسبت به بقیه روش‌ها دارد و فقط در بازه‌های زمانی ۳، ۵ و ۶ دو روش CD-SACS و DPC-DLP بهترین مقدار را

جدول ۷: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی با جریمه تقسیم برای مجموعه داده CIT-HEP PH.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۱۳۰۵	۰/۱۹۵۲	۰/۱۲۰۷	۰/۱۳۰۹	۰/۱۴۵۲	**۰/۱۹۶۲**
۲	**۰/٤٥٦٥**	۰/۴۱۲۴	۰/۴۳۷۳	۰/۴۲۲۷	۰/۴۴۱۹	۰/۴۳۳۵
۳	۰/۴۲۷۷	۰/۴۲۶۰	۰/۴۶۹۵	۰/۴۵۴۲	**۰/٤٧٢٦**	۰/۴۶۶۳
۴	۰/۳۸۳۷	۰/۳۸۴۷	۰/۴۸۵۹	۰/۴۰۲۹	۰/۴۵۹۱	**۰/٤٨٨٢**
۵	۰/۴۷۴۹	۰/۳۸۸۲	**۰/٤٩٥١**	۰/۴۶۶۹	۰/۴۷۸۳	۰/۴۸۹۶
۶	۰/۴۶۲۱	۰/۴۵۰۰	۰/۴۸۵۶	۰/۴۵۷۲	۰/۴۸۰۵	**۰/٤٩٠٠**
۷	۰/۱۳۴۱	۰/۱۰۱۶	۰/۱۰۴۸	۰/۱۴۸۴	**۰/۱۵۱۱**	۰/۱۴۳۸
۸	۰/۳۹۵۳	۰/۴۲۲۷	۰/۴۹۸۷	**۰/۵۰۰۸**	۰/۴۸۹۱	۰/۴۹۲۹
۹	۰/۴۶۵۷	۰/۴۴۶۲	۰/۴۸۶۸	۰/۴۶۶۲	۰/۴۷۵۸	**۰/٤٩١١**
۱۰	۰/۳۹۸۲	۰/۴۴۶۹	۰/۴۸۶۷	۰/۴۶۷۱	۰/۴۷۴۹	**۰/٤٩١١**
۱۱	۰/۰۹۶۲	۰/۱۳۱۴	۰/۰۱۵۶	۰/۱۴۱۲	**۰/۱۷۳۲**	۰/۱۵۳۷
میانگین	۰/۳۴۷۷	۰/۳۴۵۹	۰/۳۷۱۵	۰/۳۶۹۰	۰/۳۸۵۶	**۰/۳۹٤۲**

جدول ۸: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی با جریمه تقسیم برای مجموعه داده COLLEGEMSG.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۳۹۹۵	۰/۳۹۹۵	۰/۴۹۱۸	۰/۴۵۲۹	۰/۴۷۷۲	**۰/٤٩٩٢**
۲	۰/۴۱۵۸	۰/۴۹۰۴	۰/۵۰۰۷	۰/۴۶۹۷	۰/۴۹۳۱	**۰/۵۰۱۰**
۳	۰/۴۸۴۵	۰/۴۳۰۷	۰/۳۹۹۲	۰/۴۶۵۳	۰/۴۸۱۶	**۰/٤٩٩٨**
۴	۰/۴۲۴۵	۰/۳۲۴۵	۰/۴۹۹۳	۰/۴۷۶۸	**۰/۵۰٤۲**	۰/۴۹۸۸
۵	۰/۰۱۱۷	۰/۰۳۴۴	**۰/۰٤٨٥**	۰/۰۳۵۹	۰/۰۴۱۱	۰/۰۴۰۸
۶	۰/۰۳۸۱	۰/۰۳۳۹	۰/۰۵۴۴	۰/۰۴۵۳	۰/۰۵۸۲	**۰/۰۷٤٦**
۷	۰/۱۳۲۱	۰/۱۳۶۳	۰/۰۵۸۸	۰/۰۵۰۴	۰/۰۵۹۱	**۰/۰٦٨٦**
میانگین	۰/۲۷۲۳	۰/۲۶۴۲	۰/۲۹۳۲	۰/۲۸۵۲	۰/۳۰۲۱	**۰/۳۱۱۸**

جدول ۹: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی با جریمه تقسیم برای مجموعه داده SX-MATHOVERFLOW.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۳۲۶۸	۰/۲۷۲۲	۰/۱۸۲۹	۰/۳۳۴۶	۰/۴۵۹۳	**۰/٥٦٦١**
۲	۰/۱۲۱۲	۰/۰۴۱۷	۰/۰۰۰۲	۰/۰۵۷۳	**۰/۱۲۸۸**	۰/۰۰۰۲
۳	۰/۰۰۰۵	۰/۰۰۰۴	**۰/۰۰۰۸**	۰/۰۰۰۳	۰/۰۰۰۵	۰/۰۰۰۵
۴	۰/۰۰۱۲	۰/۰۰۱۳	۰/۰۰۳۱	**۰/۰۰٤٧**	۰/۰۰۲۵	۰/۰۰۱۸
۵	۰/۱۶۰۰	۰/۱۵۹۰	۰/۱۳۲۱	۰/۱۵۴۸	۰/۱۶۶۸	**۰/۱۷۵۰**
۶	۰/۱۰۹۰	۰/۱۷۴۷	۰/۰۱۰۱	۰/۱۶۱۲	۰/۱۹۰۶	**۰/۲۰۸۰**
۷	۰/۰۱۸۸	۰/۱۳۹۳	۰/۰۱۴۶	۰/۱۵۴۹	۰/۱۳۶۲	**۰/۱۵۷۷**
۸	۰/۰۱۰۴	۰/۰۰۲۳	۰/۰۱۰۴	۰/۰۰۹۴	۰/۰۱۰۵	**۰/۰۱۰۹**
میانگین	۰/۰۹۳۵	۰/۰۹۸۹	۰/۰۴۴۳	۰/۱۰۹۷	۰/۱۳۶۹	**۰/۱٤۰۰**

دارند که به دلیل تعداد تغییرات متفاوت (حذف و اضافه شدن یال‌ها و نودها) در بازه‌های زمانی مختلف است. به طور میانگین بعد از روش پیشنهادی، روش‌های CD-SACS و DynCRep بهترین مقدار معیار مدولاریتی نیومن در مجموعه داده cit-Hep Ph را به دست آورده‌اند. بر اساس نتایج جدول ۵، روش پیشنهادی در مجموعه داده CollegeMsg در شش بازه زمانی و به طور میانگین بهترین مقدار معیار مدولاریتی نیومن را نسبت به بقیه روش‌ها دارد و فقط در بازه زمانی دو، روش louvain بهترین مقدار را دارد که اختلاف آن با روش پیشنهادی خیلی اندک و قابل قبول است. به طور میانگین بعد از روش پیشنهادی، روش‌های CD-SACS و louvain بهترین مقدار معیار مدولاریتی نیومن در مجموعه داده CollegeMsg را به دست آورده‌اند. بر اساس نتایج جدول ۶، روش پیشنهادی در مجموعه داده sx-mathoverflow در ۵ بازه زمانی از هشت بازه بهترین مقدار معیار مدولاریتی نیومن را نسبت به بقیه روش‌ها دارد و در سه بازه زمانی ۲، ۳ و ۴ روش‌های Louvain و DynCRep بهترین مقدار را دارند که به دلیل تعداد کم تغییرات یال‌ها و نودها در این بازه‌های زمانی است. به طور میانگین، روش‌های IncNSA، DynCRep و CD-SACS بهترین مقدار معیار مدولاریتی نیومن در مجموعه داده sx-mathoverflow را به دست آورده‌اند. همان طور که مشاهده می‌شود، طبق نتایج به‌دست‌آمده از جداول ۴ تا ۶، روش پیشنهادی در اکثر بازه‌های زمانی هر مجموعه داده، عملکرد بهتری نسبت به سایر روش‌های مورد مقایسه داشته است.

نتایج به‌دست‌آمده برای معیار مدولاریتی با جریمه تقسیم به تفکیک مجموعه داده در جداول ۷ تا ۹ آمده است.

طبق نتایج جدول ۷، روش پیشنهادی در مجموعه داده cit-Hep Ph، در ۵ بازه زمانی و همچنین به طور میانگین بهترین مقدار معیار مدولاریتی با جریمه تقسیم را نسبت به بقیه روش‌ها دارد. به طور میانگین بعد از

جدول ۱۰: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی چگالی برای مجموعه داده CIT-HEP PH.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۴۷۲۴	۰/۴۷۹۶	۰/۴۷۴۰	۰/۴۶۳۱	۰/۴۶۷۹	**۰/۵۱۳۵**
۲	**۰/۱۹۷۲**	۰/۱۴۰۳	۰/۰۶۴۷	۰/۰۷۲۱	۰/۰۷۹۲	۰/۰۷۰۸
۳	۰/۱۲۸۴	۰/۰۶۰۹	۰/۰۳۰۸	۰/۱۰۲۱	**۰/۱۳۱۷**	۰/۰۹۵۱
۴	۰/۱۱۸۳	۰/۱۰۸۶	۰/۰۱۵۳	۰/۰۸۵۹	۰/۱۱۶۵	**۰/۱۲۱۰**
۵	۰/۰۶۳۲	۰/۰۵۹۰	۰/۰۰۵۱	۰/۱۰۷۸	**۰/۱۲۴۲**	۰/۱۱۱۸
۶	۰/۰۵۸۶	۰/۱۲۶۱	۰/۰۰۱۰	۰/۰۷۶۳	۰/۱۱۴۶	**۰/۱۲۸۱**
۷	۰/۲۳۹۴	۰/۱۵۰۵	۰/۱۸۱۳	۰/۱۹۴۶	۰/۲۲۵۹	**۰/۲۵۷۷**
۸	۰/۱۴۶۶	۰/۱۴۱۲	۰/۱۰۰۹	۰/۱۴۱۹	۰/۱۵۳۷	**۰/۱۶۷۹**
۹	۰/۰۷۰۶	۰/۰۴۳۳	۰/۱۰۱۲	۰/۱۴۵۳	۰/۱۹۳۷	**۰/۲۰۸۲**
۱۰	۰/۱۳۱۷	۰/۱۰۰۱	۰/۱۰۳۲	۰/۱۴۶۷	۰/۱۷۶۲	**۰/۲۱۰۰**
۱۱	۰/۱۶۹۰	۰/۲۱۶۵	۰/۱۳۳۴	۰/۱۸۴۸	۰/۲۰۵۹	**۰/۲۲۳۳**
میانگین	۰/۱۶۳۲	۰/۱۴۷۸	۰/۱۱۰۱	۰/۱۵۶۴	۰/۱۸۰۹	**۰/۱۹۱۶**

جدول ۱۱: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی چگالی برای مجموعه داده COLLEGEMSG.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۱۷۰۶	۰/۱۸۶۸	**۰/۲۱۹۵**	۰/۱۸۲۰	۰/۱۹۸۳	۰/۲۰۰۲
۲	۰/۰۰۰۲	۰/۰۰۰۸	۰/۰۰۰۷	۰/۰۰۱۷	۰/۰۰۱۹	**۰/۰۰۳۰**
۳	۰/۰۰۰۱	۰/۰۰۰۵	۰/۰۰۰۵	۰/۰۰۱۴	۰/۰۰۱۵	**۰/۰۰۲۱**
۴	۰/۰۰۰۵	۰/۰۰۰۲	۰/۰۰۰۲	۰/۰۰۰۵	۰/۰۰۰۵	**۰/۰۰۰۸**
۵	۰/۲۴۳۵	۰/۱۹۰۸	**۰/۳۷۳۸**	۰/۳۵۱۳	۰/۳۶۷۲	۰/۳۳۹۳
۶	۰/۱۲۵۸	۰/۱۲۰۱	۰/۱۲۹۵	۰/۱۵۴۶	۰/۱۸۷۵	**۰/۲۰۸۱**
۷	۰/۱۶۲۶	۰/۲۰۴۰	۰/۲۰۶۲	۰/۲۲۲۷	۰/۲۵۶۱	**۰/۲۸۶۹**
میانگین	۰/۱۰۰۵	۰/۱۰۰۵	۰/۱۳۲۹	۰/۱۳۰۶	۰/۱۴۴۷	**۰/۱۴۸۶**

جدول ۱۲: مقایسه نتایج به‌دست‌آمده با معیار مدولاریتی چگالی برای مجموعه داده SX-MATHOVERFLOW.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۰/۳۲۶۸	۰/۲۷۲۲	۰/۱۸۲۹	۰/۳۴۱۹	۰/۴۴۲۷	**۰/۵۶۶۱**
۲	۰/۱۲۱۲	۰/۰۴۱۷	۰/۰۰۰۲	۰/۰۵۲۷	**۰/۱۶۱۲**	۰/۰۰۰۲
۳	۰/۰۰۰۴	۰/۰۰۰۴	**۰/۰۰۰۸**	۰/۰۰۰۴	۰/۰۰۰۵	۰/۰۰۰۵
۴	۰/۰۰۱۱	۰/۰۰۱۰	**۰/۰۰۳۱**	۰/۰۰۱۶	۰/۰۰۱۹	۰/۰۰۱۸
۵	۰/۰۶۰۰	۰/۰۵۹۰	۰/۱۳۲۱	**۰/۱۳۴۹**	۰/۱۲۸۶	۰/۰۷۵۰
۶	۰/۰۰۹۰	۰/۰۷۴۷	۰/۰۰۰۱	۰/۱۵۶۳	۰/۱۸۷۲	**۰/۲۰۸۰**
۷	۰/۰۱۸۸	۰/۱۰۹۳	۰/۰۱۴۶	۰/۱۱۷۹	۰/۱۴۰۵	**۰/۱۵۷۷**
۸	۰/۰۰۴۹	۰/۰۰۳۸	**۰/۰۱۶۴**	۰/۰۱۰۲	۰/۰۱۲۳	۰/۰۱۰۹
میانگین	۰/۰۶۷۸	۰/۰۷۰۳	۰/۰۴۳۸	۰/۱۰۲۰	**۰/۱۳۴۴**	۰/۱۲۷۵

روش پیشنهادی، روش‌های CD-SACS و louvain بهترین مقدار معیار مدولاریتی با جریمه تقسیم در مجموعه داده cit-Hep Ph را به دست آورده‌اند. بر اساس نتایج جدول ۸، روش پیشنهادی در مجموعه داده CollegeMsg در ۵ بازه زمانی و به طور میانگین بهترین مقدار معیار مدولاریتی با جریمه تقسیم را نسبت به بقیه روش‌ها دارد و فقط در بازه‌های زمانی ۴ و ۵، روش‌های CD-SACS و louvain بهترین مقدار را دارند که اختلاف آنها با روش پیشنهادی خیلی اندک و قابل قبول است. به طور میانگین بعد از روش پیشنهادی، روش‌های CD-SACS و louvain بهترین مقدار معیار مدولاریتی با جریمه تقسیم در مجموعه داده CollegeMsg را به دست آورده‌اند. بر اساس نتایج جدول ۹، روش پیشنهادی در مجموعه داده sx-mathoverflow در ۵ بازه زمانی از ۸ بازه بهترین مقدار معیار مدولاریتی با جریمه تقسیم را نسبت به بقیه روش‌ها دارد و در سه بازه زمانی ۲، ۳ و ۴ روش‌های CD-SACS، Louvain و DynCRep بهترین مقدار را دارند که به دلیل تعداد کم تغییرات یال‌ها و نودها در این بازه‌های زمانی است. به طور میانگین، روش‌های IncNSA و CD-SACS بهترین مقدار معیار مدولاریتی با جریمه تقسیم در مجموعه داده sx-mathoverflow را به دست آورده‌اند. طبق نتایج به‌دست‌آمده از جداول ۷ تا ۹ برای اکثر مجموعه داده‌ها و بازه‌های زمانی، روش ارائه‌شده بر اساس معیار مدولاریتی با جریمه تقسیم بهتر عمل نموده است.

نتایج به‌دست‌آمده برای معیار مدولاریتی چگالی نیز به تفکیک مجموعه داده در جدول‌های ۱۰ تا ۱۲ آمده است.

طبق نتایج جدول ۱۰، روش پیشنهادی در مجموعه داده cit-Hep Ph در ۸ بازه زمانی و همچنین به طور میانگین بهترین مقدار معیار مدولاریتی چگالی را نسبت به بقیه روش‌ها دارد و فقط در بازه‌های زمانی ۲، ۳ و ۵ روش‌های DPC-DLP،CD-SACS بهترین مقدار را دارند. به طور

جدول ۱۳: نتایج به‌دست‌آمده برای زمان برای مجموعه داده CIT-HEP PH.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۳٫۲۲۳	۲٫۴۸۶	۰٫۰۷۹	۲٫۷۶۹	۵٫۱۲۷	**۰٫۰۷۸**
۲	۶٫۲۴۲	۸٫۳۶۱	۴٫۴۵۱	۵٫۵۲۷	۹٫۴۴۱	**۳٫۶۷۵**
۳	۶۴٫۸۹۳	۶۳٫۱۳۶	۵۹٫۸۴۵	۶۳٫۰۰۲	۶۹٫۴۲۱	**۱۷٫۱۳۰**
۴	۹۴٫۱۸۲	۹۳٫۵۳۷	۹۱٫۶۳۷	۹۵٫۱۲۸	۱۰۸٫۱۴۲	**۵۱٫۸۶۵**
۵	۱۲۹٫۰۶۱	۱۲۸٫۵۸۰	۱۲۶٫۵۳۸	۱۲۹٫۹۳۸	۱۴۱٫۰۱۸	**۹۵٫۶۶۲**
۶	۱۸۵٫۹۰۳	۱۸۷٫۰۵۷	۱۸۲٫۶۳۷	۱۸۵٫۳۶۷	۱۹۵٫۶۲۸	**۱۴۶٫۶۶۱**
۷	۲۴۸٫۰۷۸	۲۵۰٫۷۱۷	۲۴۶٫۵۷۵	۲۴۹٫۱۸۶	۲۶۱٫۳۵۷	**۲۵۰٫۲۴۰**
۸	۱۸۳٫۵۴۳	۱۸۴٫۴۲۶	۱۸۱٫۵۳۴	۱۸۲٫۳۵۲	۲۰۱٫۹۱۴	**۱۴۵٫۶۵۸**
۹	۱۹۴٫۶۸۶	۱۹۶٫۲۸۶	۱۹۱٫۵۲۷	۱۹۳٫۴۲۹	۲۱۷٫۰۶۵	**۱۸۲٫۰۱۵**
۱۰	۲۸۲٫۸۰۷	۲۸۶٫۵۲۶	۲۸۱٫۵۱۳	۲۸۸٫۳۷۵	۳۰۹٫۱۲۸	**۱۷۲٫۹۱۱**
۱۱	۱۲۷٫۱۱۹	۱۲۷٫۸۵۸	۱۲۴٫۵۷۴	۱۲۵٫۹۵۲	۱۴۲٫۱۷۵	**۹۳٫۸۷۶**
میانگین	۱۳۸٫۱۵۸	۱۳۸٫۹۹۷	۱۳۵٫۵۳۷	۱۳۸٫۲۷۵	۱۵۰٫۹۴۷	**۱۱۴٫۵۲۵**

جدول ۱۴: نتایج به‌دست‌آمده برای زمان برای مجموعه داده COLLEGEMSG.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۱٫۶۵۲۵	۱٫۳۳۴۸	**۰٫۳۳۴۴**	۰٫۵۸۲۷	۲٫۷۶۲۸	۰٫۷۶۵۶
۲	۱۱٫۰۹۸۱	۲۰٫۹۷۴۹	۱۹٫۶۵	۲۱٫۱۲۷۵	۲۹٫۱۶۷۳	**۱۸٫۳۲۷۸**
۳	۴۱٫۵۸۳۲	۴۲٫۰۸۲۴	۴۰٫۹۱۳	۴۱٫۵۲۸۱	۴۷٫۳۶۷۴	**۳۷٫۶۵۵۶**
۴	۱۸٫۳۸۳۹	۱۴٫۷۹۰۰	۱۴٫۴۱۶۳	۱۵٫۰۷۵۳	۲۱٫۲۴۷۵	**۱۳٫۵۶۷۳**
۵	۴٫۸۰۵۶	۳٫۶۳۷۷	**۳٫۱۰۸۵**	۵٫۱۱۴۲	۷٫۱۳۴۷	۳٫۲۸۷۵
۶	۲٫۶۰۴۲	۱٫۷۴۶۷	**۱٫۰۶۱۷**	۱٫۸۹۹۱	۳٫۱۷۶۴	۱٫۴۲۶۸
۷	۱٫۵۳۰۲	۰٫۶۶۳۰	**۰٫۲۵۹۴**	۱٫۴۶۵۵	۳٫۲۰۴۶	۰٫۶۹۷۴
میانگین	۱۱٫۶۶۵۳۹	۱۲٫۱۷۵۶۴	۱۱٫۳۹۱۹	۱۲٫۳۹۸۹۱	۱۶٫۲۹۴۳	**۱۰٫۸۱۸۲۹**

جدول ۱۵: نتایج به‌دست‌آمده برای زمان برای مجموعه داده SX-MATHOVERFLOW.

برچسب زمانی	DPC-DLP	IncNSA	louvain	DynCRep	CD-SACS	روش پیشنهادی
۱	۲۳٫۴۳	۲۳٫۰۰	۲۱٫۹۶	۲۵٫۱۱	۳۱٫۱۴	**۱۴٫۳۸**
۲	۴۹۱٫۸۹	۵۱۹٫۹۰	۴۶۰٫۵۰	۴۹۵٫۸۷	۵۹۲٫۱۷	**۳۰۰٫۸۹**
۳	۱۸۷۱٫۸۶	۱۳۷۱٫۳۱	۱۵۱۵٫۳۵	۱۵۰۱٫۹۴	۱۹۱۲٫۰۱	**۹۹۷٫۶۸**
۴	۱۴۰۹٫۸۹	۱۴۲۵٫۱۴	۱۸۰۴٫۸۳	۱۶۴۲٫۴۴	۲۰۵۳٫۸۳	**۱۱۹۸٫۷۰**
۵	۱۶۱۲٫۸۹	۱۵۷۵٫۰۲	۱۵۷۳٫۹۷	۱۴۱۲٫۲۱	۲۰۶۳٫۶۲	**۱۰۴۸٫۲۱**
۶	۱۵۰۵٫۳۸	۱۵۰۴٫۸۳	۱۵۰۳٫۹۲	۱۵۸۶٫۱۰	۱۹۲۷٫۴۲	**۱۰۰۲٫۱۳**
۷	۱۴۸۰٫۹۶	۱۴۸۰٫۲۱	۱۴۷۹٫۴۰	۱۵۰۷٫۰۷	۱۷۵۳٫۴۳	**۹۸۶٫۰۳**
۸	۸۱۱٫۱۴	۱۰۱۲٫۵۹	۱۱۸۴٫۱۳	۱۰۹۲٫۱۳	۱۳۷۹٫۰۲	**۱۴٫۳۸**
میانگین	۱۱۵۰٫۹۳	۱۱۱۴٫۰۰	۱۱۹۳٫۰۱	۱۱۵۷٫۸۶	۱۴۶۴٫۰۸	**۳۰۰٫۸۹**

میانگین بعد از روش پیشنهادی، روش‌های CD-SACS و DPC-DLP بهترین مقدار معیار مدولاریتی چگالی در مجموعه داده cit-Hep Ph را به دست آورده‌اند. بر اساس نتایج جدول ۱۱، روش پیشنهادی در مجموعه داده CollegeMsg در ۵ بازه زمانی و به طور میانگین بهترین مقدار معیار مدولاریتی چگالی را نسبت به بقیه روش‌ها دارد و فقط در بازه‌های زمانی ۱ و ۵، روش louvain بهترین مقدار را دارند که اختلاف آنها با روش پیشنهادی خیلی اندک و قابل قبول است. به طور میانگین بعد از روش پیشنهادی، روش‌های CD-SACS و louvain بهترین مقدار معیار مدولاریتی چگالی در مجموعه داده CollegeMsg را به دست آورده‌اند. مطابق با نتیجه‌های جدول ۱۲، روش پیشنهادی در مجموعه داده sx-mathoverflow در ۳ بازه زمانی بهترین مقدار معیار مدولاریتی چگالی را نسبت به بقیه روش‌ها دارد و در بقیه بازه‌های زمانی، روش‌های CD-SACS، Louvain و DynCRep بهترین مقدار را دارند.

نتایج به‌دست‌آمده برای معیار زمان برحسب ثانیه نیز به تفکیک مجموعه داده در جدول‌های ۱۳ تا ۱۵ آمده است. با در نظر گرفتن نتایج به‌دست‌آمده ملاحظه می‌شود که روش پیشنهادی عملکرد مناسب‌تری را در قیاس با روش‌های مورد مقایسه روی مجموعه داده‌های معرفی‌شده و با معیارهای ارزیابی مختلف گزارش نموده است.

۵-۴ تأثیر هم‌پوشانی در روش پیشنهادی

با توجه به اینکه انجمن‌ها و گروه‌های شکل‌یافته در یک شبکه در دنیای واقعی می‌توانند با هم همپوشانی داشته باشند، هدف اصلی در این پژوهش، ارائه روشی برای مواجهه با همپوشانی انجمن‌هاست که طی آن سعی می‌شود تا انجمن‌هایی که همپوشانی داشته باشند، حفظ شوند و اعضای آنها بتوانند به هر دو انجمن دارای همپوشانی تعلق داشته باشند. در جدول ۱۶ میانگین معیارهای ارزیابی مدولاریتی نیومن، مدولاریتی با

جدول ۱۶: نتایج به‌دست‌آمده برای روش پیشنهادی با در نظر گرفتن همپوشانی نسبت به در نظر نگرفتن همپوشانی.

مجموعه داده	Cit-Hep Ph			sx-mathoverflow			CollegeMsg		
مدولاریتی	نیومن	جریمه تقسیم	چگالی	نیومن	جریمه تقسیم	چگالی	نیومن	جریمه تقسیم	چگالی
با همپوشانی	۰/۲۲۱۹	۰/۳۹۴۲	۰/۱۹۱۶	۰/۰۸۷۴	۰/۱۴۰۰	۰/۱۲۷۵	۰/۲۶۷۰	۰/۳۱۱۸	۰/۱۴۸۶
بدون همپوشانی	**۰/۱۶۴۷**	**۰/۳۱۴۰**	**۰/۱۴۴۲**	**۰/۰۷۱۱**	**۰/۱۱۹۳**	**۰/۰۹۸۵**	**۰/۲۳۳۲**	**۰/۲۷۷۴**	**۰/۱۱۹۷**

جدول ۱۷: نمایش زمان اجرای روش پیشنهادی بدون موازی‌سازی نسبت به نسخه موازی‌سازی‌شده.

مجموعه داده	Cit-Hep Ph	sx-mathoverflow	CollegeMsg
بدون موازی‌سازی	۱۱۴/۵۲۵	۷۹۲/۱۸۵۴	**۱۰/۸۱۸۲۹**
با موازی‌سازی	**۸۹/۵۸۷۴**	**۴۸۲/۱۴۷۵**	۱۲/۸۴۶۳

جریمه تقسیم و مدولاریتی چگالی روش پیشنهادی با در نظر گرفتن همپوشانی نسبت به در نظر نگرفتن همپوشانی در بازه‌های زمانی مجموعه داده‌ها Cit-Hep Ph، sx-mathoverflow و CollegeMsg نشان داده شده است.

همان طور که در جدول ۱۶ مشاهده می‌شود، روش پیشنهادی با همپوشانی بر روی همه مجموعه‌های داده میانگین معیارهای ارزیابی بهتری نسبت به روش پیشنهادی بدون همپوشانی ارائه می‌دهد.

۵-۵ تأثیر موازی‌سازی روش پیشنهادی

یکی از چالش‌های اصلی در ارتباط با روش‌های کشف انجمن، قابلیت موازی‌سازی است. از آنجا که این الگوریتم‌ها معمولاً روی گراف‌هایی با تعداد گره‌ها و یال‌های بسیار بالا اجرا می‌گردد، لازم است الگوریتم‌ها قابلیت موازی‌سازی داشته باشند. در جدول ۱۷ میانگین زمان اجرای روش پیشنهادی بدون موازی‌سازی نسبت به نسخه موازی‌سازی‌شده در بازه‌های زمانی مجموعه داده‌ها Cit-Hep Ph، sx-mathoverflow و CollegeMsg نشان داده شده است.

همان طور که در جدول ۱۷ مشاهده می‌شود، نسخه موازی‌سازی‌شده روش پیشنهادی بر روی مجموعه‌های داده Cit-Hep Ph و sx-mathoverflow که بزرگ‌تر از CollegeMsg هستند، میانگین زمان اجرای کمتری نسبت به روش پیشنهادی بدون موازی‌سازی ارائه می‌دهد؛ اما در شبکه کوچک CollegeMsg زمان اجرا به دلیل سربار موازی‌سازی بیشتر هم می‌شود. میزان کاهش زمان اجرای روش پیشنهادی برای sx-mathoverflow بیش از ۳۹٪ و برای Cit-Hep Ph نزدیک به ۲۲٪ بهبود گزارش نمود. در حالی که برای CollegeMsg به دلیل اینکه کوچک‌بودن و افزایش سربار ناشی از موازی‌سازی افزایش زمان اجرا را شاهد بودیم.

۶- نتیجه‌گیری

تشخیص انجمن در بسیاری از کاربردها مانند اینترنت، شبکه‌های اجتماعی، شبکه‌های موبایل، پزشکی و سیستم‌های مراقبت از سلامت، مدیریت ارتباط با مشتری و ... کاربرد دارد. رشد داده‌های موجود در بستر اینترنت از یک سو و کاربردهای تجاری از سوی دیگر سبب گسترش شبکه‌های اجتماعی و نیاز به تشخیص انجمن در آنها شده است. با این حال، پیچیدگی محساباتی بالا، ذات پویایی این شبکه‌ها، در نظر گرفتن تأثیر گره‌ها در شناسایی انجمن و همپوشانی بین انجمن‌ها از جمله مهم‌ترین چالش‌های حل این مسئله است. در این مقاله روشی ارائه نمودیم که برای تمامی چالش‌های ذکرشده راه حل ارائه می‌نماید. بر اساس نتایج به‌دست‌آمده، الگوریتم پیشنهادی توانسته که در اغلب موارد در معیارهای ماژولاریتی، نتایج مناسب‌تری را نسبت به الگوریتم‌های مورد مقایسه کسب کند. دلیل این امر می‌تواند استفاده از گره‌های تأثیرگذار در تولید انجمن باشد. در الگوریتم پیشنهادی در زمان تعیین گره‌های تأثیرگذار از دو معیار شامل تعداد همسایگی و تجزیه $k-shell$ استفاده شده که هر دو درجه گره را هدف قرار می‌دهند؛ لذا انجمن سعی می‌کند تا حول گرهی شکل یابد که بیشترین ارتباط را با سایر گره‌های دیگر داشته باشد. از سوی دیگر در روش پیشنهادی سعی شده تا هر گرهی که پتانسیل تولید انجمن را دارد، یعنی دارای تأثیرگذاری بالاست و یا فاقد همسایگی با انجمن‌های جاری است، انجمن خود را شکل دهد. نهایتاً انجمن‌هایی که بر اساس دو معیار میزان برازندگی و نرخ همپوشانی توانایی ادغام دارند با یکدیگر ادغام می‌شوند؛ در نتیجه در روش پیشنهادی، تنها انجمن‌هایی باقی می‌مانند که از یک سو توانایی شکل‌دهی یک انجمن را داشته و از سوی دیگر، توانایی ادغام با سایر انجمن‌های دیگر را نداشته باشند. در صورتی که در سایر روش‌ها سعی شده تا از معیار ماژولایتی پایه به‌عنوان یک معیار برای تصمیم‌گیری در خصوص ادغام دو انجمن یا تشکیل یک انجمن جدید استفاده شود. از سوی دیگر در خصوص اینکه کدام گره‌ها به عنوان مرکز انجمن شناخته شوند، ایده یا تصمیم منسجم و مناسبی وجود ندارد. تنها در صورتی که الگوریتم تشخیص دهد که با جداسازی یک یا چند گره، معیار ماژولایتی پایه بهبود می‌یابد، آنگاه عملیات تشکیل انجمن جدید صورت می‌گیرد. علاوه بر این، الگوریتم در فاز ارزیابی و ادغام انجمن‌ها به یال‌های بین انجمنی رسیدگی کرده و تا حد امکان انجمن‌های دارای همپوشانی را ادغام کرده است؛ لذا در خصوص معیار ماژولایتی، الگوریتم پیشنهادی به نسبت سایر روش‌ها نتایج مطلوب‌تری را کسب کرده است. موازی‌سازی الگوریتم در پرهزینه‌ترین گام الگوریتم یعنی محاسبات مربوط به روش $k-shell$ نقش مهمی در کاهش زمان اجرای روش پیشنهادی دارد. نکته مهم اینکه هرچه مقیاس شبکه بزرگ‌تر شود، میزان کاهش زمان اجرا قابل توجه‌تر خواهد بود. برای شبکه‌های کوچک سربار موازی‌سازی نه‌تنها باعث کاهش زمان اجرا نمی‌شود، بلکه سرعت آن را کاهش می‌دهد.

مراجع

[1] A. Reihanian, M. R. Feizi-Derakhshi, and H. S. Aghdasi, "An enhanced multi-objective biogeography-based optimization for overlapping community detection in social networks with node attributes," *Information Sciences*, vol. 622, pp. 903-929, Apr. 2023.

[2] S. K. Gupta and D. P. Singh, "Seed community identification framework for community detection over social media," *Arab. J. Sci. Eng.*, vol. 48, no. 2, pp. 1829-1843, 2023.

[3] S. Mishra, S. S. Singh, S. Mishra, and B. Biswas, "Multi-objective based unbiased community identification in dynamic social networks," *Comput. Commun.*, vol. 214, pp. 18-32, 15 Jan. 2024.

[4] M. Sabzekar, S. BaradaranNejad, and M. Khazaeipoor, "A node-centric approach for community detection in dynamic networks," *J. Electr. Comput. Eng. Innov.*, vol. 12, no. 2, pp. 305-318, Jul. 2024.

[5] X. Li, Y. Xin, C. Zhao, Y. Yang, and Y. Chen, "Graph convolutional networks for privacy metrics in online social networks," *Applied Sciences*, vol. 10, no. 4, Article ID: 1327, 2020.

[6] Z. Ma and S. Nandy, "Community detection with contextual multilayer networks," *IEEE Trans. Inf. Theory*, vol. 69, no. 5, pp. 3203-3239, May 2023.
[7] M. Rostami, M. Oussalah, K. Berahmand, and V. Farrahi, "Community detection algorithms in healthcare applications: a systematic review," *IEEE Access*, vol. 11, pp. 30247-30272, 2023.
[8] F. S. Gharehchopogh, "An improved harris hawks optimization algorithm with multi-strategy for community detection in social network," *J. Bionic Eng.*, vol. 20, pp. 1175-1197, 2023.
[9] D. Jin, *et al.*, "A survey of community detection approaches: from statistical modeling to deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 2, pp. 1149-1170, Feb. 2023.
[10] M. Seifikar, S. Farzi, and M. Barati, "C-Blondel: an efficient louvain-based dynamic community detection algorithm," *IEEE Trans. Comput. Soc. Syst.*, vol. 7, no. 2, pp. 308-318, Apr. 2020.
[11] S. Ahajjam, M. El Haddad, and H. Badir, "A new scalable leader-community detection approach for community detection in social networks," *Soc. Networks*, vol. 54, pp. 41-49, Jul. 2018.
[12] S. Bahadori, H. Zare, and P. Moradi, "PODCD: probabilistic overlapping dynamic community detection," *Expert Syst. Appl.*, vol. 174, Article ID: 114650, 15 Jul. 2021.
[13] C. Bothorel, V. L. Dao, and P. Lenca, "Community structure: a comparative evaluation of community detection methods," *Netw. Sci.*, vol. 8, no. 1, pp. 1-41, Mar. 2020.
[14] S. Fortunato, "Community detection in graphs," *Phys. Rep.*, vol. 486, no. 3-5, pp. 75-174, Feb. 2010.
[15] M. E. J. Newman, "Modularity and community structure in networks," in *Proc. Natl. Acad. Sci*vol. 103, no. 23, pp. 8577-8582, Jun. 2006.
[16] W. Li, X. Zhou, C. Yang, Y. Fan, Z. Wang, and Y. Liu, "Multi-objective optimization algorithm based on characteristics fusion of dynamic social networks for community discovery," *Inf. Fusion*, vol. 79, no. C, pp. 110-123, Mar. 2022.
[17] Q. Ni, J. Guo, W. Wu, and H. Wang, "Influence-based community partition with sandwich method for social networks," *IEEE Trans. Comput. Soc. Syst.*, vol. 10, no. 2, pp. 819-830, Apr. 2023.
[18] H. Long, X. Li, X. Liu, and W. Wang, "BBTA: detecting communities incrementally from dynamic networks based on tracking of backbones and bridges," *Appl. Intell.*, vol. 53, pp. 1084-1100, 2023.
[19] H. Safdari, M. Contisciani, and C. De Bacco, "Reciprocity, community detection, and link prediction in dynamic networks," *J. Phys. Complex.*, vol. 3, no. 1, Article ID: 15010, 2022.
[20] S. Kumar, A. Mallik, and S. S. Sengar, "Community detection in complex networks using stacked autoencoders and crow search algorithm," *J. Supercomput.*, vol. 79, no. 3, pp. 3329-3356, 2023.
[21] T. Ma, Q. Liu, J. Cao, Y. Tian, A. Al-Dhelaan, and M. Al-Rodhaan, "LGIEM: global and local node influence based community detection," *Futur. Gener. Comput. Syst.*, vol. 105, pp. 533-546, Apr. 2020.
[22] T. Wang, S. Chen, X. Wang, and J. Wang, "Label propagation algorithm based on node importance," *Phys. A Stat. Mech. its Appl.*, vol. 551, Article ID: 124137, Aug. 2020.
[23] C. Li, *et al.*, "NANI: an efficient community detection algorithm based on nested aggregation of node influence," 2017.
[24] D. Zhuang, M. Chang, and M. Li, "DynaMo: dynamic community detection by incrementally maximizing modularity," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 5, pp 1934-1945, May 2021.
[25] S. A. Seyedi, A. Lotfi, P. Moradi, and N. N. Qader, "Dynamic graph-based label propagation for density peaks clustering," *Expert Syst. Appl.*, vol. 115, pp. 314-328, Jan. 2019.
[26] F. Liu, J. Wu, C. Zhou, and J. Yang, "Evolutionary community detection in dynamic social networks," in *Proc. Int. Joint Conf. on Neural Networks*, 7 pp. , Budapest, Hungary, 14-19 Jul. 2019.
[27] X. Su, J. Cheng, H. Yang, M. Leng, W. Zhang, and X. Chen, "IncNSA: detecting communities incrementally from time-evolving networks based on node similarity," *Int. J. Mod. Phys. C*, vol. 31, no. 7, Article ID: 2050094, 2020.
[28] M. Cordeiro, R. P. Sarmento, and J. Gama, "Dynamic community detection in evolving networks using locality modularity optimization," *Soc. Netw. Anal. Min.*, vol. 6, Article ID: 15, 2016.
[29] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," in *Proc. of the 11th ACM SIGKDD Int. Conf. on Knowledge Discovery in Data Mining*, pp. 177-187, Chicago, IL, USA, 21-24 Aug. 2005.
[30] A. Paranjape, A. R. Benson, and J. Leskovec, "Motifs in temporal networks," in *Proc. of the 10th ACM Int. Conf. on Web Search and Data Mining*, pp. 601-610, Cambridge, UK, 6-10 Dec. 2017.
[31] P. Panzarasa, T. Opsahl, and K. M. Carley, "Patterns and dynamics of users' behavior and interaction: network analysis of an online community," *J. Am. Soc. Inf. Sci. Technol.*, vol. 60, no. 5, pp. 911-932, May 2009.

مصطفی سبزه‌کار تحصیلات خود را در رشته مهندسی کامپیوتر در مقطع کارشناسی در سال ۱۳۸۶ در دانشگاه خوارزمی و مقاطع کارشناسی ارشد و دکتری مهندسی کامپیوتر را در دانشگاه فردوسی مشهد و به ترتیب در سال‌های ۱۳۸۹ و ۱۳۹۶ در دانشگاه فردوسی مشهد به پایان رساند. او هم‌اکنون استادیار گروه مهندسی کامپیوتر دانشگاه صنعتی بیرجند می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: یادگیری ماشین، بیوانفورماتیک و شبکه‌های اجتماعی.

شیما برادران‌نژاد مدرک کارشناسی مهندسی کامپیوتر را از دانشگاه بیرجند در سال ۱۳۸۶ دریافت کرده است. همچنین، مدرک کارشناسی‌ارشد مهندسی کامپیوتر با گرایش نرم‌افزار را از دانشگاه آزاد اسلامی واحد بیرجند در سال ۱۴۰۱ دریافت نمود. زمینه‌های پژوهشی مورد علاقه وی شامل شناسایی الگو و الگوریتم‌های فراابتکاری است.

مهدی خزاعی‌پور مدرک کارشناسی مهندسی کامپیوتر را در سال ۱۳۸۲ از دانشگاه آزاد اسلامی واحد مشهد، مدرک کارشناسی‌ارشد مهندسی کامپیوتر را از دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران در سال ۱۳۸۷ دریافت کرده است. همچنین، وی دارای مدرک دکترای مهندسی کامپیوتر با گرایش توسعه سیستم‌های نرم‌افزاری از دانشگاه آزاد اسلامی واحد کرمان، کرمان، می‌باشد. وی در حال حاضر استادیار گروه کامپیوتر دانشگاه آزاد اسلامی واحد بیرجند است. زمینه‌های پژوهشی مورد علاقه وی شامل داده‌کاوی، برآورد تلاش نرم‌افزار، و الگوریتم‌های تکاملی می‌باشد.

مهدی خرد تحصیلات خود را در رشته مهندسی کامپیوتر در مقطع کارشناسی در سال ۱۳۸۹ و کارشناسی ارشد فناوری اطلاعات در سال ۱۳۹۴ در دانشگاه بیرجند گذراند. وی اکنون دانشجوی دکتری فناوری اطلاعات در دانشگاه قم است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: یادگیری ماشین، داده کاوی، الگوریتم‌های فراابتکاری و دینامیک محاسباتی.