



دانشگاه کاشان
University of Kashan

مجله محاسبات نرم

SOFT COMPUTING JOURNAL

تارنمای مجله: scj.kashanu.ac.ir



چارچوب پیش‌بینی پیوند با استفاده از شبکه عصبی گرافی مبتنی بر زیرگراف*

سید مهدی وحیدی پور^{1*}، استادیار، ریحانه کرمی¹، دانشجوی کارشناسی ارشد

¹ دانشکده مهندسی برق و کامپیوتر، دانشگاه کاشان، کاشان، ایران.

اطلاعات مقاله

چکیده

تاریخچه مقاله:

دریافت 31 مرداد ماه 1402

پذیرش 3 بهمن ماه 1402

کلمات کلیدی:

شبکه‌های پیچیده

شبکه عصبی گرافی

پیش‌بینی پیوند

رده‌بندی

زیرگراف

یادگیری بازنمایی گراف

پیش‌بینی پیوند یکی از موضوع‌های مهم در تجزیه و تحلیل شبکه‌های پیچیده است. پیش‌بینی پیوند می‌تواند توسط یک رده‌بند انجام شود؛ به طوری که بردار ویژگی یک جفت گره، ورودی آن باشد. خروجی رده‌بند نشان می‌دهد که آیا میان آن جفت گره پیوندی پیش‌بینی می‌شود یا خیر (رده یک یا رده صفر). برای استخراج بردار ویژگی یک جفت گره می‌توان از شبکه‌های عصبی گرافی (GNN) استفاده نمود که در این صورت روش حل مساله پیش‌بینی پیوند مبتنی بر شبکه عصبی گرافی خواهد بود. در این مقاله، یک روش حل مساله پیش‌بینی پیوند مبتنی بر شبکه عصبی گرافی به نام Graph Auto Encoder (GAE) به عنوان روش پایه در نظر گرفته شده است. یکی از مشکلات اساسی در این روش آن است که بردار ویژگی استخراج شده توسط شبکه عصبی گرافی به ازای جفت گره‌های متفاوت، می‌تواند یکسان باشد. برای رفع این مشکل، در این مقاله با استفاده از مفهوم زیرگراف روش پایه بهبود داده شده و چارچوب جدیدی با نام Sub-Graph Auto Encoder (SGAE) پیشنهاد شده است. چارچوب پیشنهادی بر اساس معیارهای مختلف ارزیابی و با روش پایه مقایسه شده است که نتایج نشان‌دهنده بهبود عملکرد آن است. به عنوان مثال روش SGAE به طور متوسط نسبت به روش پایه GAE در معیارهای دقت، F1-Score، متوسط صحت و مساحت زیر نمودار صحت-فراخوانی، بهبود 5/5، 5/5، 5/75 و 5/87 را ایجاد کرده است.

© 1402 نویسندگان. مقاله با دسترسی آزاد تحت مجوز CC-BY

1. مقدمه

موجودیت و پیوندها نشان‌دهنده ارتباط بین آنها است. یکی از موضوع‌های علمی مهم مرتبط با شبکه، پیش‌بینی پیوند در گراف است که می‌توان آن را به عنوان پیش‌بینی روابط در شبکه مطرح کرد. پیش‌بینی پیوند، به بررسی پیش‌بینی روابط جدید می‌پردازد [2]. هدف پیش‌بینی پیوند، یافتن احتمال شکل‌گیری پیوند در شبکه، براساس اتصال‌های مشاهده شده کنونی است [2]. پیش‌بینی پیوند اساسی‌ترین مساله‌ای است که تلاش می‌کند احتمال وجود پیوند بین دو گره براساس پیوندهای مشاهده شده و ویژگی‌های گراف تخمین زده می‌شود [3].

سیاری از سیستم‌های اجتماعی، بیولوژیکی و اطلاعاتی می‌توانند به خوبی توسط شبکه‌های پیچیده، به صورت موجودیت‌هایی که با یکدیگر در ارتباط هستند، توصیف شوند [1]. یکی از روش‌های توصیف گراف است؛ به صورتی که گره‌ها نشان‌دهنده

* نوع مقاله: پژوهشی

* نویسنده مسئول

پست(های) الکترونیک: vhaidipour@kashanu.ac.ir (وحیدی پور)

r.karami@grad.kashanu.ac.ir (کرمی)

شرح زیر است:

- ارائه چارچوب SGAE مبتنی بر ترکیب زیرگراف و روش GAE که مساله پیش‌بینی پیوند را به‌طور کاراتری حل می‌کند. برای ایجاد SGAE یک الگوریتم دو مرحله‌ای شامل استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف نیاز است که دو ایده متفاوت برای انجام آنها نیز پیشنهاد شده است.
- ایده‌ای برای استخراج زیرگراف که در SGAE پیشنهاد شده است با روش پرکاربرد h-hop [11] رقابت می‌کند.
- ایده‌ای برای ایجاد بردار ویژگی زیرگراف پیشنهاد شده است که در چارچوب پیشنهادی از روش شبکه عصبی کانولوشنال⁸ [12] کارا تر است.

ساختار مقاله در ادامه به شرح زیر است: بخش دوم به مرور ادبیات و کارهای مرتبط می‌پردازد. بخش سوم روشی را که چارچوب پیشنهادی، مبتنی بر آن ایجاد شده است را معرفی می‌کند. در بخش چهارم به معرفی روش‌های پیشنهادی پرداخته شده است. در بخش پنجم، آزمایش‌های انجام شده و نتایج آنها گفته می‌شود و در آخر، در بخش ششم، نتیجه‌گیری بیان می‌شود.

2. مرور ادبیات و کارهای مرتبط

در این بخش به مرور ادبیات و کارهای مرتبط پرداخته می‌شود. با توجه به این که در این مقاله از زیرگراف، برچسب‌گذاری، بردار ویژگی و شبکه عصبی گرافی برای حل مساله پیش‌بینی پیوند استفاده شده است، در ادامه این مفاهیم شرح داده می‌شوند.

1.2. زیرگراف⁹

گراف $G(V, E)$ با مجموعه گره V و مجموعه پیوند E را در نظر بگیرید. زیرگراف G_k زیرمجموعه‌ای از گره‌ها و پیوندهای گراف است که به صورت $S_G(\hat{V}, \hat{E})$ نشان داده می‌شود؛ به طوری که $\hat{V} \subseteq V$ و $\hat{E} \subseteq E$ باشد.

اخیراً از مفهوم تعبیه¹ برای حل مساله پیش‌بینی پیوند استفاده می‌شود [4]-[7]؛ تعبیه، یک بردار ویژگی است که تعداد درایه‌های آن (ابعاد) به نسبت اندازه گراف بسیار کم است. به ازای هر المان در گراف (مانند گره، پیوند، زیرگراف و حتی کل گراف) می‌توان تعبیه ایجاد کرد. به عبارت دیگر، یک المان در گراف به کمک تعبیه‌ها در یک فضای دیگری (به نام فضای تعبیه) بازنمایی² می‌شود. روش‌های یادگیری بازنمایی گراف³ به روش‌های تولید بردارهای تعبیه گفته می‌شود که توسط مدل‌های یادگیری ایجاد شده باشند [1].

یکی از روش‌های یادگیری بازنمایی گراف، شبکه‌های عصبی گرافی⁴ (GNN) است [8]، که در آن برای هر گره یک تعبیه ایجاد می‌شود. از GNN می‌تواند در حل مساله پیش‌بینی پیوند استفاده می‌شود که یکی از معروف‌ترین آن‌ها روش GAE⁵ است [9]. در روش GAE برای پیش‌بینی پیوند میان دو گره هدف، ابتدا تعبیه‌ی آن جفت گره هدف به دست می‌آید و سپس با استفاده از ترکیب دو تعبیه یک بازنمایی (بردار ویژگی) ایجاد می‌شود؛ براساس این بازنمایی احتمال وجود پیوند تخمین زده می‌شود. یکی از ضعف‌های روش GAE این است که در این روش تنها بازنمایی گره‌ها یادگرفته می‌شود و بازنمایی پیوند از روی آنها ساخته می‌شود. به همین دلیل اگر جفت گره‌های دو سر دو پیوند غیرهم‌ریخت، به صورت دو به دو هم‌ریخت⁶ باشند، بازنمایی یکسانی برای پیوندها ایجاد می‌شود [10].

برای رفع این ضعف روش GAE، در این مقاله پیشنهاد می‌شود تا مفهوم و ویژگی زیرگراف به آن اضافه شود. در همین راستا، چارچوب پیشنهادی SGAE⁷ پیشنهاد می‌شود. زیرگراف‌ها زیرمجموعه‌ای از گراف را نشان می‌دهد که شامل تعدادی گره و روابط بین آنها است. با توجه به اینکه، برای حل مساله پیش‌بینی پیوند از زیرگراف استفاده شده است، نوآوری‌های این مقاله به

¹ Embedding

² Representation

³ Graph Representation Learning

⁴ Graph Neural Network

⁵ Graph Auto Encoder

⁶ Isomorph

⁷ Sub-Graph Auto Encoder

⁸ Convolutional Neural Network

⁹ Subgraph

گره‌های هدف از گره‌های دیگر و حفظ ترتیب گره‌ها در صورت تکرار را داشته باشند [10].

3.2. بردار ویژگی³

بردار ویژگی، برداری تک بعدی و یا چند بعدی است؛ که می‌تواند شامل ویژگی‌های المان‌های گراف باشد؛ که به آن بردار بازنمایی نیز گفته می‌شود. در این مقاله بردار ویژگی گره (A) ، گراف (G) و زیرگراف (S_G) ، به ترتیب با $h(A)$ ، $h(G)$ و $h(S_G)$ نشان داده می‌شود. روش‌های متفاوتی برای ایجاد بردار ویژگی معرفی شده است که در این جا به دو دسته سنتی و ایجاد تعبیه دسته‌بندی شده است.

- روش‌های سنتی: روش‌های سنتی ایجاد بردار ویژگی، روش‌هایی هستند که در آنها به صورت دستی اعداد درون بردار ویژگی مشخص می‌شوند و می‌توان دلیل و روش ایجاد این اعداد را توضیح داد. برای مثال ماتریس همسایگی یک بردار ویژگی دو بعدی ساده است که هر درایه درون آن نشان‌دهنده وجود و یا عدم وجود پیوند است.
- روش‌های ایجاد تعبیه: تعبیه نوعی بردار ویژگی با ابعاد پایین است که به وسیله مدل‌های یادگیری ایجاد می‌شود و مقادیر آن قابل تحلیل نیست. یکی از روش‌های ایجاد تعبیه استفاده از شبکه عصبی گرافی می‌باشد.

4.2. شبکه عصبی گرافی

شبکه عصبی گرافی یا GNN یک چارچوب یادگیری عمیق در گراف است که در سال‌های اخیر بسیار مورد توجه قرار گرفته است. شکل‌های اولیه GNN در مراجع [15] - [17] آمده است. GNN برای هر گره، به وسیله پیام‌ها و اطلاعات ارسال شده توسط گره‌های همسایه آن گره، تعبیه ایجاد می‌کند. GNN در هر لایه از سه بخش اساسی تشکیل شده است؛ (1) محاسبه پیام⁴ (2) جمع‌کننده⁵ و (3) به‌روزرسانی¹؛ که در بخش محاسبه پیام،

استخراج یک زیرگراف به تعیین مجموعه‌های V و E گفته می‌شود به گونه‌ای که شامل المان یا المان‌های مشخصی باشد. به طور مثال برای پیش‌بینی پیوند میان دو گره مشخص (هدف)، زیرگراف استخراج شده از گراف اصلی باید شامل جفت گره هدف باشد. در ادامه دو دسته از روش‌های استخراج زیرگراف مرور می‌شوند:

- مبتنی بر همسایگی: روش‌های استخراج زیرگراف مبتنی بر همسایگی، گره‌های همسایه‌ی سطح‌های متفاوت گره یا پیوند هدف را به عنوان گره‌های زیرگراف استخراج می‌کنند؛ به طوری که اگر A گره هدف باشد و $N_h(A)$ بیانگر همسایه‌های گره A در سطح h باشد، $V \subseteq N_h(A)$ است و پیوندهای بین گره‌های زیرگراف در مجموعه E قرار می‌گیرد. یکی از پرکاربردترین روش‌های ایجاد زیرگراف مبتنی بر همسایگی، روش h-hop [11] است که در بخش 4.1.2 به معرفی آن پرداخته شده است.
- مبتنی بر جامعه¹: جامعه یک نوع زیرگراف است؛ که به طور گسترده‌ای در شبکه‌ها وجود دارد و به عنوان یکی از ضروری‌ترین و مفیدترین استراتژی‌های شبکه شناخته شده است. جامعه مجموعه‌ای از گره‌هایی است که احتمالاً ویژگی‌های مشترکی دارند و یا نقش‌های مشابهی در شبکه ایفا می‌کنند. پیوندهای بین گره‌ها در یک جامعه یکسان مترکم و در بین جوامع متفاوت پراکنده است [13]، [14].

2.2. برچسب‌گذاری²

برچسب‌گذاری به معنی نسبت دادن برچسب به گره‌ها یا پیوندهای گراف است. برچسب‌ها می‌توانند نشان‌دهنده اولویت، ترتیب، اهمیت و ... گره‌ها یا پیوندها باشند. هدف اصلی برچسب‌گذاری، نشان دادن نقش‌های متفاوت گره‌ها یا پیوندهای مختلف در گراف است [4]. روش‌های برچسب‌گذاری گره استفاده شده در پیش‌بینی پیوند باید توانایی متمایز کردن

³ Feature Vector

⁴ Message Computation

⁵ Aggregation

¹ Community

² Labelling

5.2. کارهای مرتبط

پیش‌بینی پیوند یک مساله کلیدی در تحلیل شبکه است. در پیش‌بینی پیوند تلاش می‌شود که احتمال ایجاد پیوند بین دو گره در آینده، براساس پیوندهای مشاهده شده و ویژگی‌های شبکه تخمین زده شود. در بسیاری از کارها برای بهبود مساله پیش‌بینی پیوند از بردار ویژگی‌های زیرگراف استفاده شده است. روش‌های ایجاد بردار ویژگی زیرگراف را همانطور که در بخش 2.3 گفته شد می‌توان به دو دسته (1) روش‌های سنتی و (2) روش‌های ایجاد تعبیه تقسیم‌بندی کرد. در دسته اول بردار ویژگی به صورت سنتی و بر اساس ویژگی‌های قابل مشاهده زیرگراف ایجاد می‌شود در صورتی که در دسته دوم بردار ویژگی به وسیله شبکه‌های عصبی و یا روش‌های قدم‌زدن تصادفی ایجاد می‌شوند. در روش ایجاد تعبیه شبکه‌های عصبی، اکثراً از روش‌های سنتی برای ایجاد بردار ویژگی اولیه برای ورودی شبکه عصبی استفاده می‌شود.

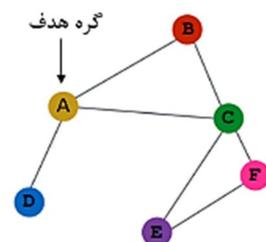
در این بخش به بررسی کارهایی پرداخته شده که: (1) از رده‌بند² برای حل مساله استفاده کرده‌اند و (2) در حل مساله پیش‌بینی پیوند از ویژگی‌ها و اطلاعات زیرگراف برای ایجاد بردار ورودی رده‌بند استفاده کرده‌اند.

در پژوهش‌های [18] - [20] از روش‌های سنتی برای ایجاد بردار ویژگی زیرگراف استفاده شده است. در این پژوهش‌ها ابتدا برای هر جفت گره هدف به وسیله روش h-hop زیرگراف استخراج شده و سپس گره‌های زیرگراف به وسیله روش‌های برچسب‌گذاری، برچسب‌گذاری می‌شوند. در این مرحله برای هر زیرگراف برچسب‌گذاری شده به صورت سنتی بردار ویژگی ایجاد می‌شود. در مرجع [18] از وجود و یا عدم وجود پیوند بین هر جفت گره زیرگراف و در مراجع [19] و [20] از 5 معیار شباهت و معیار وزن برای شبکه‌های وزن‌دار برای ایجاد بردار ویژگی زیرگراف استفاده می‌شود. در آخر برای پیش‌بینی پیوند جفت گره هدف، بردار ویژگی زیرگراف به رده‌بند داده می‌شود. در پژوهش‌های [4] - [7] از روش‌های ایجاد تعبیه برای ایجاد

بردار ویژگی گره‌ها به پیام تبدیل می‌شود. در بخش جمع‌کننده، پیام همسایه‌های گره با یکدیگر جمع و تبدیل به یک بردار ویژگی می‌شود و در بخش به‌روزرسانی، به منظور از بین نرفتن اطلاعات گره مورد نظر، بردار ویژگی تولید شده در بخش دوم با پیام آن گره به وسیله توابعی مانند جمع و الصاق ترکیب شده و در نهایت بردار ویژگی جدید ایجاد می‌شود.

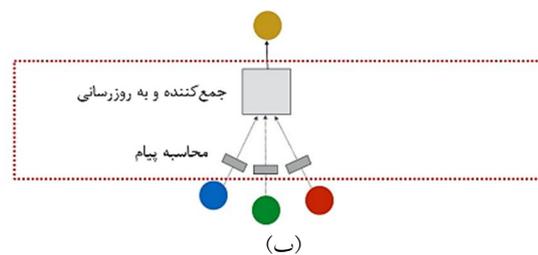
در شکل (1) ساختار کلی یک شبکه عصبی گرافی تک لایه نشان داده شده است. در قسمت (الف) شکل، گراف ورودی نشان داده شده که گره A در آن گره هدفی است که تعبیه آن در قسمت (ب) با توجه به همسایه‌های آن محاسبه می‌شود. همان‌طور که نشان داده شده است، مجموعه همسایه‌های گره A شامل گره‌های B، C و D است؛ که در قسمت (ب)، بردارهای ویژگی آنها ابتدا در قسمت محاسبه پیام، به پیام تبدیل شده، سپس در قسمت جمع‌کننده و به‌روزرسانی، پیام همسایه‌ها جمع شده و بردار ویژگی جدیدی را ایجاد می‌کند، سپس بردار ویژگی ایجاد شده با بردار ویژگی گره A ترکیب شده و در نهایت تعبیه جدید گره A به دست می‌آید.

گراف ورودی



(الف)

شبکه عصبی گرافی



(ب)

شکل (1): تصویر ساختار کلی شبکه عصبی گرافی تک لایه.

² Classifier

¹ Update

سمت چپ شکل، بخش رمزنگار قرار دارد که در آن ابتدا گراف ورودی با جفت گره‌های هدف (A, B) و (C, D) که به ترتیب در رده‌های 1 و 0 قرار دارند، به شبکه عصبی گرافی داده می‌شوند. شبکه عصبی گراف، برای هر گره گراف یک تعبیه ایجاد می‌کند. تعبیه‌های ایجاد شده در بخش رمزنگار به بخش رمز گشا داده می‌شود. همان‌طور که در ابتدای بخش رمزگشا نشان داده شده است، تعبیه جفت گره‌های هدف (A, B) و (C, D) به یکدیگر الصاق شده و برای پیش‌بینی پیوند به رده‌بند شبکه عصبی داده می‌شود.

4. چارچوب پیشنهادی SGAE

چارچوب پیشنهادی SGAE بر اساس روش پایه GAE است که در حل مساله پیش‌بینی پیوند ویژگی زیرگراف شامل دو جفت گره هدف را نیز در نظر می‌گیرد. این چارچوب مانند روش پایه، از دو قسمت رمزنگار و رمزگشا تشکیل شده است که در قسمت رمزگشای آن به منظور استفاده از زیرگراف، یک الگوریتم دو مرحله‌ای اضافه شده است (شکل 3). در قسمت رمزنگار چارچوب SGAE، همانند روش پایه، گراف ورودی دریافت و برای تمام گره‌های آن به وسیله شبکه عصبی گرافی تعبیه ایجاد می‌شود. از تعبیه‌های استخراج شده، در قسمت رمزگشا استفاده می‌شود. تغییرهای چارچوب پیشنهادی نسبت به روش پایه در قسمت رمزگشا است. در قسمت رمزگشا، یک الگوریتم دو مرحله‌ای برای تزریق زیرگراف به روش پایه اضافه شده است. در مرحله اول این الگوریتم، برای هر جفت گره هدف، یک زیرگراف استخراج می‌شود. سپس در مرحله دوم، برای هر زیرگراف یک بردار ویژگی زیرگراف ایجاد می‌شود. در انتها و در پیش‌بینی‌کننده، تعبیه گره‌های هدف و بردار ویژگی زیرگراف به صورتی به یکدیگر الصاق می‌شوند که بردار ویژگی زیرگراف در انتها قرار گیرد. بردار ویژگی ایجاد شده را بردار ویژگی پیوند هدف مبتنی بر زیرگراف می‌نامیم که برای پیوند هدف (A, B) با بردار ویژگی پیوند مبتنی بر زیرگراف به عنوان ورودی به رده‌بند شبکه عصبی داده می‌شود.

بردار ویژگی زیرگراف استفاده شده است. در مراجع [4] و [5] ابتدا برای هر جفت گره هدف با روش h-hop یک زیرگراف استخراج و برچسب‌گذاری می‌شود. در پژوهش اول به وسیله روش‌های قدم‌زدن تصادفی و در پژوهش دوم به وسیله شبکه‌های عصبی گرافی، تعبیه گره‌های زیرگراف ایجاد می‌شود. این تعبیه‌ها به منظور پیش‌بینی پیوند به رده‌بند شبکه‌های عصبی داده می‌شود. در پژوهش [7] ابتدا برای هر گره هدف، به وسیله روش h-hop، زیرگراف‌هایی با عمق‌های متفاوت استخراج و سپس گراف‌لت‌های¹ زیرگراف‌های ایجاد شده برای هر گره شمرده و بردار ویژگی آن ایجاد می‌شود. این بردار ویژگی به شبکه رمزنگار خودکار گراف داده برای هر گره یک تعبیه ایجاد می‌کند. تعبیه جفت گره‌های هدف به یکدیگر متصل شده و برای پیش‌بینی پیوند به شبکه عصبی داده می‌شود. در پژوهش [6] یک روش برای ایجاد تعبیه گره‌های شبکه مبتنی بر قدم‌زدن تصادفی در زیرگراف جامعه پیشنهاد و برای حل مساله پیش‌بینی پیوند استفاده شده است.

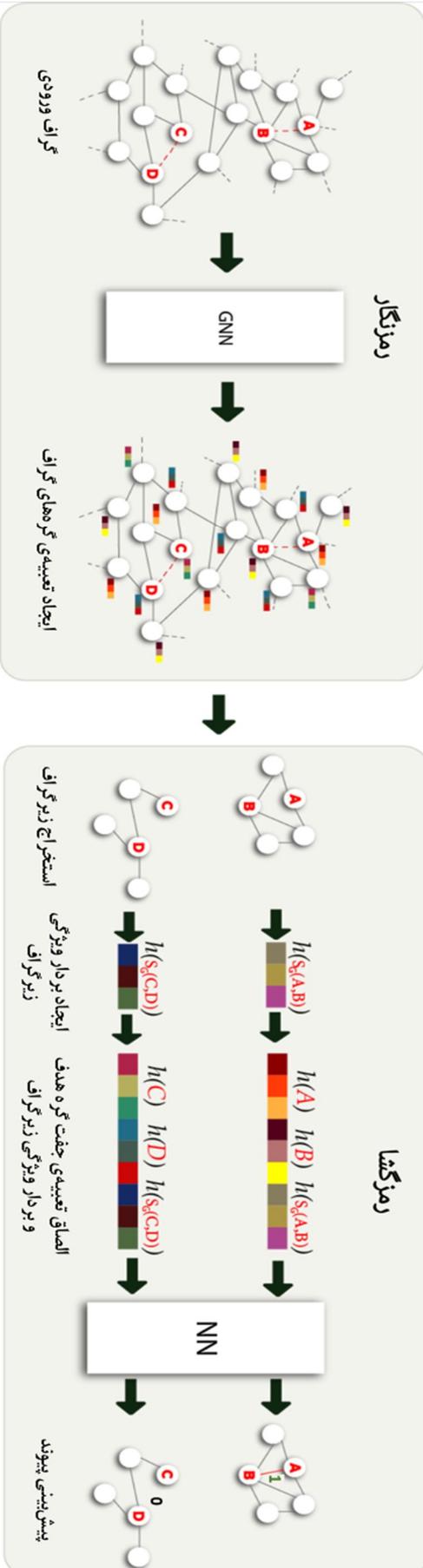
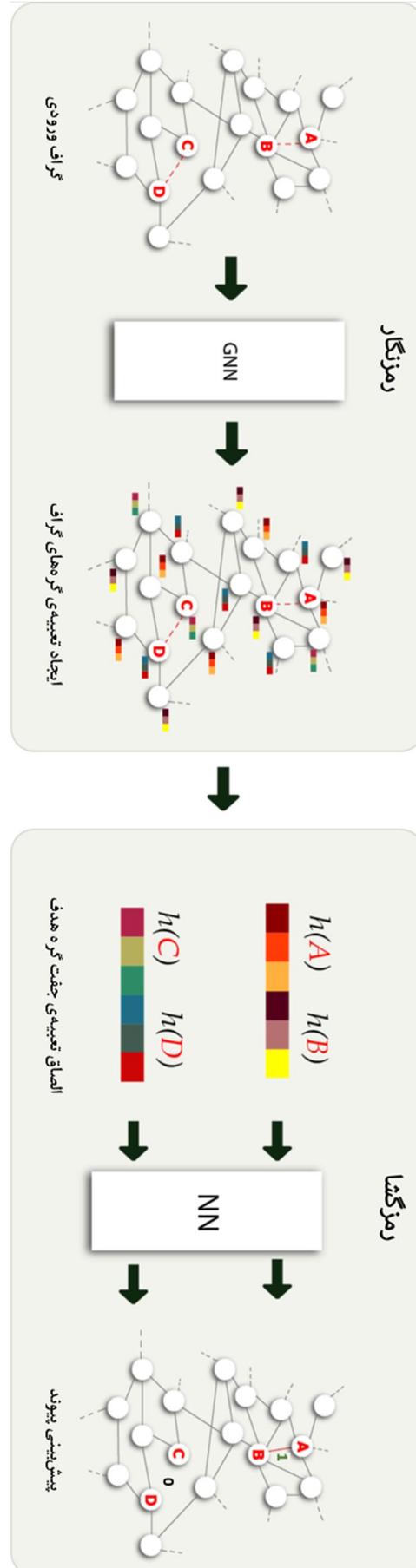
3. روش GAE

یکی از روش‌هایی که از تعبیه برای پیش‌بینی پیوند استفاده می‌کند، روش GAE [9] است. این روش به دو بخش رمزنگار و رمزگشا تقسیم می‌شود. در بخش رمزنگار تعبیه‌های گره‌های گراف ایجاد و در بخش رمزگشا پیش‌بینی پیوند انجام می‌شود.

- رمزنگار: در بخش رمزنگار، به وسیله شبکه عصبی گرافی تعبیه گره‌های گراف ایجاد می‌شود. شبکه عصبی گرافی ماتریس همسایگی و ماتریس ویژگی گراف را به عنوان ورودی گرفته و تعبیه گره‌های گراف را ایجاد می‌کند.
- رمزگشا: در بخش رمزگشا، تعبیه جفت گره هدف به یکدیگر الصاق شده و بردار ویژگی پیوند هدف را ایجاد می‌کند. بردار ویژگی پیوند هدف به منظور پیش‌بینی وجود و یا عدم وجود پیوند به رده‌بند شبکه عصبی ساده داده شده و پیش‌بینی پیوند انجام می‌شود.

در شکل (2) مثالی از روش GAE نشان داده شده است. در

¹ Graphlet



بردار ویژگی هستند¹. در استخراج زیرگراف مبتنی بر جفت گره هدف، ابتدا گره‌ای فرضی به نام $DummyTarget$ فرض می‌شود و برای این گره فرضی، با استفاده از رابطه (1) بردار ویژگی محاسبه می‌شود ($h(DummyTarget)$ این بردار را نشان می‌دهد). چنانچه زیرگراف تنها مبتنی بر یک گره، استخراج شود آنگاه بدیهی است که تک گره هدف همان گره $DummyTarget$ است و بردار ویژگی گره جدید همان بردار ویژگی گره هدف است.

$$h(DummyTarget) = \frac{h(A) + h(B)}{2} \quad (1)$$

هر گره مانند z فاصله‌ای با گره $DummyTarget$ دارد که با $d(z)$ نشان داده می‌شود. بعد از به دست آوردن فاصله گره‌های گراف با گره $DummyTarget$ ، گره‌های گراف براساس فاصله از کوچک به بزرگ مرتب شده و k نزدیک‌ترین گره به گره $DummyTarget$ استخراج شده و زیرگراف را می‌سازند. تعداد گره‌های زیرگراف، طبق رابطه (2) که در مقاله [19] معرفی شده است، به دست می‌آید.

$$k = \left\lceil \frac{2|E|}{|V|} \cdot \left(1 + \frac{2|E|}{|V| \cdot (|V| - 1)}\right) \right\rceil \quad (2)$$

در شکل (4)، شبه‌کد روش استخراج زیرگراف DIS برای جفت گره هدف نشان داده شده است. در این شبه‌کد، ابتدا ورودی‌ها و خروجی‌های تابع نشان داده شده است. ورودی‌های تابع شامل گراف ورودی، جفت گره‌های هدف که با A, B نشان داده شده است، بردار ویژگی گره‌های گراف و تعداد گره‌های زیرگراف (k) و خروجی آن زیرگراف $S_G(A, B)$ است. در خط اول شبه‌کد، بردار ویژگی $DummyTarget$ همانطور که نشان داده شده است (رابطه (1))، به صورت میانگین بردار ویژگی گره‌های هدف محاسبه می‌شود. در خط سوم، فاصله بین گره‌های گراف و گره $DummyTarget$ برای همه گره‌های گراف به جز جفت گره هدف محاسبه و در خط چهارم فاصله

شکل (3) چارچوب پیشنهادی SGAE را نشان می‌دهد. گراف G به عنوان ورودی به چارچوب داده می‌شود. همانطور که در این شکل نشان داده شده است، در بخش رمزنگار بوسیله شبکه عصبی گرافی برای تمام گره‌های گراف G ، تعبیه ایجاد می‌شود (به عنوان مثال برای گره‌های A و B ، به ترتیب تعبیه‌های $h(A)$ و $h(B)$ تولید می‌شود). در بخش رمزگشا برای هر جفت گره هدف (A, B) و (C, D) زیرگراف‌های $S_G(A, B)$ و $S_G(C, D)$ ایجاد می‌شود. در ادامه با استفاده از روش ایجاد بردار ویژگی زیرگراف، بردار ویژگی زیرگراف‌های $h(S_G(A, B))$ و $h(S_G(C, D))$ ایجاد می‌شود. بعد از ایجاد بردار ویژگی زیرگراف، تعبیه گره‌های هدف (به عنوان مثال $h(A)$ و $h(B)$) و بردار ویژگی زیرگراف (به عنوان مثال $h(S_G(A, B))$) به یکدیگر الصاق شده و بردار ویژگی پیوند هدف مبتنی بر زیرگراف (به عنوان مثال $h(A, B)_{S_G}$) را ایجاد می‌کند؛ که برای پیش‌بینی پیوند به ورودی رده‌بند شبکه عصبی داده می‌شود. در نهایت رده‌ی هر جفت گره پیش‌بینی می‌شود؛ به طوری که اگر بین جفت گره هدف پیوندی پیش‌بینی شود، در رده 1 و در غیر این صورت در رده 0 رده‌بندی می‌شود.

همان‌طور که گفته شد، در چارچوب پیشنهادی، یک الگوریتم دو مرحله‌ای وجود دارد که شامل استخراج زیرگراف و ساخت بردار ویژگی زیرگراف است. در ادامه، روش‌های پیشنهادی برای این دو مرحله ارائه می‌شود. در انتهای این قسمت نیز یک تحلیل برای SGAE ارائه خواهد شد.

1.4. روش پیشنهادی استخراج زیرگراف

در این بخش دو روش برای استخراج زیرگراف معرفی می‌شود؛ که یکی از آنها روش پیشنهادی DIS و دیگری روش h -hop است.

1.1.4. روش DIS

در روش پیشنهادی DIS، فرض می‌شود که گره‌های گراف دارای

¹ این بردار ویژگی می‌تواند بردار ویژگی ذاتی گره‌ها، بردار ویژگی ساخته شده برای آنها، بردار امبدینگ و یا هر ترکیبی از بردارهای مختلف باشد.

Algorithm2: Extract Subgraph h-hop

Input: $G(V, E)$: input graph,
 $\{A, B\}$: target nodes pair,
hops: the size of hops,
k: the size of subgraph.

Output: $S_G(A, B)$: subgraph.

- 1: for *h* in *hops*:
- 2: $subgraph\ nodes = subgraph\ nodes \cup \mathcal{N}_h(A) \cup \mathcal{N}_h(B)$
- 3: $S_G(A, B) \leftarrow subgraph\ nodes[:k] \cup \{A, B\}$
- 4: return ($S_G(A, B)$)

شکل (5): شبه‌کد روش استخراج زیرگراف H-HOP.

2.4. روش پیشنهادی ایجاد بردار ویژگی زیرگراف

در این بخش دو روش برای ایجاد بردار ویژگی زیرگراف پیشنهاد می‌شود. در این روش‌ها مانند روش استخراج زیرگراف DIS فرض می‌شود که گره‌های گراف دارای بردار ویژگی هستند. روش‌های پیشنهادی ایجاد بردار ویژگی زیرگراف بر روی زیرگراف‌هایی که دارای تک و یا جفت گره هدف می‌باشند قابل استفاده می‌باشند. در این روش‌ها ابتدا گره *DummyTarget* و بردار ویژگی آن همان‌طور که در قسمت قبل توضیح داده شد، مشخص می‌شود؛ سپس با استفاده از روش‌های پیشنهادی بردار ویژگی زیرگراف ایجاد می‌شود. این روش‌ها در ادامه توضیح داده شده است.

1.2.4. روش NDP

در این روش گره‌ها بر اساس فاصله بردارهای ویژگی آنها با گره‌های هدف جریمه می‌شوند؛ به همین دلیل این روش جریمه فاصله نرمال یا NDP² نام‌گذاری شده است. در این روش به هر گره زیرگراف، بر اساس فاصله بردار ویژگی آنها با بردار ویژگی گره *DummyTarget*، یک وزن نسبت داده می‌شود؛ که آن را برای گره *Z* به صورت $w(Z)$ نشان می‌دهیم. روش محاسبه وزن

گره‌ها به صورت صعودی مرتب می‌شود. در خط پنجم، *k* تا از گره‌های نزدیک‌تر به گره *DummyTarget* انتخاب و با جفت گره‌های هدف اجتماع شده و زیرگراف S_G استخراج می‌شود.

Algorithm1: Extract Subgraph DIS

Input: $G(V, E)$: input graph,
 $\{A, B\}$: target nodes pair,
h: feature vector of nodes in graph $\{h(z) \mid z \in V\}$,
k: the size of subgraph.

Output: $S_G(A, B)$: subgraph.

- 1: $h(DummyTarget) \leftarrow (h(A) + h(B))/2$
- 2: for each *z* in $V \setminus \{A, B\}$:
- 3: $d(z) \leftarrow Distance(h(z), h(DummyTarget))$
- 4: $sort\ nodes \leftarrow Sort\ graph\ nodes\ by\ d$
- 5: $S_G(A, B) \leftarrow sort\ nodes[:k] \cup \{A, B\}$
- 6: return ($S_G(A, B)$)

شکل (4): شبه‌کد روش پیشنهادی استخراج زیرگراف DIS.

2.1.4. روش h-hop

روش h-hop، یک روش استخراج زیرگراف مبتنی بر همسایگی است. در این روش برای هر گره هدف، گره‌های همسایه تا سطح *h* استخراج می‌شود. گره‌های استخراج شده (یا *k* تا از گره‌ها)، گره‌های زیرگراف (\hat{V}) و پیوندهای بین آنها پیوندهای زیرگراف (\hat{E}) را می‌سازند. چنانچه زیرگراف مبتنی بر دو گره هدف باشد؛ از اجتماع گره‌های همسایه آنها استفاده می‌شود. در شکل (5)، شبه‌کد روش استخراج زیرگراف h-hop برای جفت گره هدف نشان داده شده است. در این شبه‌کد، ابتدا ورودی‌ها و خروجی‌های تابع نشان داده شده است. ورودی‌های تابع شامل گراف ورودی، جفت گره‌های هدف برای مثال A, B ، تعداد سطح همسایگی (hops) و تعداد گره‌های زیرگراف (*k*) و خروجی آن زیرگراف $S_G(A, B)$ است. در خط دوم شبه‌کد، اجتماع گره‌های همسایه گره‌های هدف تا سطح *h* استخراج می‌شود. در خط سوم، *k* تا از آنها انتخاب و با جفت گره‌های هدف اجتماع شده و زیرگراف S_G استخراج می‌شود.

² Normal Distance Penalty

d که بعد بردار ویژگی هر گره را نشان می‌دهد، ضرب شده و بردار ویژگی زیرگراف را با ابعاد $d * 1$ را محاسبه می‌کند و در خط آخر این بردار ویژگی زیرگراف را برمی‌گرداند.

Algorithm3: Create a Subgraph Feature Vector NDP

Input: $S_G(A, B)(\hat{V}, \hat{E})$: input subgraph,
 $\{A, B\}$: target nodes pair in subgraph,
 h : feature vector of nodes in subgraph $\{h(z) | z \in \hat{V}\}$.

Output: $h(S_G(A, B))$: subgraph feature vector.

- 1: $h(\text{DummyTarget}) \leftarrow (h(A) + h(B))/2$
- 2: **for each** z **in** \hat{V} :
- 3: $d(z) \leftarrow \text{Distance}(h(z), h(\text{DummyTarget}))$
- 4: $\text{sorted nodes} \leftarrow \text{Sort subgraph nodes by } d$
- 5: **for each** z **in** sorted nodes :
- 6: $\hat{w} \leftarrow \log((1 - d(z))/d(z))$
- 7: **for each** z **in** sorted nodes :
- 8: $w \leftarrow \frac{\hat{w}(z)}{\sum_{i=0}^{k+2} \hat{w}(i)}$
- 9: $h(S_G(A, B)) \leftarrow w \cdot h(\text{sorted nodes})$
- 10: **return** $(h(S_G(A, B)))$

شکل (6): شبه‌کد روش پیشنهادی اول ایجاد بردار ویژگی زیرگراف.

2.2.4. روش CNN

در روش دوم، از شبکه عصبی کانولوشنال (CNN) برای ایجاد بردار ویژگی زیرگراف استفاده می‌شود؛ به‌همین دلیل این روش، روش ایجاد بردار ویژگی زیرگراف مبتنی بر شبکه کانولوشنال (و به اختصار روش CNN) نامیده شده است. در این روش ابتدا گره‌های زیرگراف بر اساس فاصله بردار ویژگی گره‌های زیرگراف با بردار ویژگی گره DummyTarget برچسب‌گذاری می‌شوند؛ این کار برای مشخص شدن ترتیب گره‌های زیرگراف در ماتریس ویژگی زیرگراف که ورودی شبکه عصبی کانولوشنال است بسیار اهمیت دارد. در این روش جفت گره همیشه برچسب‌های 1 و 2 می‌گیرد و گره‌های دیگر هرچقدر دورتر باشند، برچسب بزرگتری می‌گیرند. بعد از برچسب‌گذاری، بردار

گره‌ها از رابطه خطا AdBoost [21] الهام گرفته شده است. در این روش ابتدا به وسیله رابطه (3)، وزن اولیه هر گره به دست می‌آید؛ به طوری که $\hat{w}(z)$ وزن اولیه گره z را نشان می‌دهد.

$$\hat{w}(z) = \log((1 - d(z))/d(z)) \quad (3)$$

بعد از محاسبه وزن اولیه گره‌ها، با استفاده از رابطه (4) مقادیر وزن‌ها نرمال می‌شوند.

$$w(z) = \frac{\hat{w}(z)}{\sum_{i=0}^{k+2} \hat{w}(i)} \quad (4)$$

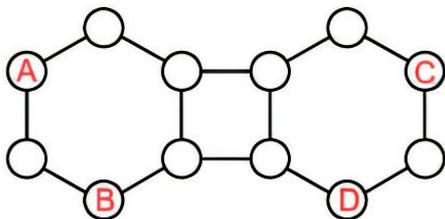
بعد از تعیین وزن گره‌های زیرگراف، بردار ویژگی هر گره i در وزن $w(i)$ ضرب شده و در نهایت بردارهای به دست آمده با یکدیگر جمع می‌شوند (رابطه (5)). بردار ویژگی $h(S_G)$ به دست آمده، بردار ویژگی زیرگراف است. باید توجه داشته باشیم که k بیانگر تعداد گره‌های زیرگراف بدون در نظر گرفتن گره‌های دوسر پیوند هدف است؛ به همین دلیل تعداد کل گره‌های زیرگراف برابر با $k + 2$ می‌باشد.

$$h(S_G) = \sum_{i=1}^{k+2} w(i) * h(i) \quad (5)$$

در شکل (6) شبه‌کد روش پیشنهادی ایجاد بردار ویژگی زیرگراف جریمه فاصله نرمال (NDP) برای جفت گره نشان داده شده است. در ابتدا ورودی‌ها و خروجی آن را مشخص می‌کند؛ که ورودی‌های آن شامل زیرگراف، جفت گره‌های هدف زیرگراف و بردار ویژگی گره‌های زیرگراف است و خروجی آن بردار ویژگی زیرگراف است که با $h(S_G(A, B))$ نشان داده شده است. در خط اول این شبه‌کد، بردار ویژگی گره DummyTarget محاسبه می‌شود و سپس در خط سوم، فاصله بردار ویژگی تمام گره‌های زیرگراف با گره DummyTarget به دست می‌آید. در خط چهارم، گره‌های زیرگراف براساس فاصله با گره DummyTarget به صورت صعودی مرتب می‌شوند. در خط ششم، برای هر گره زیرگراف به وسیله رابطه (3) یک وزن اولیه ایجاد شده و در خط هشتم وزن‌های اولیه با استفاده از رابطه (4) نرمال می‌شوند. در خط نهم بردار وزن با ابعاد $n * 1$ و ماتریس ویژگی مرتب شده زیرگراف با ابعاد $n * n$

3.4. بحث روی روش پیشنهادی

دلیل استفاده از زیرگراف در SGAE، به ضعف روش پایه GAE بازمی‌گردد؛ اگر گره‌های دو سر دو پیوند غیرهمریخت، به صورت دو به دو همریخت باشند، بردار ویژگی پیوندها یکسان خواهد شد. در روش GAE تعبیه گره‌ها یادگرفته می‌شود ولی بردار ویژگی پیوند یاد گرفته نمی‌شود. برای مثال در گراف نشان داده شده در شکل (8) گره‌های A و C و گره‌های B و D با یکدیگر همریخت‌اند به همین دلیل تعبیه این گره‌ها یکسان به دست می‌آید. بردار ویژگی پیوند (A, B) از ترکیب تعبیه‌های گره‌های A و B بردار ویژگی پیوند (C, B) از ترکیب تعبیه‌های گره‌های B و C به دست می‌آید. در نتیجه به دلیل یکسان بودن تعبیه‌های A و C بردار ویژگی پیوندهای (A, B) و (C, B) نیز یکسان خواهد بود؛ در حالی که همان‌طور که در شکل نشان داده شده است، این دو پیوند همریخت نیستند و نباید بردار ویژگی یکسانی داشته باشند. این یکی از ضعف‌های این روش است [10].



شکل (8): بازنمایی یکسان برای جفت گره‌های (A, B) و (C, B) به دست می‌آید که ضعف روش است [10].

در چارچوب پیشنهادی SGAE برای مرتفع ساختن این ضعف از بردار ویژگی زیرگراف استفاده می‌شود. در این چارچوب همان‌طور که در شکل (3) در قسمت رمزگشا نشان داده شده است، برای هر جفت گره هدف (پیوند هدف) یک بردار ویژگی زیرگراف ایجاد می‌شود؛ که به انتهای بردار ویژگی جفت گره هدف (پیوند هدف) الصاق می‌شود. از آنجایی که زیرگراف برای پیوندهای غیرهمریخت متفاوت است، بردار ویژگی آن نیز متفاوت خواهد بود؛ به همین دلیل بردار ویژگی پیوند مبتنی بر زیرگراف ایجاد شده برای پیوندهای غیرهمریخت متفاوت خواهد شد.

ویژگی گره‌ها با توجه به برجسب‌های مشخص شده، مرتب می‌شوند و ماتریس ویژگی زیرگراف ایجاد می‌شود. این ماتریس به‌عنوان ورودی به شبکه عصبی کانولوشنال داده می‌شود و خروجی آن یک بردار ویژگی است که بردار ویژگی زیرگراف نامیده می‌شود.

در شکل (7) شبه‌کد روش ایجاد بردار ویژگی زیرگراف مبتنی بر شبکه کانولوشنال برای جفت گره هدف A, B نشان داده شده است. در ابتدا ورودی‌ها و خروجی آن را مشخص می‌کنند؛ که ورودی‌های آن شامل زیرگراف، جفت گره‌های هدف زیرگراف و بردار ویژگی گره‌های زیرگراف است و خروجی آن بردار ویژگی زیرگراف است که با $h(S_G(A, B))$ نشان داده شده است. در خط اول این شبه‌کد، بردار ویژگی گره *DummyTarget* محاسبه می‌شود و سپس در خط سوم، فاصله بردار ویژگی تمام گره‌های زیرگراف با گره *DummyTarget* به دست می‌آید. در خط چهارم، گره‌های زیرگراف بر اساس فاصله به صورت صعودی مرتب می‌شوند. در خط پنجم ماتریس ویژگی مرتب شده زیرگراف به شبکه عصبی کانولوشنال داده می‌شود، خروجی این شبکه یک بردار ویژگی زیرگراف با ابعاد $1 * d$ است که در خط آخر این بردار ویژگی زیرگراف را برمی‌گرداند.

Algorithm4: Create a Subgraph Feature Vector based on CNN

Input: $S_G(A, B)(\hat{V}, \hat{E})$: input subgraph,
 $\{A, B\}$: target nodes pair in subgraph,
 h : feature vector of nodes in subgraph $\{h(z) | z \in \hat{V}\}$.

Output: $h(S_G(A, B))$: subgraph feature vector.

- 1: $h(\text{DummyTarget}) \leftarrow (h(A) + h(B))/2$
- 2: **for each** z **in** \hat{V} :
- 3: $d(z) \leftarrow \text{Distance}(h(z), h(\text{DummyTarget}))$
- 4: $\text{sorted nodes} \leftarrow \text{Sort subgraph nodes by } d$
- 5: $h(S_G(A, B)) \leftarrow \text{CNN}(h(\text{sorted nodes}))$
- 6: **return** $(h(S_G(A, B)))$

شکل (7): شبه‌کد روش پیشنهادی دوم ایجاد بردار ویژگی زیرگراف.

1.5. مجموعه داده‌ها

در آزمایش‌ها از مجموعه داده گراف‌های همگن Karate, USAir, Power, و Yeast استفاده شده است؛ که در بسیاری از پژوهش‌های مرتبط با مساله پیش‌بینی پیوند مورد تحقیق قرار گرفته‌اند [4], [18], [19], [22] - [24]. تمام گراف‌های مورد آزمایش در این مقاله بدون جهت و بدون وزن در نظر گرفته می‌شوند. مشخصات این گراف‌ها در جدول (2) نشان داده شده است. تراکم پیوندهای یک گراف به صورت نسبت تعداد پیوندهای موجود گراف $(|E|)$ به تعداد کل پیوندهایی که در یک گراف بدون جهت می‌تواند ایجاد شود، محاسبه می‌شود. در ادامه به بررسی چگونگی جداسازی مجموعه داده پرداخته شده است.

جدول (1): نامگذاری روش‌های مورد مقایسه: * روش پیشنهادی مقاله.

نام روش	روش استخراج زیرگراف	روش ایجاد بردار ویژگی
GAE	ندارد	ندارد
SGAE-DIS-NDP	DIS*	NDP*
SGAE-DIS-CNN	DIS*	CNN
SGAE-hhop-NDP	h-hop	NDP*
SGAE-hhop-CNN	h-hop	CNN

جدول (2): مشخصات مجموعه داده‌ها

مجموعه داده	گره	پیوند	تعداد گره‌های زیرگراف	تراکم پیوندها
Karate	34	77	6	0,1372
USAir	332	2126	14	0,0386
Power	4941	6593	3	0,0005
Yeast	2375	11693	10	0,0041

1.1.5. جداسازی مجموعه داده

در آزمایش‌های این مقاله از روش اعتبارسنجی hold out استفاده شده است. برای این کار مجموعه داده به دو دسته آموزشی و آزمایشی تقسیم می‌شود. در پیش‌بینی با استفاده از شبکه عصبی گرافی، در هر دسته آموزشی و آزمایشی، پیوندهای گراف به

پیچیدگی زمانی روش‌های پیشنهادی برای استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف از مرتبه $O(n)$ است. همان‌طور که در شکل‌های (4) و (5) نشان داده شده است، روش‌های استخراج زیرگراف DIS و h-hop دارای یک حلقه به اندازه گره‌های گراف است، به همین دلیل پیچیدگی زمانی این دو روش از مرتبه $O(n)$ خواهد بود. در روش‌های ایجاد بردار ویژگی زیرگراف NDP و مبتنی بر CNN همان‌طور که در الگوریتم‌های شکل‌های (6) و (7) نشان داده شده است، حلقه‌های تودرتو در این روش‌ها استفاده نشده و پیچیدگی زمانی این دو روش از مرتبه $O(n)$ است.

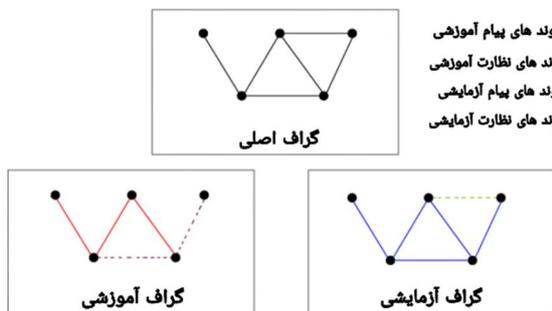
در چارچوب پیشنهادی به دلیل این که برای استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف از روش‌های متفاوتی می‌تواند استفاده شود، پیچیدگی زمانی آن بر اساس روش‌های متفاوت، متغیر خواهد بود. در صورت استفاده از هر کدام از روش‌های پیشنهاد شده برای استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف در چارچوب پیشنهادی SGAE، به این دلیل که پیچیدگی زمانی روش پایه GAE از مرتبه $O(n)$ است و فرآیند استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف به صورت جداگانه و متوالی انجام می‌شود، پیچیدگی زمانی از مرتبه $O(n)$ خواهد بود.

5. آزمایش و ارزیابی

در این بخش، سه آزمایش برای ارزیابی کارایی چارچوب پیشنهادی در پیش‌بینی پیوند بر اساس معیارهای رده‌بندی طراحی شده است. این بخش علاوه بر مقایسه میان روش پایه GAE و چارچوب پیشنهادی SGAE، روش‌های پیشنهادی استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف را نیز در چارچوب پیشنهادی مورد آزمایش قرار می‌دهد. قبل از ارائه آزمایش‌ها لازم است تا نامگذاری مناسبی برای روش‌های مورد مقایسه انجام شود. در جدول (1) این نامگذاری مشخص شده است.

آبی) که از اجتماع پیوندهای پیام و نظارت آموزشی ایجاد شده است و پیوندهای نظارت آزمایشی (نقطه‌چین سبز) نشان می‌دهد. در زمان آموزش مدل، از گراف آموزشی استفاده می‌شود، به‌صورتی که پیوندهای پیام آموزشی (خط قرمز) به مدل نشان داده شده و مدل پیوندهای نظارت آموزشی (نقطه‌چین بنفش) را پیش‌بینی می‌کند. در زمان آزمایش، پیوندهای پیام آزمایشی (خط آبی) که از اجتماع پیوندهای پیام آموزشی و پیوندهای نظارت آموزشی ایجاد می‌شوند، به مدل نشان داده شده و مدل پیوندهای نظارت آزمایشی (نقطه‌چین سبز) را پیش‌بینی می‌کند.

برای جداسازی گراف، ابتدا 46 درصد (این نسبت براساس شبکه ogbl-collab⁵ تعیین شده است) پیوندهای گراف را برای پیوندهای پیام آموزشی جدا کرده‌ایم. 10 درصد پیوندهای باقی‌مانده را برای پیوندهای آزمایشی و باقی آنها را برای پیوندهای نظارت آموزشی جدا کرده‌ایم. پیوندهای پیام آزمایشی از اجتماع پیوندهای پیام آموزشی و پیوندهای نظارت آموزشی ایجاد شده است.



شکل (9): جداسازی مجموعه داده.

2.5. برپاسازی

در پیاده‌سازی روش‌هایی که در جدول (1) معرفی شده‌اند، شبکه عصبی گرافی استفاده شده دارای یک لایه مخفی با ابعاد 256 است. بردارهای ویژگی اولیه گره‌های گراف با روش one-hot ایجاد می‌شود. نرخ یادگیری در این روش‌ها برای گراف Karate، 10^{-4} و در دیگر گراف‌ها 10^{-5} تنظیم شده است. همه مدل‌های یادگیری برای تمام گراف‌ها در 200 دوره و با بهینه‌ساز

دو دسته (1) پیوند پیام³ (2) پیوند نظارت⁴ تقسیم می‌شوند. پیوند پیام، پیوندهایی است که برای ارسال پیام به شبکه عصبی گرافی نشان داده می‌شوند؛ به این معنی که در زمان آموزش و یا آزمایش مدل پیش‌بینی کننده پیوند، مدل تنها این پیوندها را مشاهده می‌کند و براساس اطلاعات این پیوندها پیش‌بینی پیوند انجام می‌شود. پیوندهای پیام شامل دو نوع هستند:

- پیوندهای پیام آموزشی: در زمان آموزش مدل پیش‌بینی کننده پیوند تنها پیوندهای پیام آموزشی را مشاهده می‌کند و با استفاده از آنها سعی می‌کند پیوندهای نظارت آموزشی را پیش‌بینی کند.
- پیوندهای پیام آزمایشی: در زمان آزمایش مدل پیش‌بینی کننده پیوند تنها پیوندهای پیام آموزشی را مشاهده می‌کند و با استفاده از آنها سعی می‌کند پیوندهای نظارت آموزشی را پیش‌بینی کند. از اجتماع پیوندهای پیام و نظارت آموزشی ایجاد می‌شود.

پیوند نظارت، پیوندهایی است که برای نظارت بر عملکرد مدل پیش‌بینی کننده پیوند استفاده می‌شوند و به شبکه عصبی گرافی نشان داده نمی‌شوند؛ به این معنی که در زمان آموزش و یا آزمایش مدل پیوندهای نظارت حذف شده و مدل باید آنها را پیش‌بینی کند. این پیوند نیز شامل دو نوع است:

- پیوندهای نظارت آموزشی: در زمان آموزش این پیوندها حذف شده و مدل پیش‌بینی کننده پیوند آنها را پیش‌بینی می‌کند.
- پیوندهای نظارت آزمایشی: در زمان آزمایش این پیوندها حذف شده و مدل پیش‌بینی کننده پیوند آنها را پیش‌بینی می‌کند.

شکل (9) جداسازی یک گراف را نشان می‌دهد، در بالای این شکل گراف اصلی و در پایین سمت چپ گراف آموزشی با پیوندهای پیام آموزشی (خط قرمز) و پیوندهای نظارت آموزشی (نقطه‌چین بنفش) نشان داده شده است. قسمت سمت راست شکل نیز گراف آزمایشی را با پیوندهای پیام آزمایشی (خط

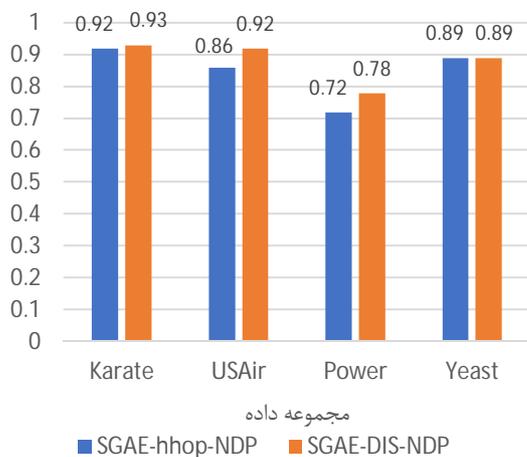
³ Message Edge

⁴ Supervision Edge

⁵ <https://ogb.stanford.edu/>

نرمال در آن استفاده شده است. به منظور مقایسه دو روش استخراج زیرگراف، استخراج زیرگراف یک بار از روش NDP و بار دیگر از روش h-hop استفاده شده است.

نتایج این آزمایش به صورت نمودار میله‌ای در شکل (10) نشان داده شده است. همان‌طور که مشاهده می‌شود، میانگین صحت روش SGAE-DIS-NDP که از روش پیشنهادی استخراج زیرگراف DIS استفاده می‌کند، در سه مجموعه داده Karate، USAir و Power بیشتر است. به طور میانگین، روش SGAE-DIS-NDP 3/25% بهتر از روش SGAE-hhop-NDP عمل کرده است.



شکل (10): نمودار متوسط صحت برای دو روش استخراج زیرگراف DIS و H-HOP در چارچوب پیشنهادی SGAE.

4.5. آزمایش دوم: ایجاد بردار ویژگی

این آزمایش با هدف ارزیابی دو روش پیشنهادی ایجاد بردار ویژگی زیرگراف در چارچوب پیشنهادی SGAE طراحی شده است. در این آزمایش از معیار ارزیابی F1-Score برای ارزیابی روش‌ها استفاده شده است.

برای مقایسه عملکرد این دو روش استخراج بردار ویژگی زیرگراف، دو دسته مقایسه صورت گرفته است. در دسته اول روش SGAE-DIS-NDP با روش SGAE-DIS-CNN (شکل (11)) و در دسته دوم روش SGAE-hhop-NDP با روش SGAE-hhop-CNN (شکل (12)) مقایسه می‌شود. در دسته اول مقایسه‌ها برای استخراج زیرگراف از روش DIS و در دسته دوم

آدام⁶ آموزش داده می‌شوند. هر مدل 10 بار اجرا شده و نتایج معیارهای ارزیابی آن در نهایت میانگین گرفته می‌شود. در هر بار اجرا تقسیم‌بندی مجموعه داده آموزشی و آزمایشی یکسان است اما ترتیب ارائه آنها به مدل به صورت تصادفی است. بعضی پارامترهای الگوریتم‌ها براساس پژوهش‌های [8] و [9] تنظیم شده است.

روش‌های پیشنهادی استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف بر پایه فاصله بین بردار ویژگی گره‌های گراف و گره *DummyTarget* عمل می‌کند؛ در این آزمایش‌ها فاصله از روش شباهت کسینوسی⁷ استفاده شده است. در شباهت کسینوسی، زاویه بین دو بردار ویژگی محاسبه می‌شود. در رابطه (6) روش محاسبه آن به وسیله بردار ویژگی گره *DummyTarget* و بردار ویژگی گره z نشان داده شده است.

$$d_{\cos}(z) = \frac{h(\text{DummyTarget}) \cdot h(z)}{\|h(\text{DummyTarget})\| \cdot \|h(z)\|} \quad (6)$$

در این آزمایش‌ها از معیارهای ارزیابی F1-score [25]، متوسط صحت⁸ [26]، دقت⁹ [25] و مساحت زیر منحنی صحت - فراخوانی¹⁰ [25] برای سنجش عملکرد روش‌ها استفاده شده است.

3.5. آزمایش اول: استخراج زیرگراف

این آزمایش با هدف مقایسه عملکرد روش استخراج زیرگراف پیشنهادی با روش h-hop در چارچوب پیشنهادی SGAE طراحی شده است. در این آزمایش به منظور ارزیابی عملکرد روش‌ها از معیار ارزیابی متوسط صحت استفاده شده است.

برای مقایسه عملکرد این دو روش استخراج زیرگراف، از دو روش SGAE-hhop-NDP و SGAE-DIS-NDP استفاده شده است. این مدل روشی برای پیش‌بینی پیوند بر پایه چارچوب SGAE است که از روش ایجاد بردار ویژگی جریمه فاصله

⁶ Adam

⁷ Cosine Similarity

⁸ Average Precision

⁹ Accuracy

¹⁰ Precision-Recall Area Under Curve

از تحلیل این دو دسته مقایسه می‌توان نتیجه گرفت که روش ایجاد بردار ویژگی NDP در صورتی که در چارچوب برای استخراج زیرگراف از یک روش متناسب با این روش مانند روش DIS استفاده شود، بهتر از شبکه عصبی کانولوشنال عمل می‌کند. اما شبکه عصبی کانولوشنال چون یک روش یادگیری است، به روش استخراج زیرگراف وابسته نیست و در اکثر حالت‌ها عملکرد قابل قبولی دارد.

5.5. آزمایش سوم

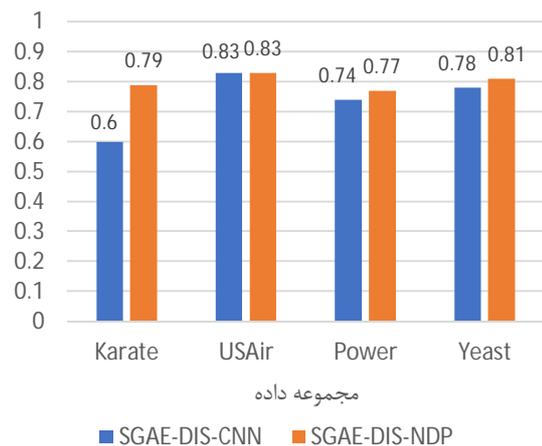
این آزمایش با هدف ارزیابی چارچوب پیشنهادی SGAE با روش پایه GAE طراحی شده است. در این آزمایش دو دسته مقایسه صورت می‌گیرد. در دسته اول مقایسه‌ها از معیارهای ارزیابی دقت، F1-score و متوسط صحت و در دسته دوم از معیار ارزیابی مساحت زیر منحنی صحت - فراخوانی استفاده شده است.

همان‌طور که در آزمایش دوم مشاهده شد، روش‌های SGAE-NDP و SGAE-hhop-CNN طبق معیار F1-score عملکرد بهتری نشان داده‌اند؛ به همین دلیل در این آزمایش برای مقایسه چارچوب پیشنهادی SGAE با روش پایه GAE، از این دو روش به نمایندگی چارچوب SGAE استفاده شده است.

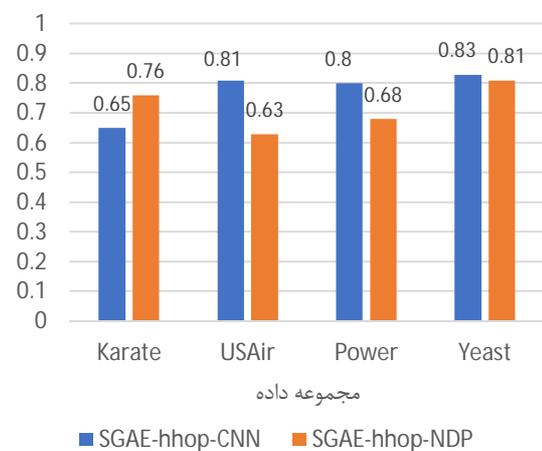
نتیجه مقایسه اول در جدول (3) نشان داده شده است. به دلیل این که آزمایش‌ها 10 بار انجام شده و نتایج میانگین گرفته شده‌اند، در این جدول انحراف معیار (به دلیل این که واریانس نتایج بسیار کوچک است از انحراف معیار استفاده شده است). نتایج به صورت \pm نیز نشان داده شده‌اند. همان‌طور که مشاهده می‌شود در هر چهار مجموعه داده، دو روش SGAE-hhop-CNN و SGAE-NDP عملکرد بهتری از روش پایه دارند. در مقایسه دوم، روش‌های SGAE-hhop-CNN و SGAE-NDP بر اساس معیار ارزیابی مساحت زیر منحنی صحت - فراخوانی با روش GAE مقایسه می‌شوند. شکل (13) منحنی‌های این سه روش را برای بهترین خروجی در 10 اجرا، به تفکیک مجموعه داده نشان داده است.

در جدول (4) مقدار مساحت زیر منحنی صحت - فراخوانی

از روش h-hop استفاده شده است. نتایج این آزمایش به صورت نمودار میله‌ای در شکل‌های (11) و (12) نشان داده شده است. همان‌طور که در شکل (11) مشاهده می‌شود، معیار F1-score برای روش SGAE-DIS-NDP در تمام مجموعه داده‌های مورد آزمایش نتیجه‌ی بهتری را نشان می‌دهد؛ اما اختلاف نتایج آن با روش SGAE-DIS-CNN در تمام مجموعه داده‌ها به‌جز Karate کم است. به طور میانگین، روش SGAE-DIS-NDP 6/25% بهتر از روش SGAE-DIS-CNN عمل کرده است. در شکل (12) معیار F1-score در اکثر مجموعه داده‌ها روش SGAE-hhop-CNN مقدار بیشتری را نشان می‌دهد و اختلاف آن با روش SGAE-hhop-NDP تقریباً زیاد است. به طور میانگین، روش SGAE-hhop-CNN 5/25% بهتر از روش SGAE-hhop-NDP عمل کرده است.



شکل (11): نتایج مقایسه اول در آزمایش دوم



شکل (12): نتایج مقایسه دوم در آزمایش دوم

روش‌ها، برای هر گراف نشان داده شده است. همان‌طور که در این جدول مشاهده می‌شود، مساحت زیر منحنی در همه مجموعه داده‌ها برای روش‌های مبتنی بر چارچوب پیشنهادی SGAE بیشتر از روش پایه GAE است. در مجموعه داده‌های Karate، Yeast و Power روش SGAE-hhop-CNN و در مجموعه داده USAir روش SGAE-DIS-NDP مقدار بیشتری را نشان می‌دهند. این نتایج نشان‌دهنده برتری چارچوب پیشنهادی بر روش پایه و تاثیر مثبت مفهوم زیرگراف بر روش پایه است.

جدول (3): نتایج مقایسه اول در آزمایش سوم

متوسط صحت	F1-Score	دقت	مدل	مجموعه داده
0,85±0,027	0,75±0	0,75±0	GAE	Karate
0,9±0,044	0,65±0,117	0,7±0,105	SGAE-hhop-CNN	
0,93±0,046	0,79±0,16	0,82±0,120	SGAE-DIS-NDP	
0,91±0,012	0,80±0,019	0,79±0,016	GAE	USAir
0,89±0,002	0,81±0,016	0,81±0,011	SGAE-hhop-CNN	
0,92±0,001	0,83±0,009	0,84±0,007	SGAE-DIS-NDP	
0,76±0,005	0,73±0,010	0,73±0,006	GAE	Power
0,85±0,027	0,80±0,010	0,78±0,012	SGAE-hhop-CNN	
0,78±0,034	0,77±0,058	0,76±0,054	SGAE-DIS-NDP	
0,87±0,009	0,77±0,013	0,78±0,009	GAE	Yeast
0,92±0,002	0,83±0,004	0,83±0,004	SGAE-hhop-CNN	
0,89±0,002	0,81±0,004	0,80±0,004	SGAE-DIS-NDP	

جدول (4): مساحت زیر منحنی صحت-فراخوانی

Yeast	Power	USAir	Karate	مدل
0,8857	0,7743	0,9160	0,8708	GAE
0,9223	0,8897	0,8945	0,9438	SGAE-hhop-CNN
0,9041	0,7918	0,9258	0,9438	SGAE-DIS-NDP

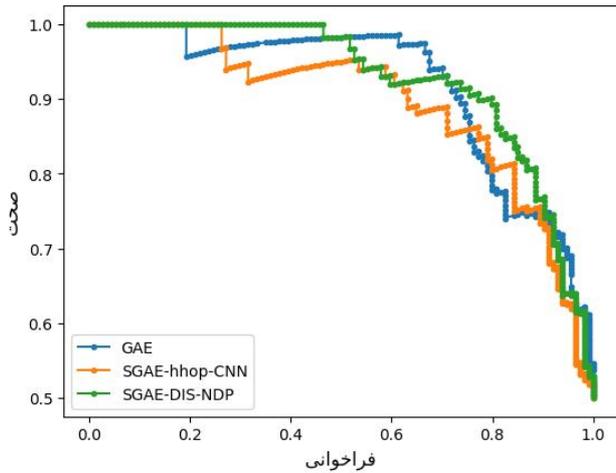
[28]. این آزمایش روی دو مجموعه داده USAir و Yeast بر اساس معیار صحت انجام شده و نتایج آن در جدول (5) نشان داده شده است. همان‌طور که مشاهده می‌شود در مجموعه داده USAir روش SGAE-hhop-CNN و در مجموعه داده Yeast روش GAT، صحت بیشتری دارد.

جدول (5): نتایج آزمایش چهارم.

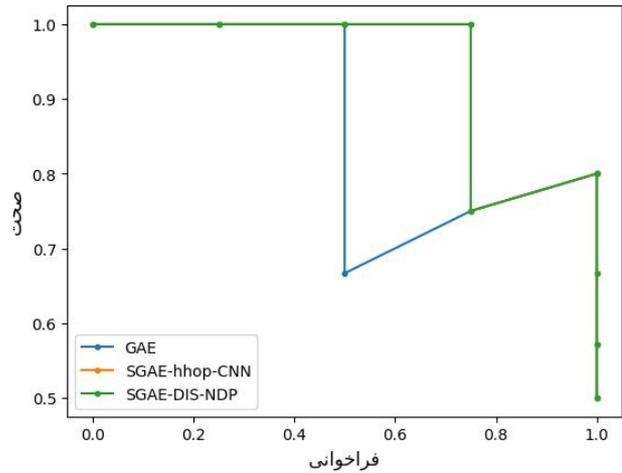
Yeast	USAir	مدل
0,74	0,82	GraphLP
0,91	0,88	GAT
0,81	0,81	GAE
0,89	0,91	SGAE-hhop-CNN

6.5. آزمایش چهارم

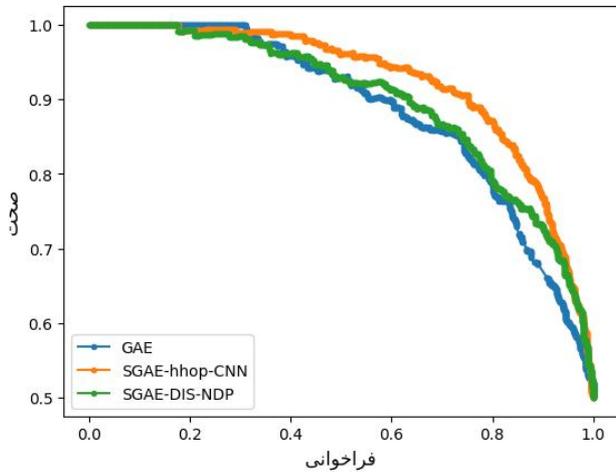
در این آزمایش به بررسی چند روش دیگر پیش‌بینی پیوند مبتنی بر GNN و مقایسه عملکرد آن‌ها با چارچوب پیشنهادی SGAE پرداخته شده است. روش‌های مورد بررسی شامل GraphLP [27] و GAT [28] است. روش GraphLP، از توانایی یادگیری ویژگی شبکه عصبی گرافی برای استخراج خودکار الگوهای ساختاری گراف‌ها و پیش‌بینی پیوند استفاده می‌کند [27]. روش GAT، برای هر گره را با ترکیب بردارهای همسایگی آن به روشی تطبیقی با وزن توجه قابل تنظیم برای همسایگی‌های مختلف یک تعبیه استخراج می‌کند و با توجه به تعبیه استخراج شده برای هر جفت گره هدف، پیش‌بینی پیوند انجام می‌شود



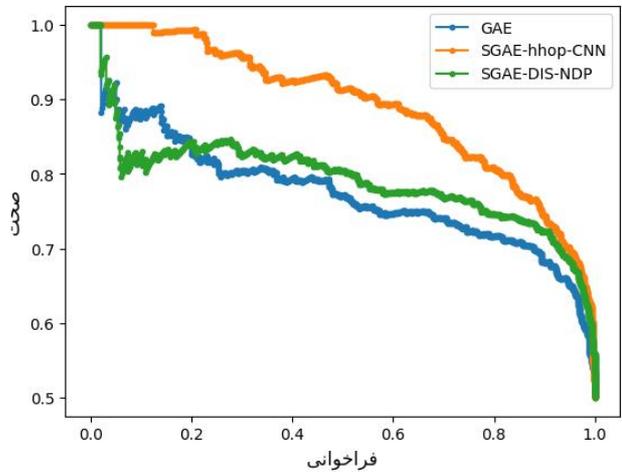
ب) منحنی صحت-فراخوانی مجموعه داده USAir



الف) منحنی صحت-فراخوانی مجموعه داده Karate



د) منحنی صحت-فراخوانی مجموعه داده Yeast



ج) منحنی صحت-فراخوانی مجموعه داده Power

شکل (13): منحنی صحت-فراخوانی

6. نتیجه‌گیری

فراخوانی، به ترتیب 5/5، 5/75 و 5/87 درصد بهتر از روش پایه GAE عمل می‌کند.

تعارض منافع: نویسندگان اعلام می‌کنند که هیچ تعارض منافعی ندارند.

در این مقاله یک چارچوب برای حل مساله پیش‌بینی پیوند با نام SGAE پیشنهاد شد؛ که از شبکه عصبی گرافی، زیرگراف و بردار ویژگی زیرگراف در آن استفاده شده است. علاوه بر چارچوب SGAE روش‌هایی برای استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف نیز پیشنهاد شده است که در چارچوب SGAE قابل استفاده است. روش‌های استخراج زیرگراف و ایجاد بردار ویژگی زیرگراف در چارچوب پیشنهادی مورد بررسی قرار گرفت. نتایج بررسی نشان داد که چارچوب پیشنهادی SGAE به طور میانگین، در معیارهای ارزیابی دقت، F1-score، متوسط صحت و مساحت زیر منحنی صحت-

- [1] A. Kumar, S.S. Singh, K. Singh, and B. Biswas, "Link prediction techniques, applications, and performance: A survey," *Physica A: Stat. Mech. Appl.*, vol. 553, p. 124289, 2020, doi: 10.1016/j.physa.2020.124289.
- [2] V. Martinez, F. Berzal, and J.-C. Cubero, "A survey of link prediction in complex networks," *ACM Comput. Surv.*, vol. 49, no. 4, pp. 1-33, 2016, doi: 10.1145/3012704.
- [3] L. Lu and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Stat. Mech. Appl.*, vol. 390, no. 6, pp. 1150-1170, 2011, doi: 10.1016/j.physa.2010.11.027.
- [4] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," *Adv. Neural Inf. Process. Syst.*, vol. 31, pp. 5171-5181, 2018, doi: 10.48550/arXiv.1802.09691.
- [5] Y. Han, D. Guan, and W. Yuan, "Learning subgraph structure with LSTM for complex network link prediction," in *Proc. 15th Int. Conf. Adv. Data Mining Appl. (ADMA)*, 2019, pp. 34-47, doi: 10.1007/978-3-030-35231-8_3.
- [6] A. Saxena, G. Fletcher, and M. Pechenizkiy, "NodeSim: Node similarity based network embedding for diverse link prediction," *EPJ Data Sci.*, vol. 11, no. 1, p. 24, 2022, doi: 10.1140/epjds/s13688-022-00336-8.
- [7] J. Feng and S. Chen, "Link prediction based on orbit counting and graph auto-encoder," *IEEE Access*, vol. 8, pp. 226773-226783, 2020, doi: 10.1109/ACCESS.2020.3045529.
- [8] T.N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016, doi: 10.48550/arXiv.1609.02907.
- [9] T.N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv:1611.07308*, 2016, doi: 10.48550/arXiv.1611.07308.
- [10] M. Zhang, P. Li, Y. Xia, K. Wang, and L. Jin, "Labeling trick: A theory of using graph neural networks for multi-node representation learning," *Adv. Neural Inf. Process. Syst.*, vol. 34, pp. 9061-9073, 2021.
- [11] K. Teru, E. Denis, and W. Hamilton, "Inductive relation prediction by subgraph reasoning," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 9448-9457, doi: 10.48550/arXiv.1911.06962.
- [12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278-2324, 1998, doi: 10.1109/5.726791.
- [13] L. Li, S. Fang, S. Bai, S. Xu, J. Cheng, and X. Chen, "Effective link prediction based on community relationship strength," *IEEE Access*, vol. 7, pp. 43233-43248, 2019, doi: 10.1109/access.2019.2908208.
- [14] D. Jin et al., "A survey of community detection approaches: From statistical modeling to deep learning," *IEEE Trans. Knowl. Data Eng.*, vol. 35, pp. 1149-1170, 2021, doi: 10.1109/TKDE.2021.3104155.
- [15] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, 2005, vol. 2, pp. 729-734, doi: 10.1109/IJCNN.2005.1555942.
- [16] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61-80, 2009, doi: 10.1109/TNN.2008.2005605.
- [17] C. Gallicchio and A. Micheli, "Graph echo state networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2010, pp. 1-8, doi: 10.1109/IJCNN.2010.5596796.
- [18] M. Zhang and Y. Chen, "Weisfeiler-Lehman Neural Machine for Link prediction," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017.
- [19] K. Ragunathan, K. Selvarajah, and Z. Kobti, "Link

- prediction by analyzing common neighbors based subgraphs using convolutional neural network,” in Proc. Eur. Conf. Artif. Intell. (ECAI), 2020, pp. 1906-1913.
- [20] K. Selvarajah, K. Rangunathan, Z. Kobti, and M. Kargar, “Dynamic network link prediction by learning effective subgraphs using CNN-LSTM,” in Proc. Int. Joint Conf. Neural Netw. (IJCNN), 2020, pp. 1-8, doi: 10.1109/IJCNN48605.2020.9207301.
- [21] Y. Freund and R.E. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting,” J. Comput. Syst. Sci., vol. 55, no. 1, pp. 119-139, 1997, doi: 10.1006/jcss.1997.1504.
- [22] Y. Wang and L. Ming, “Global path link prediction method based on improved resource allocation,” J. Phys.: Conf. Ser., vol. 2522, no. 1, p. 012023, 2023, doi: 10.1088/1742-6596/2522/1/012023.
- [23] J. Zhu et al., “SpotTarget: Rethinking the effect of target edges for link prediction in graph neural networks,” arXiv preprint arXiv:2306.00899, 2023, doi: 10.48550/arXiv.2306.00899.
- [24] N.K. Ahmed, J. Neville, R.A. Rossi, and N. Duffield, “Efficient graphlet counting for large networks,” in Proc. IEEE Int. Conf. Data Mining (ICDM), 2015, pp. 1-10, doi: 10.1109/ICDM.2015.141.
- [25] K. Abbas et al., “Application of network link prediction in drug discovery,” BMC Bioinf., vol. 22, no. 1, p. 187, Apr. 2021, doi: 10.1186/s12859-021-04082-y.
- [26] S. Scellato, A. Noulas, and C. Mascolo, “Exploiting place features in link prediction on location-based social networks,” in Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2011, pp. 1046-1054, doi: 10.1145/2020408.2020575.
- [27] X. Xian et al., “Generative graph neural networks for link prediction,” arXiv preprint arXiv:2301.00169, 2022, doi: 10.48550/arXiv.2301.00169.
- [28] W. Gu, F. Gao, X. Lou, and J. Zhang, “Link prediction via graph attention network,” arXiv preprint arXiv:1910.04807, 2019, doi: 10.48550/arXiv.1910.04807.