نشریه علمی "پدافند الکترونیکی و سایبری"
سال یازدهم، شماره ۲، تابستان ۱۴۰۲، ص۱-۱۳
شاپا الکترونیکی: ۸۹۷۹-۲۹۸۰ شاپا چاپی: ۴۳۴۷-۲۳۲۲

علمی – پژوهشی

طبقه‌بندی گره‌ها در گراف‌های استنادی با استفاده از شبکه‌های عصبی گراف

حسین حسینی[۱*]، میثم میرزایی[۲]، محمدعلی جوادزاده[۳]

۱- کارشناسی ارشد، ۲- پژوهشگر، ۳- استادیار، دانشگاه جامع امام حسین (ع)، تهران، ایران

(دریافت: ۱۴۰۲/۱۰/۰۵، بازنگری: ۱۴۰۳/۰۱/۲۵، پذیرش: ۱۴۰۲/۰۲/۱۴، انتشار: ۱۴۰۳/۰۳/۱۳)

1- DOR: https://dorl.net/dor/

ناشر: دانشگاه جامع امام حسین (ع)

چکیده

گراف‌ها، داده‌هایی هستند که روابط پیچیده بین موارد مختلفی مانند اینترنت، شبکه اجتماعی، شبکه کتابشناختی و مانند آن را در دنیای واقعی توصیف می‌کنند. یکی از مواردی که امروزه افراد بسیاری با آن سر و کار دارند، شبکه‌های اجتماعی آنلاین می‌باشد. نمایش گراف شبکه‌های اجتماعی آنلاین نظیر توییتر، وی چت و فیس بوک امروزه با کمتر از میلیاردها گره امکان‌پذیر نمی‌باشد و من باب همین موضوع، مطالعه داده‌های شبکه در مقیاس بزرگ برای محققان را به یک امر ضروری تبدیل کرده است. در مورد شبکه‌های اجتماعی، کاربران آنلاین اغلب اطلاعات محدودی دارند؛ اما برای ارائه‌دهندگان خدمات رسانه‌های اجتماعی، اطلاعات گره کاربر مانند علاقه، اعتقادات یا ویژگی‌های دیگر برای سفارشی کردن خدمات آن‌ها برای کاربران در بسیاری از برنامه‌ها مانند توصیه‌ها و جستجوی شخصی بسیار مهم است و آن را به یک چالش برای ارائه‌دهندگان خدمات تبدیل کرده است. یک راه مؤثر برای مقابله با این چالش، استنتاج اطلاعات گمشده کاربر با استفاده از ساختارهای شبکه‌ای فراگیر در رسانه‌های اجتماعی است. یکی از مهم‌ترین استنتاج‌ها در داده‌کاوی و تحلیل شبکه، طبقه‌بندی گره‌ها است که هدف آن استنتاج برچسب‌های گمشده گره‌ها بر اساس گره‌های برچسب‌گذاری شده و ساختار شبکه است. در این پژوهش وظیفه طبقه بندی گره ها بر روی مجموعه داده های شبکه با استفاده از Cora و CiteSeer ، PubMedDiabetesاستنادی مورد بررسی قرار GAT و GCN ، GraphSAGEشبکه های عصبی گراف داده شده است و به صورت کلی نتیجه حاصل شده است که شبکه عصبی بر روی مجموعه داده های شبکه استنادی ذکرشده GraphSAGE گراف برای وظیفه طبقه بندی گره ها به خوبی عمل می کند".

کلیدواژه‌ها: طبقه‌بندی گره‌ها، شبکه‌های عصبی گراف، مجموعه داده‌های شبکه استنادی

نویسنده مسئول:
krmmirzai@ihu.ac.ir

Formatted: Font: 10 pt

۱. مقدمه

با افزایش سریع منابع محاسباتی و داده‌های قابل آموزش، کارآمدی معماری‌های یادگیری عمیق در حوزه اقلیدسی از جمله تصویر، متن و صدا به اثبات رسیده است [1] [2]. در حوزه بینایی کامپیوتر، شبکه‌های عصبی کانولوشنال[۱] به طور گسترده برای حوزه‌های تحقیقاتی مختلف مانند طبقه‌بندی تصویر [3] [4]، تقسیم‌بندی معنایی [5] [6] و تشخیص عنوان تصویر [7] [8] مورد بهره‌برداری قرار می‌گیرند. در حوزه پردازش زبان طبیعی، شبکه‌های عصبی بازگشتی[۲] یا شبکه‌های طولانی کوتاه‌مدت[۳] [9] برای چندین کار مهم از جمله تجزیه و تحلیل احساسات [10] [11]، ترجمه ماشینی [12] [13] و سیستم پاسخ‌گویی به پرسش [14] [15] استفاده می‌شوند.

به عنوان یکی از ساختارهای غیراقلیدسی معمولی در ناحیه یادگیری ماشین، داده‌های گراف دارای ویژگی‌های اندازه دلخواه، ساختار توپولوژیکی پیچیده و متغیر بودن ترتیب گره‌ها می‌باشند [16]؛ بنابراین، استفاده مستقیم از پارادایم‌های یادگیری موجود مانند عملیات کانولوشن یا ادغام برای نمودار سازی داده‌های گراف دشوار است. با این حال، داده‌های گراف به دلیل توانایی قدرتمند برای نمایش اشیا و روابط آن‌ها در سناریوهای مختلف مانند تشخیص جامعه، پیش‌بینی جریان ترافیک و نمودارهای دانش، ساختاری همه‌جا حاضر و ضروری در حوزه یادگیری ماشین هستند [17]. بنا به اهمیت داده‌های گراف، این موضوع سبب شده است که تعمیم شبکه‌های عصبی موفق ذکر شده جهت تجزیه و تحلیل گراف به یک امر ضروری تبدیل شود و محققان زمان خود را وقف انجام این کار کنند. در نتیجه، شبکه‌های عصبی گراف[۴] به سرعت توسعه یافته‌اند و به پیشرفت‌های قابل‌توجهی رسیده‌اند.

در چند سال گذشته، GNNها به طور گسترده در وظایف مختلف تجزیه و تحلیل گراف، از جمله وظایف متمرکز بر گره مانند طبقه‌بندی گره و پیش‌بینی پیوند [17] [18] [19] و همچنین وظایف متمرکز بر گراف مانند تشخیص شباهت گراف و طبقه‌بندی گراف استفاده شده‌اند [20] [21] [22]. در میان وظایف ذکر شده، طبقه‌بندی گره‌ها به دلیل سناریوهای کاربردی گسترده، به یکی از رایج‌ترین جهت‌های تحقیقاتی تجزیه و تحلیل گراف تبدیل شده است. به طور خاص، هدف از وظیفه طبقه‌بندی گره، پیش‌بینی یک کلاس خاص برای هر گره بدون برچسب در گراف بر اساس اطلاعات گراف است [23]. به عنوان مثال، طبقه‌بندی گره می‌تواند موضوع تحقیقی را که هر مقاله در شبکه‌های استنادی به آن تعلق دارد، پیش‌بینی کند یا اینکه در شبکه برهمکنش پروتئین-پروتئین، به هر گره می‌توان چندین نوع هستی‌شناسی ژن اختصاص داد.

علاوه بر موارد ذکر شده، طبقه‌بندی گره در بسیاری از کاربردهای دنیای واقعی، به ویژه در شبکه اجتماعی، اهمیت نظری و ارزش کاربردی مهمی دارد. مشکل طبقه‌بندی گره با به دست آوردن یک نمایش برداری مناسب از ویژگی گره که بتواند اطلاعات ساختار شبکه را رمزگذاری کند، قابل حل می‌باشد. در حال حاضر، یادگیری عمیق و مدل‌های یادگیری مبتنی بر شبکه عصبی با موفقیت در داده‌های غنی و پیچیده در انواع چندوجهی، مانند تصویر، صدا و متن استفاده می‌شوند. این جهش عظیم را می‌توان به توانایی عالی در یادگیری جاسازی پنهان داده‌ها نسبت داد؛ با الهام از این مکانیسم یادگیری، یادگیری بازنمایی برای داده‌های شبکه توجه زیادی را در سال‌های اخیر به خود جلب کرده است. بسیاری از روش‌های تعبیه شبکه که پیشنهاد شده‌اند [24]، عملکرد امیدوارکننده‌ای را برای کاربردهای مختلف به عنوان مثال طبقه‌بندی گره، خوشه‌بندی گره، پیش‌بینی پیوند نشان می‌دهند.

تکنیک‌های جاسازی شبکه از یک بردار متراکم کم بعدی برای نمایش گره‌ها استفاده می‌کنند [25] [26] که همان‌طور که ذکر شد این یک راه کارآمد برای حل مشکلات مختلف تحلیل گراف، از جمله طبقه‌بندی گره، توصیه و پیش‌بینی لینک است. اکثر تکنیک‌های جاسازی شبکه موجود جهت طبقه‌بندی گره‌ها برای شبکه‌های گراف تک لایه استاندارد مانند DeepWalk [27]، node2vec [28] و LINE [29] و شبکه‌های عصبی گراف

---

۱ CNN
۲ RNN
۳ LSTM
۴ GNN

کلاسیک مانند GCN [30]، GAT [31] و GraphSAGE [32] طراحی شده‌اند. با این حال، بیشتر سیستم‌های تعاملی پیچیده دنیای واقعی [33] به‌عنوان شبکه‌های گراف چندلایه، از جمله شبکه‌های اجتماعی [34]، شبکه‌های استنادی-همکاری [35] که توسط چندین لایه تشکیل شده‌اند و تعاملات انواع مختلف را توصیف می‌کنند، مدل‌سازی می‌شوند. به عنوان مثال، دو کاربر می‌توانند در چندین شبکه اجتماعی مانند توییتر، فیس بوک و لینکدین به یکدیگر متصل شوند. استفاده از شبکه‌های گراف چندلایه می‌تواند توضیحات جامع و دقیق‌تری در مورد این دو کاربر ارائه دهد. هنگامی که مجموعه‌ای از گره‌ها به روش چندین نوع پیوند یا انواع رابطه به هم متصل می‌شوند، شبکه گراف چندلایه حاصل نیز شبکه گراف چندگانه یا شبکه چندگانه نامیده می‌شود [36] [37]؛ اگر گره‌ها در شبکه گراف چندگانه دارای ویژگی باشند، به چنین شبکه‌ای "گراف چندگانه نسبت داده" می‌گویند. ویژگی‌ها می‌توانند راهنمایی مفیدی برای انجام تجزیه و تحلیل داده‌های گراف به خصوص در وظیفه طبقه‌بندی گره ارائه دهند. به عنوان مثال، اگر دو کاربر در یک شبکه اجتماعی سرگرمی یا علایق مشترکی داشته باشند، این دو کاربر ممکن است به یک خوشه تعلق داشته باشند.

هدف از این مقاله، دسته‌بندی و تقسیم‌بندی داده‌ها با استفاده از الگوریتم‌های شبکه‌های عصبی گراف است تا امکان جستجو و پیدا کردن داده‌های مشابه بهبود یابد. دیگران برای دسته‌بندی داده‌ها از روش‌های سنتی الگوریتم‌های یادگیری ماشین مانند درخت تصمیم و ... استفاده می‌کردند؛ اما مشکلاتی مانند پیچیدگی محاسباتی و دشواری‌های پیاده‌سازی در این روش‌ها وجود داشت. برای حل این مشکلات، رویکردی مبتنی بر الگوریتم‌های شبکه‌های عصبی را در نظر گرفته شده است. با استفاده از این روش، داده‌ها به صورت گراف ترسیم‌شده و با استفاده از الگوریتم‌های شبکه‌های عصبی، داده‌ها به دسته‌های مشابه تقسیم‌بندی می‌شوند. ارزیابی از این کار پس از پیاده‌سازی آن نشان داد که الگوریتم شبکه عصبی GraphSAGE در مقایسه با سایر الگوریتم‌های شبکه عصبی، دقت بیشتری داشته است. نوآوری اصلی حاصل در این پژوهش، انتخاب بهترین الگوریتم با استفاده از روش‌های یکسان و مجموعه داده‌های یکسان بوده است اما سایر نوآوری‌هایی که در این پژوهش حاصل شده است به ترتیب شامل: ۱- استفاده از الگوریتم‌های شبکه‌های عصبی برای دسته‌بندی داده‌ها. ۲- افزایش دقت در پردازش داده‌ها. ۳- استفاده از روشی ساده و قابل پیاده‌سازی. در بخش‌های بعدی، جزئیات بیشتری درباره دسته‌بندی گره، الگوریتم‌های شبکه‌های عصبی و مزایا و معایب آن‌ها بیان شده است.

در ادامه در بخش ۲ به مروری بر کارهای گذشته، در بخش ۳ به روش پیشنهادی، در بخش ۴ به نتایج، در بخش ۵ به نتیجه‌گیری و در بخش ۶ به کارهای آتی پرداخته شده است.

۲. مروری بر کارهای گذشته

شبکه‌های عصبی گراف مختلف می‌توانند برای طبقه‌بندی گره‌ها از سه دسته مکانیسم کانولوشن، مکانیسم توجه و مکانیسم رمزگذار خودکار استفاده نمایند [38]؛ که در ادامه شبکه‌های عصبی گراف و مکانیسم‌های مورد استفاده در این مقاله را بیان شده است.

مکانیسم کانولوشن: مکانیسم کانولوشن گراف یکی از رایج‌ترین پارادایم‌های تجمیع اطلاعات در تجزیه و تحلیل گراف است. ایده اصلی این مکانیسم این است که از عملیات کانولوشن یا ادغام در ساختار گراف برای استخراج نمایش بالاتر برای هر گره و سپس در طبقه‌بندی گره استفاده می‌کند. مدل‌های GNN بر اساس این مکانیسم به عنوان شبکه‌های کانولوشن گراف (GCN) نشان داده می‌شوند که به ترتیب گره‌ها ارتباطی ندارند و با CNNهای روی تصاویر متفاوت هستند.

مکانیسم توجه: مکانیسم توجه گراف یکی از کاربردی‌ترین معماری‌ها در هوش مصنوعی از جمله بینایی کامپیوتری و تجزیه و تحلیل گراف است. به جای استفاده از وزن ثابت، همسایگان مختلف باید سهم‌های متفاوتی برای گره هدف داشته باشند. علاوه بر این، تعداد همسایگان برای هر گره متفاوت است. مزایای مکانیسم توجه، پرداختن به ورودی‌های متغیر و تمرکز بر مرتبط‌ترین بخش است؛ بنابراین، اعمال مکانیسم توجه در طبقه‌بندی گره‌ها طبیعی است.

Graph Convolutional Network: در سال ۲۰۱۶، توماس ان. کیپف، مکس و لینگ شبکه‌های کانولوشن گراف (GCNs) را معرفی کردند که معیار پیشرفته cora را به ۸۱,۵٪ ارتقا داد. GCN شبکه‌ای است که از لایه‌های خطی پشته‌ای با تابع فعال‌سازی تشکیل شده است. توماس ان. کیپف، مکس ولینگ تابع انتشار جدیدی را معرفی کردند که به صورت لایه عمل می‌کند و مستقیماً روی داده‌های گراف کار می‌کند. تعداد لایه‌های خطی در یک GCN اندازه همسایگی گره هدف را تعیین می‌کند که هنگام پیش‌بینی طبقه‌بندی باید در نظر گرفته شود. به عنوان مثال، یک لایه پنهان نشان می‌دهد که شبکه گراف فقط همسایگان فوری را هنگام تصمیم‌گیری طبقه‌بندی بررسی می‌کند. ورودی یک شبکه کانولوشن گراف یک ماتریس مجاورت است که نمایشگر خود گراف است. همچنین بردارهای ویژگی هر گره را به عنوان ورودی می‌گیرد. این می‌تواند به سادگی رمزگذاری یکباره ویژگی‌های هر گره باشد، در حالی که می‌توان از نسخه‌های پیچیده‌تر برای نشان دادن ویژگی‌های پیچیده یک گره استفاده کرد. یک شبکه کانولوشن گراف، یا GCN، رویکردی برای یادگیری نیمه نظارت شده بر روی داده‌های ساختاریافته گراف است. این مبتنی بر یک نوع کارآمد از شبکه‌های عصبی کانولوشن است که مستقیماً بر روی گراف‌ها کار می‌کنند. انتخاب معماری کانولوشنال از طریق یک تقریب مرتبه اول محلی از پیچیدگی‌های گراف طیفی انجام می‌شود. مدل به صورت خطی در تعداد لبه‌های گراف مقیاس می‌گیرد و نمایش لایه‌های پنهان را می‌آموزد که ساختار گراف محلی و ویژگی‌های گره‌ها را کدگذاری می‌کند.

GraphSAGE: GraphSAGE یک چارچوب استقرایی کلی است که از اطلاعات ویژگی‌های گره مانند ویژگی‌های متن برای ایجاد کارآمد جاسازی‌های گره برای داده‌های دیده نشده استفاده می‌کند.

Graph Attention Network: یک شبکه توجه گراف[۵] یک معماری شبکه عصبی است که بر روی داده‌های ساختاریافته گراف عمل می‌کند و از لایه‌های خود توجهی پوشانده شده برای رفع کاستی‌های روش‌های قبلی بر اساس پیچیدگی گراف یا تقریب آن‌ها استفاده می‌کند. با انباشتن لایه‌هایی که در آن گره‌ها می‌توانند روی ویژگی‌های همسایگی خود حضور پیدا کنند، یک GAT امکان تعیین وزن‌های مختلف برای گره‌های مختلف در یک محله را به طور ضمنی و بدون نیاز به هیچ نوع عملیات ماتریس پرهزینه مانند وارونگی یا بسته به دانستن ساختار گراف از قبل را انجام می‌دهد.

در این بخش، روندهای اخیر و برخی از پیشرفته‌ترین الگوریتم‌های طبقه‌بندی گره را مورد بحث قرار گرفته است.

DeepWalk: DeepWalk که در سال ۲۰۱۴ منتشر شد، اولین روش مبتنی بر یادگیری عمیق مهم برای نزدیک شدن به طبقه‌بندی گره‌ها بود. رویکرد DeepWalk شبیه به روشی بود که در پردازش زبان طبیعی[۶] برای استخراج تعبیه‌ها اتخاذ شد. تعبیه یک نمایش برداری از یک شی مانند یک کلمه در NLP یا یک گره در یک گراف است. برای ایجاد تعبیه‌های خود،DeepWalk مسیرهای تصادفی کوتاه شده را از داده‌های گراف برای یادگیری نمایش‌های نهفته گره‌ها انجام می‌دهد. در مجموعه داده‌هایcora، DeepWalk در آزمایش طبقه‌بندی گره‌های معیار به دقت ۶۷,۲٪ دست یافت.

Graph-BERT [39]: GCN ها برای سال‌ها معماری پیشرو بودند و تغییرات زیادی از آن‌ها متعاقباً منتشر شد. سپس، در ژانویه ۲۰۲۰، Graph-BERT وابستگی به پیوندها را حذف کرد و روشی را که معمولاً شبکه‌های گرافی نشان داده می‌شوند، دوباره قالب‌بندی کرد. این برای مقیاس‌پذیری مهم است، در حالی که دقت و کارایی بهتری را نسبت به انواع دیگر شبکه‌های عصبی گراف نشان می‌دهد.

۳. روش پیشنهادی

در این بخش به روش پیشنهادی خود خواهیم پرداخت که شامل دو بخش مجموعه داده و مراحل انجام کار می‌باشد.

۳-۱. مجموعه داده

[۵] GAT

[۶] NLP

برای وظایف طبقه‌بندی گره، پیاده‌سازی روش‌ها بر روی سه مجموعه داده معیار پرکاربرد Cora، CiteSeer و PubMed اجرا شده است که همگی شبکه‌های استنادی هستند. مجموعه داده‌ها شامل بردارهای ویژگی پراکنده برای هر سند و لیستی از پیوندهای استنادی بین اسناد هستند. پیوندهای استناد را به عنوان یال‌های (غیر جهت‌دار) در نظر می‌گیریم و یک ماتریس مجاورت باینری و متقارن A می‌سازیم. هر سند دارای یک برچسب کلاس است. در ادامه جزئیات بیشتر این مجموعه داده‌ها به تفصیل ذکر شده است. همچنین به اختصار موارد مهم آن مانند نوع، تعداد گره‌ها، تعداد لبه‌ها، تعداد کلاس‌ها و تعداد ویژگی‌ها در جدول ۱ نشان داده شده است. در جدول ۱ اطلاعات آماری مجموعه داده‌های مورد استفده در این پژوهش نشان داده شده است.

مجموعه داده PubMed Diabetes: شامل ۱۹۷۱۷ نشریه علمی از پایگاه داده PubMed مربوط به دیابت است که در یکی از سه کلاس "دیابت شیرین، تجربی"، "دیابت شیرین نوع ۱" و "دیابت شیرین نوع ۲" طبقه‌بندی شده‌اند. شبکه استناد شامل ۴۴۳۳۸ پیوند است. هر انتشار در مجموعه داده توسط یک بردار کلمه وزن‌دار TF/IDF از یک فرهنگ لغت که از ۵۰۰ کلمه منحصربه‌فرد تشکیل شده است، توصیف می‌شود.

مجموعه داده CiteSeer: شامل ۳۳۱۲ نشریه علمی است که در یکی از شش کلاس "Agents"، "AI"، "DB"، "IR"، "ML" و "HCI" طبقه‌بندی شده‌اند. شبکه استناد از ۴۷۳۲ پیوند تشکیل شده است، اگرچه ۱۷ مورد از آن‌ها منبع یا انتشار هدف دارند که در مجموعه داده وجود ندارد و تنها ۴۷۱۵ پیوند در نمودار گنجانده شده است. هر انتشار در مجموعه داده با یک بردار کلمه با ارزش ۱/۰ توصیف می‌شود که نشان‌دهنده عدم وجود / وجود کلمه مربوطه از فرهنگ لغت است. فرهنگ لغت از ۳۷۰۳ کلمه منحصربه‌فرد تشکیل شده است.

مجموعه داده Cora: شامل ۲۷۰۸ نشریه علمی است که در یکی از هفت کلاس «بر اساس مورد»، «الگوریتم‌های ژنتیک»، «شبکه‌های عصبی»، «روش‌های احتمالی»، «یادگیری تقویتی»، «یادگیری قواعد» و «نظریه» طبقه‌بندی شده‌اند. شبکه استناد شامل ۵۴۲۹ پیوند است. هر انتشار در مجموعه داده با یک بردار کلمه با ارزش ۱/۰ توصیف می‌شود که نشان‌دهنده عدم وجود / وجود کلمه مربوطه از فرهنگ لغت است. فرهنگ لغت از ۱۴۳۳ کلمه منحصربه‌فرد تشکیل شده است.

جدول (۱). اطلاعات آماری مجموعه داده‌های مورد استفاده

مجموعه داده	نوع	گره‌ها	لبه‌ها	کلاس‌ها	ویژگی‌ها
CiteSeer	شبکه استناد	۳۳۱۲	۴۷۳۲	۶	۳۷۰۳
Cora	شبکه استناد	۲۷۰۸	۵۴۲۹	۷	۱۴۳۳
Pubmed	شبکه استناد	۱۹۷۱۷	۴۴۳۳۸	۳	۵۰۰

۳-۲. مراحل انجام کار

همان طور که در روند نما ۱ مشاهده می‌شود، روش کار از ۳ مرحله اصلی تشکیل شده است و هر یک از این مراحل خود دارای زیر مراحلی می‌باشند که شرح آن در ادامه آمده است.

روند نما (۱). مراحل اجرای روش پیشنهادی

۱. **آماده‌سازی داده‌ها**:

در مرحله آماده‌سازی داده‌ها زیر مراحل شامل سه بخش بارگیری مجموعه داده مدنظر، تقسیم داده‌ها و تبدیل به آرایه‌های عددی می‌باشد که شرح هر یک به تفصیل در ادامه آمده است.

۱،۱. **بارگیری مجموعه داده مدنظر**: در این مرحله یکی از مجموعه داده‌های PubMedDiabetes، CiteSeer یا Cora را بارگذاری شده است.

۱،۲. **تقسیم داده‌ها**: در این مرحله برای آموزش مدل، بخش اعظمی از گره‌ها برای آموزش در نظر گرفته شده است و مابقی برای اعتبارسنجی و آزمایش استفاده شده است. با استفاده از نمونه‌گیری طبقه‌ای، شمارش‌های نامتعادلی به دست می‌آید و مجموعه تمرینی دارای عدم تعادل کلاسی است که ممکن است نیاز به جبران داشته باشد. با این حال، برای سادگی، عدم تعادل کلاس نادیده گرفته شده است.

۱،۳. **تبدیل به آرایه‌های عددی**: در این مرحله برای هدف طبقه‌بندی گره‌ها، از بردارهای one-hot استفاده شده است که با خروجی soft-max مدل مقایسه می‌شوند. باید از تبدیلی استفاده شود که این امکان را می‌دهد که تبدیل معکوس به راحتی انجام پذیرد تا پیش‌بینی‌ها تفسیر گردد. مجموعه داده‌های مدنظر حاوی ویژگی‌های w_X است که با کلمات موجود در آن نشریه مطابقت دارد. اگر کلمه‌ای بیش از یک‌بار در یک نشریه رخ دهد، ویژگی مربوطه روی یک تنظیم می‌شود، در غیر این صورت صفر خواهد بود.

۲. **ایجاد لایه‌های شبکه عصبی مدنظر**:

در این مرحله یکی از شبکه‌های عصبی گراف GraphSAGE، GCN و GAT انتخاب گردیده است و با ساخت پشته‌ای از لایه‌ها، مدل یادگیری ماشین طراحی شده است؛ که به تفکیک شبکه‌های عصبی گراف این موارد ذکر شده است.

GraphSAGE:

layer_sizes, generator, bias, dropout

layer_sizes لیستی از اندازه ویژگی‌های پنهان هر لایه در مدل است. در این مثال از ویژگی‌های گره پنهان ۳۲ بعدی در هر لایه استفاده شده است.

bias و dropout پارامترهای داخلی مدل هستند.

GCN:

layer_sizes, activations, generator, dropout

layer_sizes تعداد لایه‌های پنهان GCN و اندازه آن‌ها. در این حالت دو لایه GCN با هر کدام ۱۶ واحد.

activations: فعال‌سازی برای اعمال خروجی هر لایهGCN . در این حالت relu برای هر دو لایه.

dropout: نرخ خروج برای ورودی هر لایه GCN. در این مورد ۵۰ درصد.

GAT:

layer_sizes, activations, attn_heads, generator, in_dropout, attn_dropout, normalize

layer_sizes لیستی از اندازه ویژگی‌های پنهان هر لایه در مدل است. در این مثال از دو لایه GAT با ویژگی‌های گره پنهان ۸ بعدی برای لایه اول و خروجی طبقه‌بندی ۷ برای لایه دوم استفاده شده است.

activations لیستی از فعال‌سازی‌های اعمال شده بر روی خروجی هر لایه است. در این حالت برای لایه اول از relu و برای لایه دوم از softmax استفاده شده است.

attn_heads تعداد سرهای توجه در همه به جز آخرین لایه GAT در مدل است.

آرگومان‌هایی مانند bias، in_dropout، attn_dropout پارامترهای داخلی مدل هستند.

۳. **آموزش و ارزیابی**:

در مرحله آموزش و ارزیابی زیر مراحل شامل سه بخش آموزش مدل، پیش بینی با مدل و تعبیه گره‌ها می‌باشد که شرح هر یک به تفصیل در ادامه آمده است.

۳،۱. **آموزش مدل**: در این مرحله مدل واقعی را با تانسورهای ورودی و تانسورهای خروجی پیش‌بینی‌های لایه متراکم نهایی ایجاد می‌کنیم. وظیفه ما یک کار پیش‌بینی طبقه‌بندی است، بنابراین یک تابع از دست دادن متقابل آنتروپی طبقه‌بندی، مناسب است. همان‌طور که در حال آموزش مدل هستیم، می‌خواهیم عملکرد تعمیم آن را در مجموعه اعتبارسنجی نیز پیگیری کنیم که به معنای ایجاد یک تولیدکننده داده دیگر است. می‌توانیم اگر که دقت اعتبارسنجی بهبود نیافت مستقیماً از عملکرد دستور توقف زودهنگام برای توقف آموزش استفاده کنیم. اکنون لایه‌های مدل، داده‌های آموزشی، داده‌های اعتبارسنجی خود را تنظیم کرده‌ایم، بنابراین اکنون می‌توانیم مدل را با استفاده از روش برازش مدل آموزش دهیم. هنگامی که مدل را آموزش دادیم، می‌توانیم تابع loss، تابع accuracy و یا هر معیار دیگر را مشاهده کنیم. در این حالت، از تابع loss و تابع accuracy برای هر دو مجموعه آموزشی و اعتبارسنجی استفاده می‌کنیم. به عنوان آخرین بخش ارزیابی خود، مدل را در برابر مجموعه تست بررسی می‌کنیم. مجدداً داده‌های مورد نیاز برای این کار را با استفاده از روش انجام شده ذکر شده در قبل ایجاد می‌کنیم و می‌توانیم از روش ارزیابی مدل برای محاسبه مقادیر متریک برای مدل آموزش‌دیده استفاده کنیم. همان‌طور که انتظار می‌رود، این مدل در مجموعه اعتبارسنجی در طول آموزش و در مجموعه تست در اینجا به طور مشابه عمل می‌کند.

۳،۲. **پیش‌بینی با مدل**: در این مرحله پیش‌بینی‌های همه گره‌ها را دریافت می‌کنیم. از روش انجام شده ذکر شده در قبل خود برای ایجاد ورودی مورد نیاز استفاده می‌کنیم و سپس از یکی از روش‌های مدل استفاده می‌کنیم. این بار برچسب‌ها را ارائه نمی‌کنیم و در عوض فقط گره‌ها را ارائه می‌کنیم، زیرا سعی می‌کنیم این کلاس‌ها را بدون اطلاع از آن‌ها پیش‌بینی کنیم. این پیش‌بینی‌ها خروجی لایه softmax خواهند بود، بنابراین برای بدست آوردن دسته‌های نهایی، از روشی، مشخصه ویژگی هدف خود برای برگرداندن این مقادیر به دسته‌های اصلی استفاده می‌کنیم.

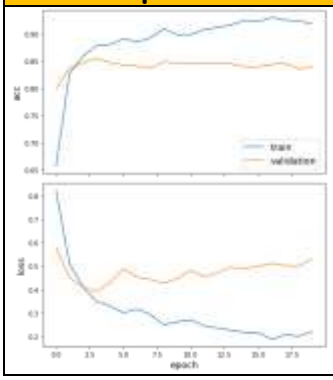
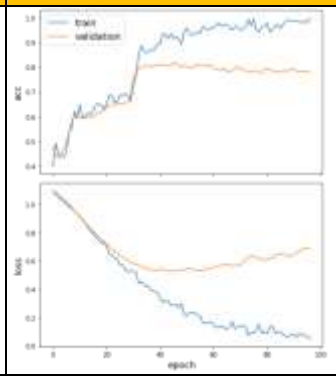
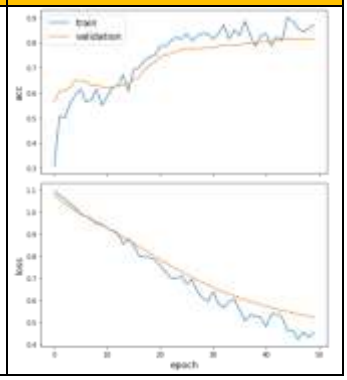
۳،۳. **تعبیه گره‌ها**: در این مرحله علاوه بر پیش‌بینی کلاس گره، تصویر دقیق‌تری از اطلاعاتی که مدل در مورد گره‌ها و همسایگی‌های آن‌ها آموخته است را نشان داده‌ایم. این جاسازی‌ها به عنوان نقاطی روی یک طرح تجسم شده‌اند که با برچسب‌های موضوع واقعی آن‌ها رنگ‌آمیزی شده‌اند. اگر مدل اطلاعات مفیدی در مورد گره‌ها بر اساس کلاس آن‌ها آموخته باشد، خوشه‌های خوبی از مقالات را در فضای جاسازی گره با مقاله‌هایی با موضوع یکسان متعلق به یک خوشه انتظار داریم ببینیم. برای ایجاد مدلی که تعبیه‌های گره را محاسبه می‌کند، از همان تانسورهای ورودی استفاده می‌کنیم و فقط تانسور خروجی را به جای لایه پیش‌بینی به یک خروجی تبدیل می‌کنیم. این تانسورها به همان لایه‌ها و وزن‌هایی متصل هستند که هنگام آموزش پیش‌بینی‌ها آموزش دادیم و بنابراین فقط از این مدل برای محاسبه/پیش‌بینی بردارهای تعبیه گره استفاده می‌کنیم. مشابه انجام پیش‌بینی‌ها برای هر گره، جاسازی‌ها را برای هر گره محاسبه می‌کنیم.

آخرین لایه شبکه‌های عصبی گراف دارای بعد خروجی با ابعاد بالایی می‌باشد، به این معنی که هر جاسازی شامل n عدد است. ترسیم مستقیم این امر مستلزم یک طرح n بعدی است که تجسم آن برای انسان دشوار است. به همین منظور باید بردارهایی با بعد ۲ بسازیم که می‌توانند بر روی یک نمودار پراکندگی دو بعدی معمولی رسم شوند. ابزارهای زیادی برای این کار کاهش ابعاد وجود دارد که دو مورد از رایج‌ترین آن‌ها، آنالیز مؤلفه‌های اصلی (PCA) (که خطی است) و جاسازی همسایه تصادفی با توزیع t (t-SNE یا TSNE) (غیر خطی) هستند. t-SNE کندتر است اما معمولاً نتایج بهتری برای رسم می‌دهد؛ بنابراین اکنون بردارهای کاهش‌یافته را به عنوان دو ستون از مقادیر جدید محاسبه می‌کنیم. مقادیر کاهش یافته شامل یک جفت اعداد (مقدار پیش‌بینی شده و مقدار حقیقی) برای هر گره است که به ازای تعداد دسته‌هایی که در یک مجموعه داده وجود دارد؛ به هر دسته، اعداد صحیح از ۱ تا n اختصاص می‌دهیم. از نظر کیفی، نمودار خوشه‌بندی خوبی را نشان می‌دهد، جایی که گره‌های یک رنگ عمدتاً با هم گروه‌بندی می‌شوند.

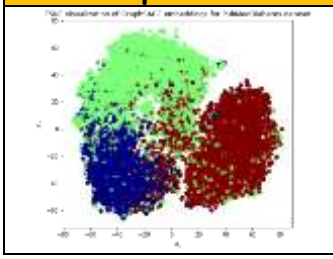
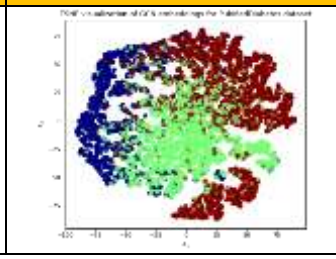
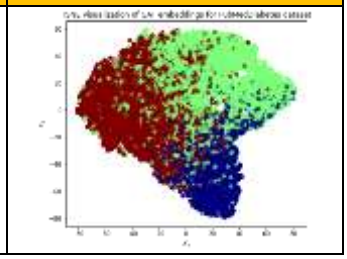
۴. نتایج

طبق آزمایش بر روی مجموعه داده PubMedDiabetes که نتایج به دست آمده آن در جدول ۲ و جدول ۳ نشان داده شده است، نتیجه گرفته شده است که به ترتیب الگوریتم‌های GraphSAGE، GCN و GAT با درصدهای ذکر شده، دقت بالایی را به دست آورده‌اند.

جدول (۲). مقایسه معیارهای Accracy و loss در زمان‌های Train و Validation بر روی نمودار هنگام اعمال الگوریتم‌های متنوع GNN بر روی مجموعه داده PubMedDiabetes

PubMedDiabetes		
GraphSAGE	GCN	GAT
		

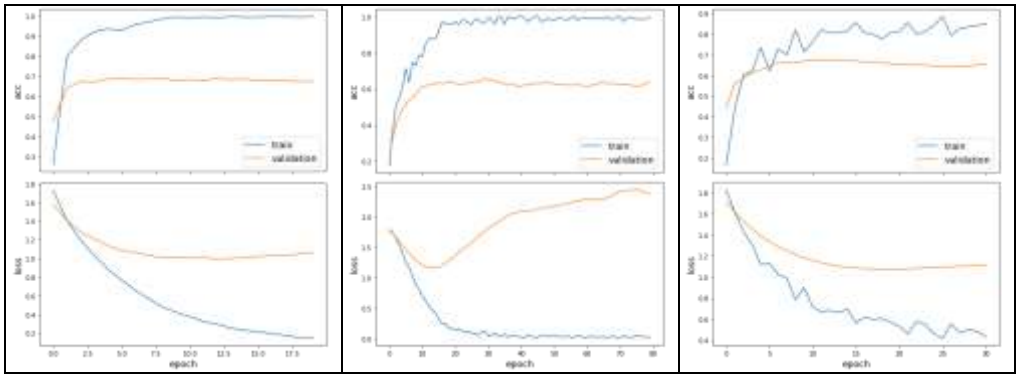
جدول (۳). مقایسه جاسازی گراف هنگام اعمال الگوریتم‌های متنوع GNN بر روی مجموعه داده PubMedDiabetes

PubMedDiabetes		
GraphSAGE	GCN	GAT
		

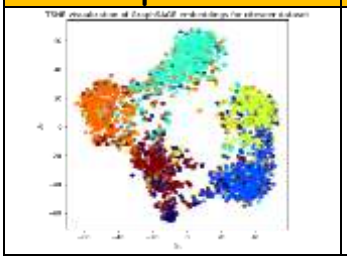
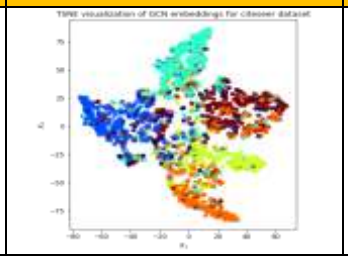
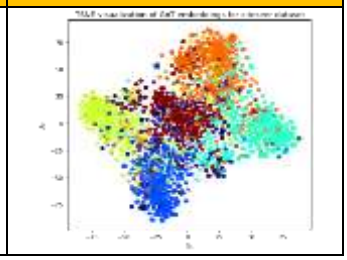
طبق آزمایش بر روی مجموعه داده CiteSeer که نتایج به دست آمده آن در جدول ۴ و جدول ۵ نشان داده شده است، نتیجه گرفته شده است که به ترتیب الگوریتم‌های GraphSAGE، GAT و GCN با درصدهای ذکر شده، دقت بالایی را به دست آورده‌اند.

جدول (۴). مقایسه معیارهای Accracy و loss در زمان‌های Train و Validation بر روی نمودار هنگام اعمال الگوریتم‌های متنوع GNN بر روی مجموعه داده CiteSeer

CiteSeer		
GraphSAGE	GCN	GAT



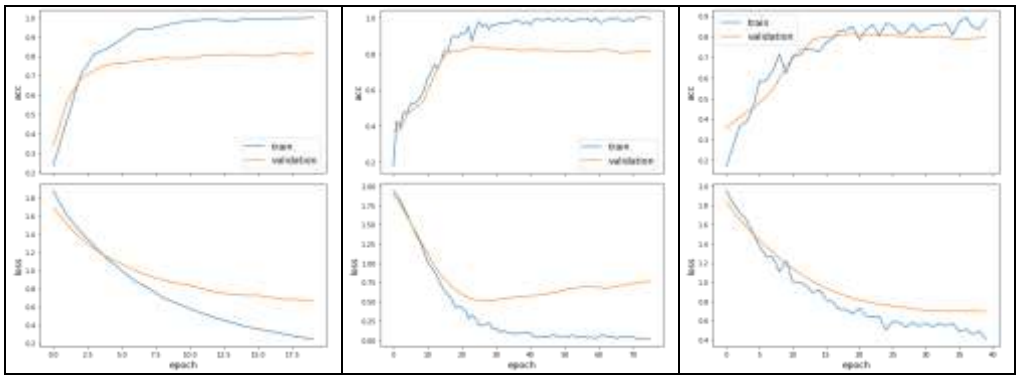
جدول (۵). مقایسه جاسازی گراف هنگام اعمال الگوریتم‌های متنوع GNN بر روی مجموعه داده CiteSeer

CiteSeer		
GraphSAGE	GCN	GAT
		

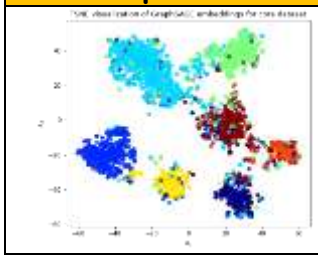
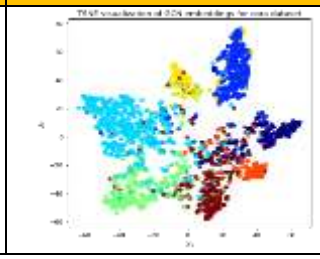
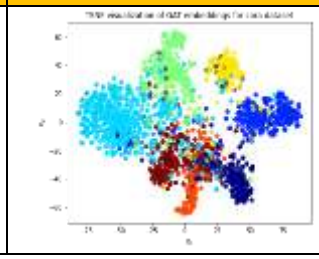
طبق آزمایش بر روی مجموعه داده Cora که نتایج به دست آمده آن در جدول ۶ و جدول ۷ نشان داده شده است، نتیجه گرفته شده است که به ترتیب الگوریتم‌های GCN، GraphSAGE و GAT با درصدهای ذکر شده، دقت بالایی را به دست آورده‌اند.

جدول (۶). مقایسه معیارهای Accracy و loss در زمان‌های Train و Validation بر روی نمودار هنگام اعمال الگوریتم‌های متنوع GNN بر روی مجموعه داده Cora

Cora		
GraphSAGE	GCN	GAT



جدول (۷). مقایسه جاسازی گراف هنگام اعمال الگوریتم‌های متنوع GNN بر روی مجموعه داده Cora

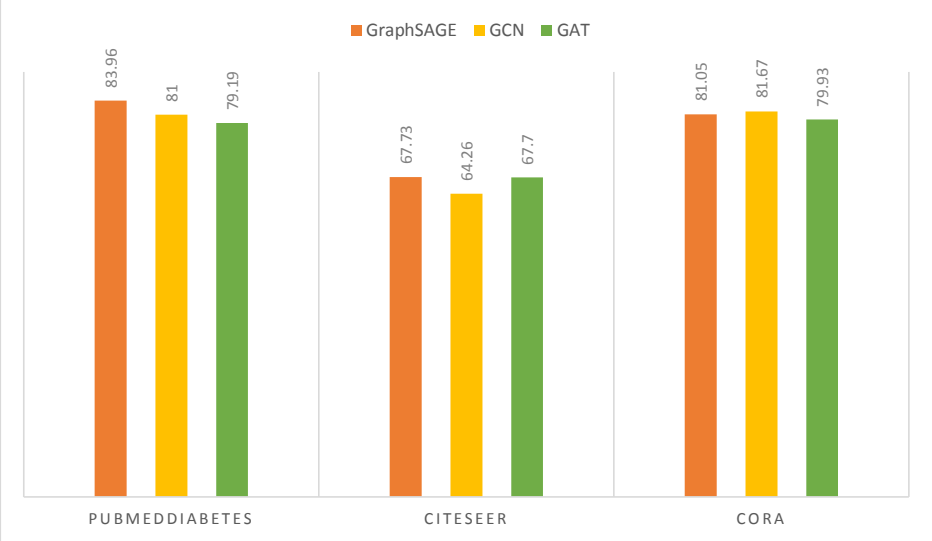
Cora		
GraphSAGE	GCN	GAT
		

در جدول ۸ و نمودار ۱ دقت به‌دست‌آمده از اعمال تمامی شبکه‌های عصبی گراف GraphSAGE، GCN و GAT بر روی مجموعه داده‌های PubMedDiabetes، CiteSeer و Cora نشان داده شده است.

جدول (۸). عملکرد الگوریتم‌های متنوع GNN در مجموعه داده‌های مختلف به صورت کمی

Node Classification		
Data Set	Algorithm	Accuracy
PubMedDiabetes	GraphSAGE	۰,۸۳۹۶
	GCN	۰,۸۱۰۰
	GAT	۰,۷۹۱۹

CiteSeer	GraphSAGE	۰,۶۷۷۳
	GCN	۰,۶۴۲۶
	GAT	۰,۶۷۷۰
Cora	GraphSAGE	۰,۸۱۰۵
	GCN	۰,۸۱۶۷
	GAT	۰,۷۹۹۳



نمودار (۱). عملکرد الگوریتم‌های متنوع GNN در مجموعه داده‌های مختلف به صورت کیفی

۵. نتیجه‌گیری

در این مقاله با استفاده از شبکه‌های عصبی گراف، پژوهشی جهت طبقه‌بندی گره‌ها انجام داده شده است. ساختار مجموعه داده‌ای که بر روی آن‌ها روش پیشنهادی ما پیاده‌سازی شده است به شکل پیوندهای استنادی بین مقالات می‌باشد. پس از آن که مدل برای پیش‌بینی، آموزش داده شد، برای دو عمل پیش‌بینی کلاس‌های گره‌ها و استفاده از وزن‌های مدل برای محاسبه جاسازی‌های برداری برای گره‌ها، از آن استفاده گردیده است. نتیجه پژوهش بدین صورت بود که شبکه عصبی گراف GraphSAGE به نسبت دو شبکه عصبی گراف دیگر از عملکرد بهتری برخوردار است.

جهت دستیابی به نتایج این پژوهش بعد از بارگذاری داده‌ها ، به سه دسته آموزش، آزمایش و اعتبارسنجی تقسیم‌بندی شده است. سپس هر کدام به آرایه عددی تبدیل شده است. در ادامه مدل شبکه عصبی با توجه به پارامترهای مورد نیاز ایجاد شده است. در نهایت مدل مورد آزمایش قرار داده شده است. پس از ایجاد شبکه‌های عصبی گراف با الگوریتم‌های GraphSAGE، GAT و GCN و اعمال آن بر روی سه مجموعه داده

PubMedDiabetes، CiteSeer و Cora نتایج نشان می‌دهد که الگوریتم GraphSAGE به نسبت دو الگوریتم دیگر بر روی مجموعه داده PubMedDiabetes دقت ۸۳/۹۶ و بر روی مجموعه داده CiteSeer دقت ۶۷/۷۳ را به دست آورده که بیشترین میزان دقت است؛ اما بر روی مجموعه داده Cora دقت ۸۱/۰۵ را به دست آورده که از دقت الگوریتم GCN کمتر می‌باشد اما به طور کلی و با در نظر گرفتن میانگین دقت، الگوریتم GraphSAGE از دقت بالایی برخوردار می‌باشد.

گفتنی است که اختلاف درصد دقت مابین اعمال روش بر روی گراف‌های جهت‌دار و غیر جهت‌دار زیر ۴ درصد می‌باشد که گراف‌های غیر جهت‌دار بیشتر هستند. و به طور کلی اکثر مقالات بر روی گراف‌های غیر جهت‌دار پژوهش خود را انجام می‌دهند. از این رو در این مقاله نیز پژوهش بر روی گراف‌های غیر جهت‌دار انجام گرفته شده است.

باید توجه داشت که هر یک از الگوریتم‌هایGraphSAGE ، GCNوGAT ، به صورت متفاوتی از لحاظ حافظه و زمان اجرایی عمل می‌کنند و این وابسته به تعداد نودها و ارتباطات آن‌ها است.

در الگوریتمGraphSAGE ، با افزایش تعداد لایه‌های مدل، مصرف حافظه و زمان اجرایی افزایش می‌یابد. همچنین، استفاده از نمونه‌برداری تصادفی در هر مرحله، مصرف حافظه را افزایش می‌دهد. اما این الگوریتم به دلیل استفاده از تکنیک‌هایی مانند mini-batchو نمونه‌برداری، از نظر زمان اجرایی به صرفه‌تر است.

در الگوریتم GCN نیز، با افزایش تعداد لایه‌ها، مصرف حافظه و زمان اجرایی افزایش می‌یابد. همچنین، در این الگوریتم باید ماتریس‌های مجاورت گراف را در هر لایه محاسبه کرد که باعث مصرف بیشتر حافظه و زمان اجرایی می‌شود.

در الگوریتمGAT ، مصرف حافظه بیشتر از GCN است زیرا برای هر نود از چندین مکانیزم توجه به همراه بردار ویژگی استفاده می‌شود و هر مکانیزم توجه، برای هر نود، یک بردار بعد بالاتر به همراه دارد. اما از نظر زمان اجرایی، این الگوریتم بهتر از GCN عمل می‌کند زیرا محاسبات ماتریس‌های مجاورت در این الگوریتم کمتر است و با استفاده از عملیات توجه، محاسبات مجدد وزن‌های گراف صورت می‌گیرد.

بنابراین، در مقایسه این سه الگوریتم، به علت ویژگی‌های خاص هر کدام، نمی‌توان به صورت کلی یک الگوریتم را بهتر از دیگری انتخاب کرد. برای استفاده از هر الگوریتم، باید به شرایط و ویژگی‌های داده‌ها و مسئله مورد نظر توجه شود تا الگوریتم بهینه‌تری انتخاب شود.

این تفاوت در دقت الگوریتم‌هایGraphSAGE ، GCNو GAT بر روی پایگاه داده‌های مختلف می‌تواند به دلیل تفاوت‌هایی در خصوصیات گراف‌های هر پایگاه داده باشد. به عبارت دیگر، خواص و ویژگی‌های مختلف گراف در هر پایگاه داده ممکن است با گراف‌های دیگر متفاوت باشد. در مثال مذکور، پایگاه داده CiteSeerشامل گرافی با خصوصیات متفاوت از دو پایگاه داده Coraو PubMedDiabetes است، این موضوع می‌تواند به عنوان یکی از دلایل تفاوت در دقت الگوریتم‌ها برای این سه پایگاه باشد.

علاوه بر این، ممکن است تعداد نودها و ارتباطات در هر پایگاه داده متفاوت باشد که می‌تواند به عنوان یک عامل دیگر برای تفاوت در دقت الگوریتم‌ها برای هر پایگاه داده مورد استفاده قرار گیرد.

بنابراین، برای انتخاب بهترین الگوریتم برای هر پایگاه داده، باید به شرایط و ویژگی‌های مختلف گراف و داده‌های مورد استفاده توجه شود و الگوریتم بهینه‌تر در هر مورد انتخاب شود.

۶. کارهای آتی

شبکه‌های عصبی گراف (GNN) برای کاربرد بالقوه در زمینه‌های متعددی که از داده‌های گراف استفاده می‌کنند، مورد بررسی قرار گرفته‌اند. با این حال، هیچ تنظیمات آموزشی استانداردی برای اطمینان از مقایسه منصفانه بین روش‌های جدید جهت انجام وظیفه طبقه‌بندی گره‌ها، از جمله معماری مدل‌های مختلف و تکنیک‌های تقویت داده‌ها وجود ندارد. برای همین می‌توان با ارائه یک معیار استاندارد که می‌تواند تنظیمات آموزشی مشابهی را

برای طبقه‌بندی گره‌ها اعمال کند، مقایسه‌ای منصفانه بین روش‌های مختلف را اعمال کرد.

۷. مراجع

[1] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, Fuad E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vol. 234, pp. 11 - 26, 2017.

[2] Qingchen Zhang, Laurence T. Yang, Zhikui Chen, Peng Li , "A survey on deep learning for big data," *Information Fusion*, vol. 42, pp. 146-157, 2018.

[3] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *26th Conference on Neural Information Processing Systems*, 2012.

[4] Weitao Wan, Yuanyi Zhong, Tianpeng Li, Jiansheng Chen, "Rethinking Feature Distribution for Loss Functions in Image Classification," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

[5] Jonathan Long, Evan Shelhamer, Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, 2015.

[6] Maria Papadomanolaki, Maria Vakalopoulou, Konstantinos Karantzalos, "A Novel Object-Based Deep Learning Framework for Semantic Segmentation of Very High-Resolution Remote Sensing Data: Comparison with Convolutional and Fully Convolutional Networks," *Remote Sensing*, vol. 11, no. 6, 2019.

[7] Long Chen, Hanwang Zhang, Jun Xiao, Liqiang Nie, Jian Shao, Wei Liu, Tat-Seng Chua, "SCA-CNN: Spatial and Channel-wise Attention in Convolutional Networks for Image Captioning," in *2017 IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[8] Jyoti Aneja, Aditya Deshpande, Alexander Schwing, "Convolutional Image Captioning," in *2018 IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[9] Sepp Hochreiter, Jürgen Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[10] Duyu Tang, Bing Qin, Ting Liu, "Document Modeling with Gated Recurrent Neural Network for Sentiment Classification," in *2015 Conference on Empirical Methods in Natural Language Processing*, 2015.

[11] Yukun Ma, Haiyun Peng, Erik Cambria, "Targeted Aspect-Based Sentiment Analysis via Embedding Commonsense Knowledge into an Attentive LSTM," in *32 AAAI Conference on Artificial Intelligence*, New Orleans, 2018.

[12] Shujie Liu, Nan Yang, Mu Li, Ming Zhou, "A Recursive Recurrent Neural Network for Statistical Machine Translation," in *52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, 2014.

[13] Jinsong Su. Shan Wu, Deyi Xiong, Yaojie Lu, Xianpei Han, Biao Zhang, "Variational Recurrent Neural Machine Translation," in *32 AAAI Conference on Artificial Intelligence*, New Orleans, 2018.

[14] Caiming Xiong, Stephen Merity, Richard Socher, "Dynamic Memory Networks for Visual and Textual Question Answering," in *33nd International Conference on Machine Learning*, 2016.

[15] Yankai Lin, Haozhe Ji, Zhiyuan Liu, Maosong Sun, "Denoising Distantly Supervised Open-Domain Question Answering," in *56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, 2018.

[16] Edward T Bullmore, Olaf Sporns, "Complex brain networks: Graph theoretical analysis of structural and functional systems," *Nature Reviews Neuroscience*, vol. 10, no. 3, 2009.

[17] Meysam Mirzaei, Aminollah Mahabadi, "Hybrid Anomaly Detection Method Using Community Detection in Graph and Feature Selection," *Journal of Electronical & Cyber Defence*, vol. 8, no. 1, pp. 17-24, 2020. (**In Persian**)

[18] James Atwood, Don Towsley, "Diffusion-Convolutional Neural Networks," in *30th Conference on Neural Information Processing Systems*, 2016.

[19] Qimai Li, Zhichao Han, Xiao-ming Wu, "Deeper Insights Into Graph Convolutional Networks for Semi-Supervised Learning," in *32nd AAAI Conference on Artificial Intelligence*, New Orleans, 2018.

[20] Muhan Zhang, Yixin Chen, "Link Prediction Based on Graph Neural Networks," in *32nd Conference on Neural Information Processing Systems*, 2018.

[21] Xiaojun Xu, Chang Liu, Qian Feng, Heng Yin, Le Song, Dawn Song, "Neural Network-based Graph Embedding for Cross-Platform Binary Code Similarity Detection," in *2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

[22] Mathias Niepert, Mohamed Ahmed, Konstantin Kutzkov, "Learning Convolutional Neural Networks for Graphs," in *33rd International Conference on Machine Learning*, 2016.

[23] Muhan Zhang, Zhicheng Cui, Marion Neumann, Yixin Chen, "An End-to-End Deep Learning Architecture for Graph Classification," in *32nd AAAI Conference on Artificial*

Attention Networks," in *5th International Conference on Learning Representations*, 2017.

[33] William L. Hamilton, Rex Ying, Jure Leskovec, "Inductive Representation Learning on Large Graphs," in *31st Conference on Neural Information Processing Systems*, 2017.

[34] Chao Li, Li Wang, Shiwen Sun, Chengyi Xia, "Identification of influential spreaders based on classified neighbors in real-world complex networks," *Applied Mathematics and Computation*, vol. 320, pp. 512-523, 2017.

[35] Etienne Gael Tajeuna, Mohamed Bouguessa, Shengrui Wang, "Modeling and Predicting Community Structure Changes in Time-Evolving Social Networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 6, pp. 1166 - 1180, 2019.

[36] Yizhou Sun, Yintao Yu, Jiawei Han, "Ranking-based Clustering of Heterogeneous Information Networks with star Network Schema," in *15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2009.

[37] Vincenzo Nicosia, Vito Latora, "Measuring and modelling correlations in multiplex networks," *PHYSICAL REVIEW E*, 2015.

[38] Zhishuang Wang, Quantong Guo, Shiwen Sun, Chengyi Xia, "The impact of awareness diffusion on SIR-like epidemics in multiplex networks," *Applied Mathematics and Computation*, vol. 349, pp. 134-147, 2019.

[39] Shunxin Xiao, Shiping Wang, Yuanfei Dai, Wenzhong Guo, "Graph neural networks in node classification: survey and evaluation," *Machine Vision and Applications*, vol. 33, no. 1, 2022.

[40] Jiawei Zhang, Haopeng Zhang, Congying Xia, Li Sun, "Graph-Bert: Only Attention is Needed for Learning Graph Representations," 2020.

Intelligence, 2018.

[24] Przemyslaw Kazienko , Tomasz Kajdanowicz, "Label-dependent node classification in the network," *Neurocomputing*, vol. 75, no. 1, pp. 199 - 209, 2012.

[25] HongYun Cai, Vincent W. Zheng, Kevin Chen-Chuan Chang, "A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616 - 1637, 2018.

[26] Qimai Li, Zhichao Han, Xiao-Ming Wu, "Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning," in *32 AAAI Conference on Artificial Intelligence*, Menlo Park, 2018.

[27] Guohao Li, Matthias Müller, Guocheng Qian, Itzel C. Delgadillo, Abdulellah Abualshour, Ali Thabet, Bernard Ghanem, "DeepGCNs: Making GCNs Go as Deep as CNNs," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, 2019.

[28] Bryan Perozzi, Rami Al-Rfou, Steven Skiena, "DeepWalk: Online Learning of Social Representations," in *20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2014.

[29] Aditya Grover, Jure Leskovec, "node2vec: Scalable Feature Learning for Networks," in *22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[30] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, Qiaozhu Mei, "LINE: Large-scale Information Network Embedding," in *24th International Conference onWorld Wide Web*, Montreal, 2015.

[31] Thomas N. Kipf, Max Welling, "Semi-Supervised Classification with Graph Convolutional Networks," in *4th International Conference on Learning Representations*, 2016.

[32] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, Yoshua Bengio, "Graph

Classification of nodes in citation graphs using graph neural networks

(Received: ……; Accepted: ……)

Abstract

Graphs are data that describe complex relationships between different things in the real world, such as the Internet, social network, bibliographic network, and so on. One of the things that many people deal with today is online social networks. The graph display of online social networks such as Twitter, WeChat and Facebook is not possible today with less than billions of nodes, and for this reason, the study of large-scale network data has become a necessity for researchers. Regarding social networks, online users often have limited information; But for social media service providers, user node information such as interest, beliefs, or other characteristics are very important to customize their services for users in many applications such as recommendations and personalized search, making it a challenge for service providers. An effective way to deal with this challenge is to infer missing user information using pervasive network structures in social media. One of the most important inferences in data mining and network analysis is node classification, which aims to infer the missing labels of nodes based on labeled nodes and network structure. In this research, we have performed the task of node classification on the PubMedDiabetes, CiteSeer and Cora citation network datasets using GraphSAGE, GCN and GAT neural networks and we have generally concluded that the GraphSAGE neural network on the network datasets The cited reference works well for the node classification task.

Keywords: Classification of nodes, graph neural networks, citation network datasets.