به نام خدا

ریاست محترم پژوهشکده علوم کامپیوتر پژوهشگاه دانش‌های بنیادی

با اهدای سلام و احترام

اینجانب مصطفی صالحی، عضو هیات علمی و دانشیار دانشگاه تهران، به منظور اجراء طرح **"تعبیه گراف در شبکه‌های چندلایه"**، تقاضای تنظیم قرارداد پژوهشی تک پروژه غیرمقیم را از آن پژوهشکده دارم.

سه نفر از متخصصین صاحب نظر در این حوزه پژوهشی عبارتند از:

- آقای دکتر حمیدرضا ربیعی، عضو هیات علمی دانشگاه صنعتی شریف، ایمیل: rabiee@sharif.edu
- آقای دکتر سامان هراتی‌زاده، عضو هیات علمی دانشگاه تهران، ایمیل: haratizadeh@ut.ac.ir
- آقای دکتر بهزاد اکبری، عضو هیات علمی دانشگاه تربیت مدرس، ایمیل: b.akbari@modares.ac.ir

به پیوست جزئیات طرح پیشنهادی آورده شده است.

با تشکر

مصطفی صالحی

عضو هیات علمی و دانشیار رسمی دانشگاه تهران

موسس آزمایشگاه شبکه‌های کامپیوتری و اطلاعاتی دانشگاه تهران، https://coinlab.ut.ac.ir/

ایمیل: mostafa_salehi@ut.ac.ir

عنوان طرح پیشنهادی

پیش‌بینی یال با استفاده از روش‌های تعبیه گراف در شبکه‌های چندلایه

مجری اصلی

مصطفی صالحی، عضو هیات علمی و دانشیار دانشگاه تهران

موسس آزمایشگاه شبکه‌های کامپیوتری و اطلاعاتی دانشگاه تهران، https://coinlab.ut.ac.ir/

آدرس: تهران، خ کارگر شمالی، روبروی خیابان دهم، دانشکده علوم و فنون نوین، طبقه سوم، اتاق ۳۳۷

تلفن همراه: ۰۹۱۲۳۴۰۳۹۷۷، تلفن ثابت: ۰۲۱۸۶۰۹۳۲۹۸، آدرس ایمیل: mostafa_salehi@ut.ac.ir

چکیده

امروزه شاهد مدل کردن بسیاری از سیستم‌های دنیای واقعی به‌عنوان شبکه‌های اطلاعاتی هستیم، هر شبکه همچون شبکه‌های اجتماعی، شبکه‌های زیستی و شبکه‌های ارتباطی، شامل مجموعه‌ای از موجودیت‌هاست که از طریق ارتباطات گوناگون می‌توانند با یکدیگر در تعامل باشند. استخراج اطلاعات از این شبکه‌ها و تحلیل آن‌ها از اهمیت ویژه‌ای برخوردار است. روش‌های زیادی برای انجام این تحلیل‌ها پیشنهاد شده که به تازگی، روش‌هایی همچون تعبیه‌گراف که از بازنمایی گره‌های گراف در یک فضای برداری استفاده می‌کنند، توجه جامعه پژوهشی را به خود منعطف ساخته‌است. یکی از کاربردهای تعبیه گراف، پیش‌بینی یال یا ارتباطات آتی است که در حوزه‌های مختلف ازجمله پزشکی، پیشنهاد کالا به مشتریان، پیشنهاد دوست به کاربران شبکه‌های اجتماعی کاربرد دارد. حال آنکه، بسیاری از شبکه‌های جدید به صورت شبکه‌های چندلایه مدل می‌شوند، این شبکه‌ها علاوه‌بر اطلاعات درون‌لایه‌ای، برخی اطلاعات بین‌لایه ای نیز ارائه می‌کنند. مدل‌هایی که برای استفاده برروی شبکه‌های تک‌لایه طراحی شده‌اند، به‌دلیل تحلیل مجزای هرلایه از شبکه، موجب ساده‌سازی بیش ازحد و از دست رفتن اطلاعات بین‌لایه‌ای می‌شوند. در طرح پیشنهادی چارچوبی ارائه می‌دهیم که قابلیت استفاده بر روی شبکه‌های تک‌لایه و چندلایه را داشته‌باشد، برای لایه‌های مختلف تعبیه متناسب با گره‌ها و یال‌های آن لایه تولید کند، سپس از تعبیه‌های به‌دست آمده و اطلاعات بین‌لایه‌ای موجود در شبکه‌های چندلایه، برای پیش‌بینی پیوندهای آتی در لایه‌های محتلف استفاده کند. همچنین در این مدل، مشکل روش‌های موجود، در نحوه استفاده از دادگانی که اطلاعاتی از ویژگی گره‌های آن‌ها دردسترس نیست، به‌صورت بهیه‌تری رفع شده و روشی جهت تولید ویژگی برای گره‌ها در این نوع از شبکه‌ها، پیشنهاد می‌گردد.

واژه‌های کلیدی:

تعبیه گراف، بازنمایی گراف، شبکه‌های چندلایه، پیش‌بینی یال، یادگیری عمیق

Title:

Graph Embedding in Multilayer Networks

Abstract:

Nowadays, many real-world systems are being modeled as information networks. Each network consists set of entities that can interact with each other through various communications. Extracting information from these networks and analyzing them is of particular importance. Graphs like social networks, biological networks, and communication networks are abundantly found in many different real-world applications. Various methods have been proposed to perform these analyses. Recently, methods like graph embedding, which uses the representation of graph nodes in vector space, have attracted the attention of the research community. Moreover, many new networks are modeled as multilayer networks. These types of networks provide some inter-layer information in addition to intra-layer information. The models designed for use on single-layer networks cause oversimplification and loss of inter-layer information, due to the separate analysis of each layer of the network. Therefore, newer models were introduced for use in multi-layer networks. One of the applications of graph embedding is to predict potential and future edges or connections that do not currently exist in the graph. Today, we are witnessing the increasing progress of the need to link prediction in various fields, such as medicine, product recommendation, and offering friends to social network users. To do so, various methods have been introduced, both in the field of single-layer and multilayer networks, but the existing methods have problems such as high time complexity, on the other hand, they do not provide a suitable solution for using networks where information about the characteristics of their nodes is not available. In this project, we present a framework that can more optimally solve the problem of the unavailability of node features, generate different embedding for different layers, then by utilizing these embeddings besides inter-layer information, try to predict future links in every layer.

Keywords:

Graph Embedding, Graph Representation Learning, Multilayer Networks, Link Prediction, Deep Learning

فهرست مطالب

۱ – مقدمه

تجزیه و تحلیل گراف در سال‌های اخیر به دلیل گستردگی شبکه‌ها در دنیای واقعی، توجه زیادی را به خود جلب کرده است. گراف‌ها (شبکه‌ها) برای بیان و نمایش اطلاعات در زمینه‌های مختلف از جمله زیست‌شناسی (شبکه‌های متقابل پروتئین-پروتئین[1]، علوم اجتماعی (شبکه های دوستی) و زبان شناسی (شبکه‌های کلمات هم‌رخداد[2]) مورد استفاده قرار گرفته است. مدل‌سازی تعاملات متقابل موجودیت‌ها در قالب گراف، محققان را قادر ساخته است تا سیستم‌های شبکه‌ای مختلف را به‌صورتی سیستماتیک و نظام‌مندتر درک کنند. به عنوان مثال، از شبکه‌های اجتماعی[3] برای اهدافی همچون دوست‌یابی، توصیه محتوا[4] و همچنین تبلیغات استفاده شده است [۱]. از دیرباز یکی از مسائل مهم مطرح در اکثر سیستم‌های دنیای واقعی، پیش‌بینی آینده آن‌هاست، که با مدل شدن این سیستم‌های پیچیده در قالب شبکه‌های اطلاعاتی، این مسئله به مسئله پیش‌بینی یال [۲] مبدل می‌گردد.

پژوهشگران در طول سال‌های اخیر دریافته‌اند که مدل کردن برخی از شبکه‌های پیچیده به شبکه‌هایی که تنها یک نوع یال و گره دارند منجربه ازدست‌رفتن بخش زیادی از اطلاعات و نهایتاً موجب تحلیل ناقص و همچنین پیش‌بینی اشتباه می‌گردد، آن‌ها برای حل این چالش شبکه‌های اطلاعاتی چندلایه را مطرح کردند که در آن، یال‌ها می‌توانند از انواع مختلفی باشند.

در ادامه، چشم اندازی کلی از انواع شبکه‌های اطلاعاتی، روش‌های تعبیه گراف و همچنین کاربردهای آن ازجمله پیش‌بینی یال در آن‌ها را ارائه می‌دهیم. هدف از این بخش ترسیم یک تصویر و دید کلی، همچنین ایجاد پس‌زمینه مناسب برای مطالب موجود در بخش‌های بعدی می‌باشد.

یک شبکه اطلاعاتی[5] یک انتزاع از اطلاعات موجود در جهان واقع است به‌گونه‌ای‌که تمرکز عمده این شبکه اطلاعاتی برروی نحوه اتصال گره‌ها و تعاملات بین آن‌ها می‌باشد. این سطح از انتزاع نه‌تنها در ارائه و مرتب‌سازی اطلاعات جهان واقع نقش دارد بلکه یک ابزار کاربردی و اصلی برای به‌دست‌آوردن اطلاعات و دانش موجود در آن می‌باشد [۳]. حال در ادامه به تعریف رسمی شبکه‌های اطلاعاتی می‌پردازیم.

---

[1] Protein-Protein interaction networks
[2] Word co-occurrence networks
[3] Social networks
[4] Content recommendation
[5] Information Networks

تعریف ۱-۱ شبکه‌های اطلاعاتی

یک گراف جهت دار به‌صورت $G(V,E)$ را در نظر بگیرید، همچنین یک تابع نگاشت نوع اشیا $t: V \to A$ و تابع نگاشت نوع یال $\phi: E \to R$ که در آن هر گره $v \in V$ متعلق به یک موجودیت خاص $t(v) \in A$ و هر یال $e \in E$ متعلق به یک رابطه خاص $\phi\ (E) \in R$ می‌باشد. حال دو یال متعلق به یک رابطه هستند، اگر نقاط ابتدایی هر دو یال از یک‌نوع گره و نقاط انتهایی آن‌ها نیز از یک‌نوع گره باشد [۳].

۱-۱- انواع شبکه‌های اطلاعاتی

به‌طور کلی می‌توان شبکه‌های اطلاعاتی را به دو گروه عمده شبکه‌های تک‌لایه[1] و یا شبکه‌های چندلایه[2] تقسیم‌بندی کرد. در ادامه به معرفی هر یک از این شبکه‌ها خواهیم پرداخت.

- **شبکه‌های تک‌لایه**

شبکه تک لایه شبکه‌ای است که دارای گره‌های یکسان می‌باشد که در آن گره‌ها فقط از یک طریق امکان ارتباط با یکدیگر را دارند. در برخی مقالات از این نوع شبکه‌ها به‌عنوان شبکه‌های همگن[3] نیز یاد می‌شود. در شکل بالا مثالی از این شبکه آورده شده که گره‌ها نویسندگان مقالات و یال‌ها ارتباط بین آن‌ها (ارجاعات) را نشان می‌دهد.

- **شبکه‌های چندلایه**

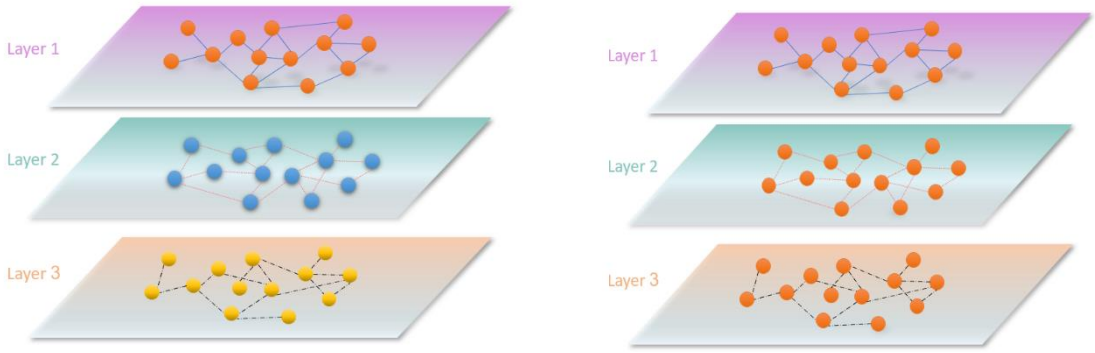
این شبکه از اجتماع چندین شبکه تک‌لایه تشکیل شده به گونه‌ای که نوع گره‌ها در هر لایه از لایه دیگر متفاوت می‌باشد و درنتیجه نوع ارتباطات در هر لایه با لایه دیگر تفاوت خواهد داشت. یک نوع خاصی از شبکه‌های چند لایه وجود دارد که شبکه‌های چندگانه[4] نام دارد، در این نوع شبکه گره‌ها در بین همه لایه‌ها یکسان می‌باشد اما این نوع ارتباطات بین گره‌هاست که در هر لایه با دیگری متفاوت می‌باشد [۴].

تعریف ۱-۲- شبکه‌های چندگانه

[1] Single- Layer
[2] Multi-Layer
[3] Homogeneous Networks
[4] Multiplex Network

فرض کنیم $G = (V, E_1 E_2 \dots E_N)$ یک شبکه چندگانه باشد که در آن V نشان دهنده مجموعه گره‌های هم‌نوع بوده و $E_j (j \epsilon 1,2,\dots,N)$ نیز مجموعه یال‌های نوع j باشد، همچنین N معادل تعداد لایه‌های این شبکه می‌باشد.

از این نوع شبکه‌ها در دنیای واقعی به وفور یافت می‌شود. برای نمونه می‌توان از شبکه مواصلاتی بین شهرها نام‌برد. اگر شهرها را به عنوان گره‌های این شبکه در نظر بگیریم روش‌های مواصلاتی مختلف بین این شهرها ازجمله هوایی، زمینی و ریلی، لایه‌های مختلف این شبکه خواهند بود که نوع ارتباط بین گره‌ها درون لایه‌ها با



شکل ۱- ۱. (سمت راست): نمایی از یک شبکه چندگانه که در آن نوع گره‌ها یکسان ولی ارتباطات در هر لایه متفاوت است. (سمت چپ): نمایی از یک شبکه چندلایه که در آن نوع گره‌ها و ارتباطات، هردو، در هر لایه متفاوت هستند.

یکدیگر متفاوت است.

باوجود تفاوت‌های یادشده بین این دو نوع شبکه، در بسیاری از موارد از کلمه شبکه‌های چندلایه به‌جای شبکه‌های چندگانه استفاده می‌شود، به‌همین‌جهت ما نیز در ادامه برای ساده‌سازی به‌جای کلمه شبکه‌های چندگانه از شبکه‌های چندلایه استفاده خواهیم کرد.

۲-۱- تعبیه گراف

جدا از بحث سیستم‌های اطلاعاتی و مزیت‌های سیستم‌های چندلایه، یکی از مسائل اصلی مطرح در این حوزه، بحث یادگیری ماشین[1] روی این گراف‌ها و یافتن راهی برای ارائه و استخراج اطلاعات موجود در ساختار گراف به یک مدل یادگیری ماشین است. به عنوان مثال، در مسئله پیش‌بینی لینک در یک شبکه اجتماعی، ممکن است شخصی بخواهد خصوصیات بین جفت گره‌ها، از قبیل قدرت رابطه یا تعداد دوستان مشترک را مورد

[1] Machine Learning

توجه قرار دهد. یا در مورد دسته‌بندی گره، ممکن است اطلاعاتی درباره موقعیت سراسری یک گره در گراف یا ساختار همسایگی محلی گره برای ما مهم باشد. چالشی که از دیدگاه یادگیری ماشینی وجود دارد این است که هیچ روش سرراستی برای رمزگذاری[1]و تبدیل چنین اطلاعاتی با بُعد بالا، از ساختار گراف به یک بردار ویژگی وجود ندارد [۵]. برای استخراج اطلاعات ساختاری از گراف‌ها، رویکردهای سنتی اغلب به خلاصه برداری از ویژگی‌های آماری گراف‌ها (به عنوان مثال، درجه یا ضرایب خوشه‌بندی[2]) تکیه دارند.

با این حال، این رویکردها محدود هستند چرا که نمی‌توانند در طول یادگیری سازگار شوند. اخیراً رویکردهای زیادی پدید آمده‌اند که به دنبال یادگیری بازنمایی‌هایی[3]هستند تا اطلاعات ساختاری یک گراف را رمزگذاری[4]کنند. ایده این رویکردها، یادگیری نگاشتی است که گره‌ها یا کل گراف (یا زیرگراف) را به عنوان نقاطی در یک فضای برداری با بعد کم‌تر بکار برند. هدف، بهینه‌سازی این نگاشت است، به گونه‌ای که روابط بین گره‌ها در فضای تعبیه شده[5] ساختار گراف اصلی را منعکس کند. حال پس از بهینه‌سازی فضای تعبیه، تعبیه‌های آموخته شده می‌توانند به عنوان ورودی برای الگوریتم‌های یادگیری ماشین مورد استفاده قرار گیرند. تمایز اصلی بین روش‌های یادگیری بازنمایی[6]و کارهای قبلی نحوه برخورد آن‌ها با مسئله بازنمایی ساختار گراف است. کارهای قبلی این مشکل را به عنوان یک مرحله پیش‌پردازشی، با استفاده از آمارهای گرافی که به‌صورت دستی برای استخراج اطلاعات ساختاری گراف‌ها تعریف می‌شدند، حل می‌کرد. در مقابل، رویکردهای یادگیری بازنمایی این مشکل را به عنوان جزئی از وظیفه یادگیری ماشین، با استفاده از یک روش داده‌محور برای یادگیری تعبیه‌هایی که ساختار گراف را رمزگذاری می‌کنند، حل می‌کند[۶].

تعبیه گراف[7]روشی است برای تبدیل گره‌ها، یال‌ها و ویژگی‌های آن‌ها به فضای برداری (بعد پایین‌تر) درعین حفظ حداکثری خواصی مانند ساختار گراف، اطلاعات گره‌ها و یال‌ها. گراف‌ها بسیار پیچیده هستند زیرا می‌توانند از نظر مقیاس، ویژگی و محتوا متفاوت باشند یک ساختار مولکولی می‌تواند به‌عنوان یک گراف کوچک، به‌صورت استاتیک نمایش داده شود، درحالی‌که یک شبکه اجتماعی می‌تواند توسط یک گراف بزرگ، متراکم و پویا به نمایش درآید. درنهایت، این امر پیدا کردن یک روش تعبیه که بتوان در شبکه‌های مختلف از آن استفاده کرد را دشوارتر می‌کند. روش‌های مختلف برای اهداف مختلف ارائه می‌شود که عملکرد هریک در

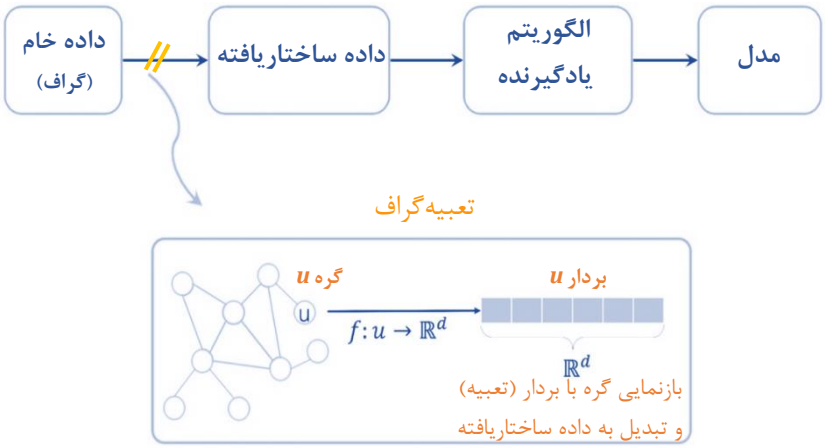
[1] Encode
[2] Clustering coefficient
[3] Representation
[4] Encoding
[5] Embedding
[6] Representation learning
[7] Graph embedding

مجموعه‌داده‌های مختلف متفاوت است [۷]. قبل از ارائه تعریف تعبیه باید اشاره کرد که اصطلاح تعبیه گراف (بازنمایی گراف)[1] در ادبیات به دو صورت مختلف مورد استفاده قرار گرفته است:

(آ). بازنمایی کل گراف به‌صورت یکجا در فضای برداری

(ب). بازنمایی هر گره به‌صورت مجزا در فضای برداری.

ما در این طرح، از رویکرد دوم استفاده می‌کنیم و تعاریف بر مبنای رویکرد دوم می‌باشد.



شکل ۱- ۲. فرآیند تبدیل داده گرافی به داده‌ی قابل‌استفاده برای الگوریتم‌های یادگیری ماشین و ساخت مدل

تعریف۱-۳- تعبیه گراف: با توجه به گراف $G(V,E)$، تعبیه گراف یک تابع نگاشت[2] است:

$f : v_i \rightarrow Y_i$ به‌گونه‌ای که $d \ll |X|$ (ابعاد فضای‌نگاشت‌شده از فضای قبلی بسیار کم‌تر است) و تابع f معیار شباهت تعریف شده در گراف G را حفظ کند (Y_i معادل تعبیه گره i و v_i: گره i، d: ابعاد تعبیه، V: گره‌های گراف، E: یال‌های آن و X بیانگر بردار ویژگی گره‌ها می‌باشد).بنابراین تعبیه، هر گره را به یک بردار ویژگی[3] با ابعاد کم‌تر نگاشت و سعی می‌کند میزان قدرت اتصال بین گره‌ها را حفظ کند. به عنوان مثال، فرض‌کنیم ماتریس S ماتریس شباهت گره‌های موجود در گراف باشد و وزن یال S_{ij} به‌عنوان مقیاسی برای شباهت بین گره‌های v_i و v_j در نظر گرفته‌شود، حال اگر $S_{ij} > S_{ik}$ باشد، v_i و v_j به‌گونه‌ای نگاشت می‌شوند که در فضای جدید گره j به گره i نزدیک‌تر از فاصله گره k به گره i باشد.

[1] Graph Representation
[2] Mapping function
[3] Feature vector

بنابراین هدف، رمزگذاری و نگاشت گره‌ها به گونه‌ای است که شباهت[1] و نزدیکی گره‌ها در فضای تعبیه، تقریبی از شباهت آن‌ها در گراف اصلی باشد:

شباهت (u, v) در گراف اصلی $\approx$ $Y_v Y_u^T$ (**شباهت (u, v) در فضای تعبیه** (ضرب داخلی دو بردار))

مسئله‌ای که اینجا به‌چشم می‌خورد، تعریف شباهت است، اینکه چه زمانی دو گره را (در شبکه اصلی) شبیه هم گوییم یا به بیان دیگر، چه زمانی دو گره، تعبیه یکسان یا شبیه به‌همی را خواهند داشت؟

(الف). حالتی‌که به‌همدیگر متصل باشند و یالی بینشان برقرار باشد.

(ب). حالتی‌که همسایه‌های بیش‌تری باهم به اشتراک بگذارند.

(پ). و یا اینکه از لحاظ ساختاری شبیه‌به‌هم باشند و نقش‌های مشابهی در شبکه ایفا کنند.

معیارهای شباهت مختلفی را می‌توان برای گره‌ها در شبکه متصور شد و هریک از این تعریف‌ها بسته‌به نوع کاربرد و هدفی که از تعبیه داریم می‌توانند درست باشند.

۳-۱- تعریف موضوع

به‌طور کلی هدف تعبیه گراف در گام اول استخراج اطلاعات گراف و نگاشت آن‌ها به یک فضای برداری با بُعد کم‌تر و در گام دوم استفاده از تعبیه‌های به‌دست‌آمده در الگوریتم‌های یادگیری ماشین جهت کاربردهای مختلف از جمله دسته‌بندی گره‌ها، پیش‌بینی یال، خوشه‌بندی یا تصویرسازی می‌باشد که در ادامه به توضیح مختصر این کاربردها خواهیم پرداخت. تاکید عمده این پژوهش ارائه یک روش تعبیه بهینه و با دقت بالا برای گراف‌های دنیای واقعی همچون شبکه‌های اجتماعی و استفاده از این تعبیه در حوزه‌های کاربردی یادشده مانند پیش‌بینی یال می‌باشد. این روش تعبیه نه‌تنها باید قابلیت استفاده روی شبکه‌های تک‌لایه را دارا باشد بلکه با ارائه راهکاری نوین قابل استفاده برروی شبکه‌های چندلایه نیز باشد و کاربردهایی از قبیل پیش‌بینی یال در شبکه‌های چندلایه را به‌انجام رساند.

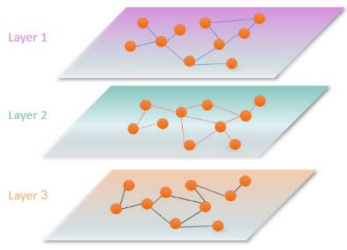
مسئله پیش‌بینی یال در شبکه‌های مختلف پیش‌بینی یال‌های بالقوه‌ای است که در حال حاضر در شبکه موجود نیستند ولی احتمال آن می‌رود که در آینده در گراف به وجود آیند. می‌توان مسئله پیش‌بینی یال را به عنوان یک مسئله دسته‌بندی نیز در نظر گرفت. در این روش هر یال را می‌توان به عنوان یک جفت رأس در نظر گرفت لذا هر نقطه متناظر با یک جفت رای در شبکه خواهد بود و باتوجه به ویژگی‌ها و اطلاعات یال مدل در بازه آموزشی[2] آموزش خواهد دید تا بتواند در بازه تست[1] به پیش‌بینی یال‌هایی که ممکن است در آینده در گراف

[1] Similarity
[2] Training Phase

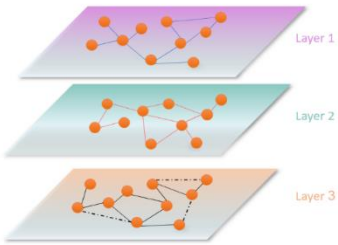
به وجود آیند بپردازد [۸]. همین رویکرد را می‌توان به شبکه‌های چندلایه هم تعمیم داد با این تفاوت که در شبکه‌های چندلایه نوع یال‌ها در هر لایه متفاوت خواهد بود لذا باید به نوع یال‌ها هم توجه کرد.

شکل پایین چارچوب و رویکر کلی مسئله پیش‌بینی یال در شبکه‌های سه لایه را نشان می‌دهد. در بخش (آ) حالت کنونی شبکه نشان داده شده است که مدل روی آن آموزش می‌بیند و بخش (ب) نتیجه پیش‌بینی مدل را نشان می‌دهد که یال‌های آتی را در لایه سه پیش‌بینی کرده‌است.

(آ) شبکه چندلایه کنونی:



(ب) شبکه چندلایه آتی:



شکل ۱- ۳. مسئله پیش‌بینی یال در شبکه‌های چندلایه (قبل و بعد از پیش‌بینی مدل)

۱-۴- اهمیت موضوع و کاربردها

به‌طور معمول، یک مدل تعریف شده برای حل مسائل مبتنی‌بر گراف، بر اساس ماتریس مجاورت[۲]گراف اصلی یا در یک فضای برداری مشتق‌شده از آن کار می‌کند. اخیراً روش‌های مبتنی‌بر نمایش شبکه‌ها در فضای برداری[۳]همچون تعبیه گراف، ضمن حفظ ویژگی‌های آن‌ها، بسیار مورد استقبال محققان قرار گرفته است [۹, ۱۰]. دستیابی به چنین تعبیه‌ای در کارهایی که درون تحلیل‌های گرافی انجام می‌پذیرد مفید است. در ادامه به معرفی این تحلیل‌ها که کاربردهای تعبیه‌گراف می‌باشند می‌پردازیم، این کاربردها به‌طورکلی به چهار دسته زیر خلاصه شود:

تصویرسازی، خوشه‌بندی، پیش‌بینی یال و دسته‌بندی گره.

---

[1] Test Phase
[2] Adjacency Matrix
[3] Vector Space Representation

- **تصویرسازی**

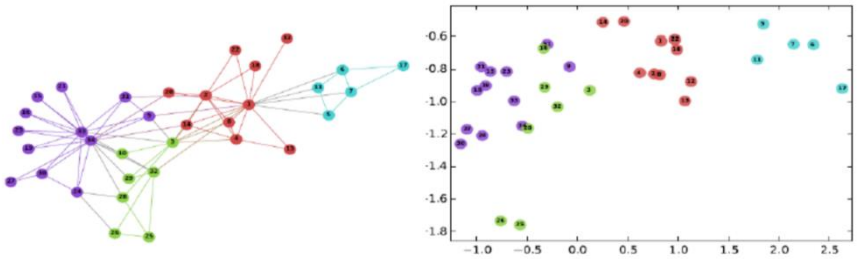
کاربرد تصویرسازی گراف می‌تواند به سال ۱۷۳۶ میلادی برگردد، زمانی‌که اویلر از آن برای حل مسئله پل‌های کونیگزبرگ[1]استفاده کرد. در سال‌های اخیر، تصویرسازی گراف کاربردهای وسیعی در مهندسی نرم‌افزار، مدارهای الکترونیکی، زیست‌شناسی و جامعه‌شناسی داشته است. باتیستا و ادز طیف وسیعی از روش‌های مورد استفاده برای ترسیم گراف‌ها و تعیین معیارهای لازم برای این منظور را بررسی کرده‌اند. هرمان و همکاران این را تعمیم داده و از منظر تصویرسازی اطلاعات گراف را مورد مطالعه قرار می‌دهند [۱]. آن‌ها روش‌های سنتی مختلفی که برای ترسیم گراف‌ها از جمله طرح‌های درختی و سه بعدی استفاده می‌شوند را مطالعه و مقایسه می‌کنند.

از آنجایی که تعبیه، گراف را در یک فضای برداری نشان می‌دهد، می‌توان از تکنیک‌های کاهش‌بُعد مانند تجزیه و تحلیل مؤلفه اصلی (PCA) برای تصویرسازی گراف استفاده کرد. نویسندگان روش $DeepWalk$ [۱۱] با تصویرسازی شبکه باشگاه کاراته زاکاری[2] مزیت روش تعبیه خود را در تصویرسازی گراف نشان دادند. همچنین شبکه همکاری نویسندگان[3]را تصویرسازی کرده و نشان دادند که این روش قادر است نویسندگان را بر اساس محتوا و موضوع مقالات خوشه‌بندی کند.

- **خوشه‌بندی**

خوشه‌بندی گراف می‌تواند دو نوع باشد: (الف) مبتنی‌بر ساختار و (ب) مبتنی‌بر ویژگی.[4] دسته اول را بازهم می‌توان به دو دسته تقسیم کرد، یعنی خوشه‌بندی مبتنی‌بر اجتماع و خوشه‌بندی مبتنی‌بر هم‌ساختاری. روش‌های مبتنی‌بر اجتماع [۱۲] هدفشان یافتن زیرگراف‌های متراکم[5](با تعداد زیادی یال‌های درون خوشه‌ای[6]و تعداد کمی

[1] Konigsberger Brucken Problem
[2] Zachary's Karate Club
[3] DBLP Co-Authorship Network
[4] Attribute based
[5] Dense Subgraphs
[6] Intra-Cluster



(ب). تصویرسازی دو بعدی تولید شده از تعبیه‌ها (آ). گراف ورودی

شکل۱- ۴. (آ). شبکه اجتماعی باشگاه کاراته زاکاری (*Zachary Karate Club*)، که درآن گره‌های مربوط به افراد درصورت دوست‌بودن به‌هم متصل می‌شوند. همچنین گره‌ها با توجه به جوامع مختلفی که در شبکه وجوددارد رنگ‌آمیزی می‌شوند. (ب). تصویرسازی دو بُعدی حاصل از تعبیه گره‌ها با استفاده از روش *DeepWalk*. فاصله بین گره‌ها در فضای تعبیه، شباهت آن‌ها را در گراف اصلی نشان‌می‌دهد، همچنین تعبیه گره‌ها باتوجه به جوامع مختلف رنگ‌آمیزی شده و به‌صورت مکانی خوشه‌بندی می‌شوند [۱۱].

یال‌های بین خوشه‌ای[1] می‌باشد. برعکس، خوشه‌بندی مبتنی‌بر هم‌ساختاری برای شناسایی گره‌ها با نقش‌های مشابه (مانند پل‌ها یا گره‌های دورافتاده[2]) طراحی شده‌است. همچنین روش‌های مبتنی‌بر ویژگی، علاوه بر پیوندهای مشاهده‌شده، از برچسب گره‌ها به منظور خوشه‌بندی آن‌ها استفاده می‌کنند.

- **دسته‌بندی گره‌ها**

اغلب در شبکه‌ها، تنها کسری از گره‌ها برچسب گذاری می‌شوند. در شبکه‌های اجتماعی، برچسب‌ها ممکن است علائق، تمایلات، عقاید یا گروه خاصی را نشان دهند. در شبکه‌های زبانی، یک سند ممکن است بر اساس موضوعات یا کلمات کلیدی برچسب گذاری شود، در حالی که برچسب موجودیت‌ها در شبکه‌های زیستی ممکن است مبتنی‌بر قابلیت‌ها و کارکرد آن‌ها باشد. به دلیل عوامل مختلف، برچسب‌ها ممکن است برای بخش بزرگی از گره‌ها ناشناخته باشند.

برای مثال، در شبکه‌های اجتماعی بسیاری از کاربران به دلیل نگرانی از حریم خصوصی برخی از اطلاعات خود را ارائه نمی‌دهند. با استفاده از گره‌های برچسب زده شده و پیوندهای موجود در شبکه، برچسب‌های گمشده قابل استنباط و استنتاج هستند. وظیفه پیش‌بینی این برچسب‌های گمشده نیز به عنوان دسته‌بندی گره‌ها شناخته می‌شود. باگات و همکاران [۱۳] روش‌های مورداستفاده در ادبیات را برای این کار بررسی کرده و آن‌ها را به دو دسته طبقه‌بندی می‌کنند، دسته اول: روش‌های مبتنی‌بر استخراج ویژگی[3] و دسته

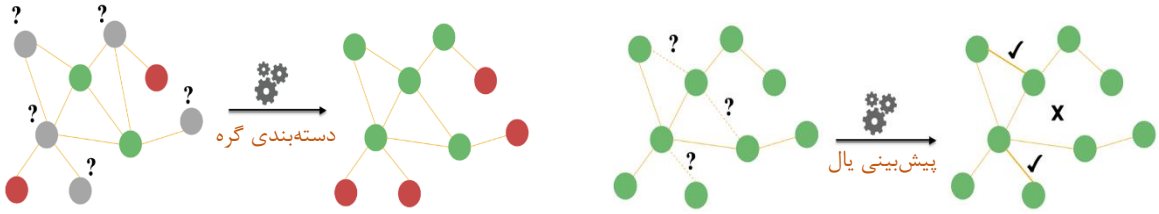
[1] Inter-cluster
[2] Outliers
[3] Feature Extraction Based Methods

دوم: قدم زدن تصادفی[1] مدل های مبتنی‌بر ویژگی، ویژگی‌هایی را برای گره‌ها براساس همسایه‌ها و گره‌های محلی‌شان ایجاد می‌کنند و سپس برای پیش‌بینی برچسب‌ها از دسته‌بندی‌هایی[2]همچون رگرسیون‌لاجستیک[3] استفاده می‌کنند. درمقابل، مدل‌های مبتنی‌بر قدم‌زنی‌تصادفی برچسب‌ها را با قدم‌زنی‌تصادفی و حرکت تصادفی از روی یک گره به‌دست می‌آورند [۱].

مقالات اخیر [۱۴] [۱۱] [۱۰] [۱] قدرت پیش‌بینی کننده تعبیه در شبکه‌های اطلاعاتی مختلف از جمله زبانی، اجتماعی، زیستی و گراف‌های همکاری را مورد ارزیابی قرار داده‌اند. آن‌ها نشان می‌دهند که تعبیه با دقت بالایی برچسب‌های گمشده را پیش‌بینی می‌کند.

- **پیش‌بینی یال**

شبکه‌ها از تعاملات مشاهده شده بین موجودیت‌ها ساخته می‌شوند که ممکن است مشاهدات ما ناقص یا نادرست باشد. این چالش در شناسایی تعاملات و پیش‌بینی اطلاعات از دست‌رفته بیش‌تر می‌شود. پیش‌بینی یال به وظیفه پیش‌بینی پیوندهای مفقود شده یا پیوندهایی که ممکن است در آینده در یک شبکه درحال گسترش



شکل ۱- ۵. دسته‌بندی گره‌ها و پیش‌بینی یال از کاربردهای تعبیه

ظاهر شوند، اشاره دارد. پیش‌بینی یال در تجزیه و تحلیل شبکه‌های بیولوژیکی بسیار فراگیر و گسترده است، جایی که در آن تأیید وجود پیوند بین گره‌ها نیازمند آزمایشات پرهزینه‌ای می‌باشد. بنابراین محدود کردن آزمایش‌ها به پیوندهایی که احتمال وقوع آن‌ها بیش‌تر است، بسیار مقرون به صرفه می‌باشد. در شبکه‌های اجتماعی از پیش‌بینی یال برای پیش‌بینی دوستی‌های احتمالی آتی استفاده می‌شود که می‌تواند در سیستم‌های توصیه‌گر[4]مورد استفاده قرار گیرد و منجربه یک تجربه کاربری رضایت بخش‌تر شود [۱۴] .

[1] Random-Walk
[2] Classifier
[3] Logistic regression
[4] Recommendation Systems

- **پیش‌بینی یال در شبکه‌های چندلایه**

پیش‌بینی ارتباطات مابین گره‌ها در شبکه‌های چندلایه نیز اهمیت بسیار زیادی دارد، چراکه شبکه‌های دنیای واقعی بسیاری وجود دارند که چندلایه یا چندگانه هستند و پیش‌بینی ارتباط آتی بین یک جفت گره روی آن شبکه‌ها می‌تواند بسیار مفید باشد. در زیر چندین مثال از این کاربردها را مرور می‌کنیم.

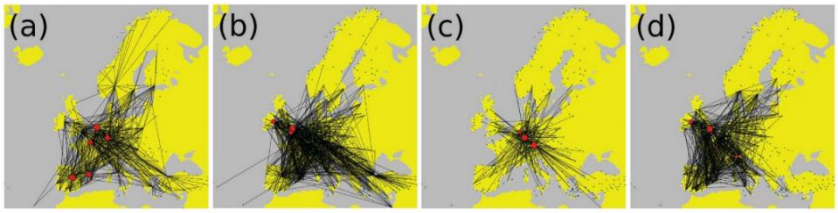
- **شبکه‌های اجتماعی**[1]

در شبکه‌های اجتماعی نحوه ارتباط افراد با یکدیگر می‌تواند متفاوت باشد، به‌طور مثال افراد در شبکه اجتماعی توییتر[2]می‌توانند به سه نوع مختلف پاسخ،[3] اشاره[4]و ریتوییت[5]با یکدیگر ارتباط برقرار کنند. به‌عنوان مثال زمانی که فردی توییتی را منتشر می‌کند افراد دیگر می‌توانند آن را ریتوییت کنند یا به آن توییت پاسخ دهند و یا فرد دیگری را مورد اشاره قرار دهند. هر یک از این ارتباطات را می‌توان به عنوان یک لایه از شبکه توییتر در نظر گرفت. درصورت پیش‌بینی صحیح یال‌ها و ارتباطات آینده می‌توان افراد مختلفی را برای یک فردی که تازه عضو آن شبکه شده پیشنهاد کرد. همچنین می‌توان از اطلاعات به‌دست آمده از این شبکه روی شبکه‌های دیگر مثل اینستاگرام استفاده کرد. در یک کاربرد دیگر می‌توان از اطلاعات به‌دست آمده از ارتباطات افراد در این شبکه‌های مجازی برای ارتباطات دنیای واقعی و بالعکس استفاده کرد [۱۵].

برای مثال شبکه توییتر به‌صورت یک شبکه چندلایه در شکل زیر به‌تصویر کشیده شده است و ارتباطات بین افراد در لایه‌های مختلف وارد شده است.

[1] Social Networks
[2] Twitter
[3] Reply
[4] Mention
[5] Retweet

- **شبکه حمل و نقل**[1] در شبکه‌های حمل و نقل شهری، بین شهری یا حتی کشوری نیز می‌توان از این شبکه‌ها استفاده نمود و برای مثال ایستگاه‌های مختلف شهری را به عنوان گره‌ها همچنین نوع ارتباط بین این ایستگاه‌ها



شکل۱- ۶. تصویر شبکه چندلایه حمل و نقل اروپا [۶۴]

را به عنوان یال این شبکه درنظر گرفت و از آنجایی که این ارتباطات می‌توانند انواع مختلفی همچون تاکسی، اتوبوس تندرو، مترو و غیره داشته باشند هر کدام یک لایه جدا از این شبکه خواهد بود. هرگونه اعمال تغییر در هر یک از گره‌ها و یا یال‌های یک لایه ممکن است در لایه‌های دیگر تأثیرگذار باشد. به‌عنوان مثال خرابی در یکی از ایستگاه‌های مترو موجب افزایش بار ترافیکی روی لایه‌های دیگر آن ایستگاه همچون اتوبوس تندرو خواهد شد و احتمالا درنتیجه موجب ایجاد لینک‌های جدید در لایه‌های دیگر خواهد شد . لذا مدل کردن این ارتباطات و پیش‌بینی برخی لینک‌ها می‌تواند در مدیریت بار ترافیکی شهری و جلوگیری از مشکلات احتمالی موثر واقع شود.

- **پزشکی، دارویی و بیوانفورماتیک**[2]

برای مثال در حوزه پزشکی می‌توان بازه‌های زمانی مختلف یک بیماری را به عنوان لایه‌های مختلف در نظر گرفت و رفتار بیماری را در لایه‌های مختلف مورد ارزیابی قرار داد و یا در حوزه دارو و کشف واکسن با تقسیم بندی نوع ارتباطات پروتئین‌ها یا نوع واکنش آن‌ها به یک نوع خاص از بیماری در لایه‌های مختلف نسبت به پیش‌بینی ارتباطات احتمالی بین پروتئین‌های مختلف اقدام کرد. همانطور که قبل‌تر نیز اشاره شد این پیش‌بینی‌ها در کاهش هزینه و زمان تولید داروها می‌تواند نقش بسیار مفیدی داشته باشد. در زیر یک مثال عملی از این کاربرد آورده شده است به گونه‌ای که در این شبکه برای درمان بیماری آسم با داشتن اطلاعاتی از سه لایه محیطی، علائم بیماری و شبکه تعاملات پروتئین-پروتئین[3]و درنهایت با پیش‌بینی ارتباطات بین آن‌ها می‌توان داروی مختص و مورد نیاز را برای هر بیمار تجویز کرد [۱۶].

[1] Transportation Network
[2] Bioanformatic
[3] Protein-Protein Interaction

۵-۱- چالش‌ها

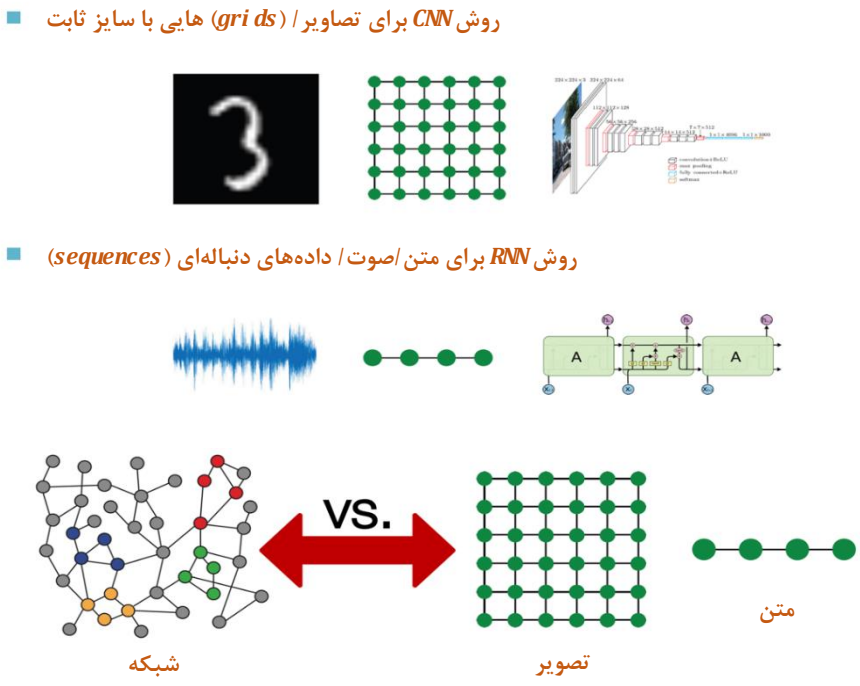
به دست آوردن بازنمایی برداری[1] از هر گره گراف ذاتاً کاری است دشوار و چالش‌های بسیاری را ایجاد می‌کند که در زیر به برخی از آن‌ها اشاره شده است:

(الف) **انتخاب ویژگی**: یک بازنمایی برداری از گره خوب، باید ساختار گراف و اتصالات بین گره‌ها را حفظ کند. اولین چالش، انتخاب ویژگی‌هایی از گراف است که تعبیه باید آن‌ها را حفظ کند. با توجه به تعدد معیارهای مسافت[2] خصوصیات و ویژگی‌های متعدد تعریف شده برای گراف‌ها، این انتخاب می‌تواند دشوار باشد، همچنین عملکرد و کارایی[3] هم ممکن است به زمینه کاربرد تعبیه و با توجه به هدفی که از آن استفاده می‌کنیم بستگی داشته باشد.

(ب) **مقیاس پذیری**[4]: بیش‌تر شبکه‌های واقعی شبکه‌هایی بزرگ هستند و حاوی هزاران یا میلیون‌ها گره و یال می‌باشند. روش‌های تعبیه باید مقیاس پذیر باشند و قادر به پردازش گراف‌های بزرگ باشند. تعریف یک مدل مقیاس پذیر می‌تواند چالش برانگیز باشد.

(پ) **ابعاد تعبیه**[5]: یافتن ابعاد بهینه بازنمایی می‌تواند سخت باشد. به عنوان مثال، ابعاد بیش‌تر ممکن است دقت در بازسازی[6] را افزایش دهد اما از نظر زمان و فضا از پیچیدگی بسیار بالایی برخوردار باشد. بسته به رویکرد ممکن است انتخاب ابعاد متفاوت باشد: به عنوان مثال، تعداد ابعاد کم‌تر ممکن است منجربه دقت بهتر در پیش‌بینی یال[7] شود چرا که مدل حاصل فقط اتصالات محلی بین گره‌ها را به‌دست خواهدآورد [۱].

ازطرف‌دیگر اکثر ابزارها و الگوریتم‌های طراحی‌شده برای یادگیری عمیق تنها روی دنباله‌ها، شبکه ($grid$)ها و

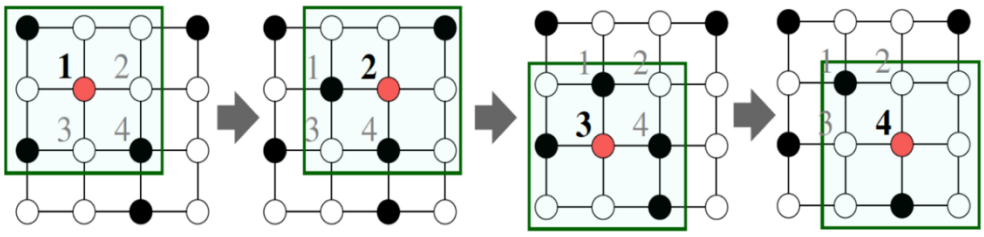


شکل ۱- ۷. مقایسه ساختار داده‌های تصویری، متن و شبکه‌های دنیای واقعی [۶]

[1] Vector representa
[2] Distance metrics
[3] Performance
[4] Scalability
[5] Dimensionality of e
[6] Reconstruction
[7] Link prediction

لتیس‌ها[1]کاربرد دارند.

همانطور که در شکل بالا مشخص است، یک تصویر را می‌توان به عنوان یک $grid$ در نظر گرفت. بنابراین روش CNN می‌تواند روی آن حرکت کند چرا که تنها با یک لتیس ($lattice$) بسیار مرسوم طرف هستیم. و یا در قسمت دوم شکل ۱-۹، یک داده صوتی را می‌بینیم، آنچه که در دنیای پردازش زبان طبیعی (NLP[2]) شاهد هستیم، چه صوت و چه متن را می‌توانیم به عنوان داده‌های متوالی یا دنباله‌ای در نظر بگیریم. در واقع در اینجا



شکل ۱- ۸. حرکت CNN روی داده‌های تصویر

هم با یک گراف زنجیره‌ای[3]طرف هستیم. اما آنچه که در شبکه‌ها وجود دارد بسیار متفاوت‌تر است. شبکه‌ها در دنیای واقعی چه از لحاظ پیچیدگی و چه از لحاظ شکل و ساختار هیچ‌گونه شباهتی به این نوع داده‌ها (عکس، متن یا صوت) ندارند (قسمت سوم شکل ۱-۹).

(ت) **عدم وجود ترتیب خاص برای گره‌ها در گراف**: در گراف‌ها و شبکه‌ها ما نمی‌توانیم یک شماره گذاری ثابت روی گره‌ها داشته باشیم چرا که گره‌ها مرتب نیستند، درضمن هیچ گره‌ای را نمی‌توانیم به عنوان گره مبدا در نظر بگیریم و شماره گذاری را از آن شروع کنیم، بنابراین بی‌شمار شماره‌گذاری متفاوت برای گره‌های آن می‌توان متصور شد. بنابراین از الگوریتم‌هایی همچون CNN که بدلیل مرتب بودن گره های تصویر و ثابت بودن جای‌گیری آن‌ها مورد استفاده قرار می‌گیرد دراین‌جا نمی‌توان استفاده کرد چراکه با شماره‌گذاری‌های متفاوت برای گره‌ها شاهد تعبیه‌های متفاوت خواهیم بود.

(ث) **پویایی شبکه‌ها و چند حالته[4]بودن گره‌ها:** مسئله دیگری که پیچیدگی شبکه‌ها را بیش‌تر می‌کند بحث پویایی شبکه‌ها می‌باشد، چراکه شبکه‌ها پیوسته در حال تغییرند و در طول زمان گره‌ها و یال‌های جدید افزوده و یا حذف می‌شوند، همچنین گره‌ها می‌توانند از انواع گوناگونی باشند، بطور مثال آنچه که در دنیای دادگان

[1] lattices
[2] Natural language processing
[3] Chain graph
[4] Multi-modal

نویسندگان مقالات داریم، گره‌ها می‌توانند مقالات، نویسندگان، موضوع و غیره باشد که می‌بینیم جنس گره‌ها باهم متفاوت‌اند.

(ج) **پیچیدگی خاص تعبیه روی شبکه‌های چندلایه:** همانگونه که اشاره شد شبکه‌های چندلایه خود به دلیل متفاوت بودن نوع گره‌ها یا یال‌ها در هر لایه دارای پیچیدگی‌های خاص می‌باشد و هرگونه تعبیه‌ای این تفاوت نوع یال‌ها یا گره‌ها را باید مورد توجه قرار دهد. از طرفی این سوال پیش می‌آید که در این شبکه‌ها لایه چه ارتباطی با یکدیگر دارند. لایه‌ها چه میزان با یکدیگر متفاوت یا شبیه هستند و اگر شباهت دارند معیار شباهت باید چه معیاری باشد. آیا تعبیه‌های هر لایه از لایه دیگر متفاوت خواهد بود یا تعبیه لایه‌ها روی همدیگر تأثیرگذار خواهند بود.

موارد ذکر شده ازجمله چالش‌هایی است که کار تعبیه گراف و به تبع آن کاربردهایی همچون پیش‌بینی یال روی شبکه‌های چندلایه را بیش از پیش با چالش مواجه خواهد کرد.

۲- پیشینه تحقیق

در یک دهه گذشته تحقیقات زیادی در زمینه تعبیه گراف با تمرکز بر طراحی الگوریتم‌های جدید تعبیه انجام شده‌است. محققان در تازه‌ترین تلاش‌های خود، الگوریتم‌های تعبیه مقیاس‌پذیر را به سمتی سوق داده‌اند که می‌توانند برروی گراف‌هایی با میلیون‌ها گره و یال اعمال شوند. ما در ادامه، پیشینه‌ای از سیر پیشرفت تحقیقات در این زمینه را ارائه می‌دهیم، سپس به بررسی یک طبقه‌بندی از تکنیک‌های تعبیه گراف شامل، (الف) روش‌های تجزیه‌ماتریس، (ب) تکنیک‌های قدم‌زنی‌تصادفی و (پ) یادگیری‌عمیق می‌پردازیم. سپس به‌جهت‌اینکه در فصل‌های بعدی روش تعبیه جدیدی پیشنهاد شده و روی یکی از کاربردهای تعبیه همچون پیش‌بینی یال مورد ارزیابی قرار می‌گیرد، در ادامه به روش‌های پیش‌بینی یال و تاریخچه آن نیز پرداخته شده است.

۲-۱- تعبیه گراف، زمینه تحقیق و تکامل

در اوایل دهه ۲۰۰۰ میلادی، محققان، الگوریتم تعبیه گراف را به‌عنوان بخشی از تکنیک‌های کاهش‌بعد[1] توسعه دادند. آن‌ها یک گراف شباهت را برای مجموعه‌ای از n نقطه D-بُعدی براساس همسایگی ایجاد می‌کنند و سپس گره‌های گراف را در یک فضای برداری d-بُعدی تعبیه می‌کنند، به گونه‌ای‌که $d \ll D$ باشد. ایده این تعبیه این بود که گره‌های متصل‌به‌هم را در فضای برداری (فضای تعبیه)، نزدیک به‌هم قرار دهند. الگوریتم‌های بردار ویژه لاپلاسین[2] و تعبیه خطی‌محلی[3] [۱۷] نمونه‌هایی از الگوریتم‌های مبتنی‌بر این روش بودند. البته مقیاس‌پذیری مشکل اصلی این روش‌ها است به‌گونه‌ای که پیچیدگی زمانی آن‌ها $O(|V|^2)$ می‌باشد.

از سال ۲۰۱۰، تحقیقات در مورد تعبیه گراف به دلیل تنک[4]بودن شبکه‌های دنیای واقعی، به سمت ایجاد تکنیک‌های تعبیه گراف مقیاس‌پذیر سوق پیدا کرد. روش‌هایی همچون شبکه‌های کانولوشنی‌گرافی(GCN[5]). رویکردهای مقیاس‌پذیر جدیدی هستند که پیچیدگی زمانی آن‌ها $O(|V|)$ می‌باشد [۱].

---

[1] Dimensionality Reduction
[2] Laplacian Eigenmaps (LE)
[3] Locally Linear Embedding (LLE)
[4] Sparse
[5] Graph Convolutional Networks (GCN)

۲-۲- دسته‌بندی انواع روش‌های تعبیه گراف

در سال‌های اخیر شاهد تحقیقات گسترده‌ای در مورد تعبیه گره‌ها هستیم که منجربه تنوع پیچیده‌ای از نمادها و مدل‌های مختلف شده‌است. بنابراین، قبل از بحث در مورد تکنیک‌های مختلف، ابتدا یک چارچوب رمزگذاری-رمزگشایی[1]یکپارچه را ارائه می‌کنیم. میتوان روش‌های مختلف تعبیه را حول دو تابع نگاشت کلیدی سازماندهی کرد: تابع رمزگذار، که هر گره را به یک بردار کم‌بُعدتر، نگاشت یا تعبیه می‌کند و تابع رمزگشا، که اطلاعات ساختاری درباره گراف را از تعبیه‌های آموخته‌شده رمزگشایی می‌کند (شکل ۲-۱).

تعریف ۲-۱- رمزگذار (ENC): تابعی است که

(۱,۲)

$$ENC : V \to \mathbb{R}^d$$

گره‌ها را به تعبیه‌های برداری $Y_i \in \mathbb{R}^d$ (که در آن Y_i تعبیه مربوط به گره $v_i \in V$ است) نگاشت می‌کند.

تعریف ۲-۲- رمزگشا (DEC): تابعی است که مجموعه‌ای از تعبیه‌های گره‌ها را گرفته و آمار گرافی و اطلاعات مشخص شده توسط کاربر را از این تعبیه‌ها رمزگشایی می‌کند. به عنوان مثال، رمزگشا ممکن است وجود یال بین گره‌ها را با توجه به تعبیه‌های آن‌ها پیش‌بینی کند [۱۸]، یا ممکن است اجتماع یک گره را پیش‌بینی کند [۲۰] [۱۹] . دراصل، رمزگشاهای زیادی می‌توان استفاده کرد. با این حال، عمدتاً از یک رمزگشای زوجی استفاده می‌کنند:

(۲,۲)

$$DEC : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$$

وقتی از رمزگشای زوجی روی یک زوج تعبیه (Y_i, Y_j) استفاده کنیم، بازسازی شباهت بین یک زوج‌گره (v_i, v_j) در گراف اصلی به‌دست می‌آید که هدف، بهینه‌سازی توابع نگاشت رمزگذار و رمزگشا برای به حداقل رساندن خطا در این بازسازی[2]خواهد بود، به گونه ای که:

$$DEC\left(ENC(v_i), ENC(v_j)\right) = DEC(Y_i, Y_j) \approx S_G(v_i, v_j) \quad (۳,۲)$$

[1] Encoder-Decoder
[2] Reconstruction Error

که در آن S_G ماتریس شباهت بین گره‌ها در گراف G می‌باشد که توسط کاربر تعریف شده است. به عبارت دیگر، ما می‌خواهیم مدل رمزگذار-رمزگشای خود را طوری بهینه کنیم تا بتوانیم شباهت بین جفت گره‌ها را در گراف اصلی (S_G) از روی تعبیه‌های (Y_i, Y_j) رمزگشایی کنیم. به عنوان مثال، می‌توان شباهت بین گره‌ها در گراف اصلی را متناظر با ماتریس مجاورت آن (A) در نظر گرفت:

$$S_G(v_i, v_j) = A_{ij} \qquad (۴,۲)$$

طوری‌که درصورت همسایه بودن دو گره، عدد یک و درغیر این‌صورت عدد صفر برای آن جفت گره در ماتریس شباهت (S_G) لحاظ کرد [۲۰]. یا ممکن است معیار تشابه بین دو گره را با توجه به احتمال هم‌رخداد بودن[1] آن دو گره در یک قدم‌زنی تصادفی با طول ثابت تعریف کرد، روشی که در مدل‌های مبتنی‌بر قدم‌زنی تصادفی شاهد آن هستیم [۱۴]. در عمل، در بیش‌تر روش‌ها هدف بازسازی، به حداقل رساندن خطای $\mathcal{L}$ برروی مجموعه‌ای از جفت‌گره‌های آموزشی (D) می‌باشد:

$$\mathcal{L} = \sum_{v_i, v_j \in D} \ell\left(DEC(Y_i, Y_j), S_G(v_i, v_j)\right) \qquad (۵,۲)$$

که در آن ℓ یک تابع زیان[2] است که توسط کاربر تعیین می‌شود و درواقع میزان اختلاف بین مقادیر شباهت به‌دست‌آمده از رمزگشایی تعبیه گره‌ها ($DEC(Y_i, Y_j)$) و شباهت بین گره‌ها در گراف اصلی ($S_G(v_i, v_j)$) را اندازه‌گیری می‌کند [۲۰]. پس از بهینه‌سازی سیستم رمزگذار- رمزگشا، می‌توانیم از رمزگذار آموزش‌دیده برای تولید تعبیه گره‌ها استفاده کنیم که از آن به‌عنوان ورودی برای کارهای یادگیری ماشین استفاده می‌شود. به‌عنوان مثال، می‌توان تعبیه‌های آموخته‌شده را به یک دسته‌بند رگرسیون‌لجستیک[3] داد تا اجتماع متعلق به یک گره را پیش‌بینی کند، یا می‌توان از فاصله بین تعبیه‌ها برای توصیف پیوندهای دوستی در یک شبکه اجتماعی استفاده کرد [۱۴].

حال با علم به رویکرد رمزگذار-رمزگشا، در ادامه یک دسته‌بندی از رویکردهای تعبیه ارائه می‌کنیم و روش‌های تعبیه را به سه دسته کلی طبقه‌بندی می‌کنیم: (۱). مبتنی‌بر تجزیه‌ماتریس، (۲). مبتنی‌بر قدم‌زنی‌تصادفی و (۳). مبتنی‌بر یادگیری‌عمیق. سپس ویژگی‌های هر یک از این دسته‌ها را توضیح می‌دهیم. همچنین نمونه‌هایی از هرگروه در جدول ۲-۱، ارائه شده است.

---

[1] Co-Occurrence
[2] Loss Function
[3] Logistic Regression Classifier

۲-۳- روش‌های مبتنی‌بر تجزیه‌ماتریس

روش‌های اولیه یادگیری تعبیه‌گره‌ها تا حد زیادی روی رویکردهای مبتنی‌بر تجزیه‌ماتریس متمرکز شده‌اند، که به‌طور مستقیم از تکنیک‌های کلاسیک برای کاهش‌بُعد الهام گرفته می‌شوند [۲۰]. الگوریتم‌های مبتنی‌بر تجزیه، اتصالات بین گره‌ها را به‌صورت ماتریس نشان می‌دهند و این ماتریس را برای به‌دست آوردن تعبیه ماتریس حاصل، تجزیه (فاکتورگیری) می‌کنند. ماتریس مورد استفاده برای نشان دادن تشابه بین گره‌ها می‌تواند شامل ماتریس مجاورت، ماتریس لاپلاسی[1] ماتریس احتمال انتقال گره[2] و یا ماتریس تشابه $Katz$ [3] باشد [۱].

۱-۲-۳- برداروِیژه لاپلاسی

یکی از اولین و شناخته‌شده‌ترین نمونه‌ها، تکنیک بردار ویژه لاپلاسی (LE) [۲۱] است که می‌توانیم در چارچوب رمزگذار-رمزگشا، تابع رمزگذار و رمزگشای آن را بدین صورت تعریف کنیم:
تابع رمزگذار این روش یک ماتریس لاپلاسین و همچنین تابع رمزگشای آن بدین صورت می‌باشد:

(۶,۲)

$$DEC\left(Y_i, Y_j\right) = \left\|Y_i - Y_j\right\|_2^2$$

همچنین تابع زیان، جفت گره‌ها را با توجه به شباهت آن‌ها در گراف اصلی (مثل A_{ij}) وزن‌دهی می‌کند [۲۱]:

(۷,۲)

$$\mathcal{L} = \frac{1}{2}\sum_{(v_i,v_j)} DEC\left(Y_i, Y_j\right).S_G\left(v_i, v_j\right)$$

در این روش هدف این است که وقتی دو گره در گراف اصلی شبیه هم هستند (برفرض مثال وقتی وزن A_{ij} زیاد است)، تعبیه دو گره را نزدیک‌هم نگه دارد [۱]. برای این منظور تابع زیان را به حداقل می‌رسانند.
بردار ویژه لاپلاسی از یک تابع زیان[4] درجه دو استفاده می‌کند. بنابراین تابع هدف بیش‌تر بر حفظ تفاوت بین گره‌ها تأکید می‌کند تا شباهتشان. این مسئله ممکن است موجب عدم حفظ ساختار محلی شود، چراکه اگر به

[1] Laplacian matrix
[2] Node transition probability matrix
[3] Katz similarity matrix
[4] Loss Function

تابع هدف توجه کنیم شباهت بین یال‌ها ($S_G\left(v_i, v_j\right)$) از درجه یک است ولی تفاوت بین یال‌ها از درجه دو می‌باشد ($\left\|Y_i - Y_j\right\|_2^2$).

۲-۳-۲-روش‌های تجزیه‌ماتریس مبتنی‌بر ضرب داخلی

به دنبال روش بردار ویژه لاپلاسی، اخیراً روش‌های تعبیه متعددی تولید شده‌است که در آن تابع رمزگشا از حاصل ضرب داخلی تعبیه‌ها بدست می‌آید:

$$DEC\left(Y_i, Y_j\right) = Y_i Y_j^T \quad (۸,۲)$$

به‌گونه‌ای که استحکام رابطه بین دو گره، متناسب با ضرب داخلی تعبیه‌های آن‌ها می‌باشد. الگوریتم‌های فاکتورگیری گراف (GF[1]) [۲۲]، $GraRep$ [۲۳] و $HOPE$ [۲۴] همه در این کلاس قرار می‌گیرند و هر سه روش از یک رمزگشا با ضرب داخلی استفاده می‌کنند، همچنین تابع زیان آن‌ها یک تابع خطای میانگین مربعات (MSE[2]) می‌باشد:

$$\mathcal{L} = \sum_{(v_i, v_j)} \left\|DEC\left(Y_i, Y_j\right) - S_G(v_i, v_j)\right\|_2^2 \quad (۹,۲)$$

در واقع تفاوت اصلی روش‌های یاد شده (GF، $GraRep$ و $HOPE$)، در معیار شباهت استفاده‌شده روی گراف اصلی می‌باشد ($S_G(v_i, v_j)$).

در روش **فاکتورگیری گراف** (GF)، معیار شباهت را ماتریس مجاورت تعریف می‌کنند:

(۱۰,۲)

$$S_G\left(v_i, v_j\right) = A_{ij}$$

روش $GraRep$ به منظور اخذ همسایگی‌های مرتبه بالاتر معیار شباهت دو گره در گراف اصلی را توان‌های بالاتر ماتریس مجاورت در نظر می‌گیرد، بر فرض مثال:

$$S_G\left(v_i, v_j\right) = A_{ij}^2 \quad (۱۱,۲)$$

[1] Graph Factorization
[2] Mean-Squared-Error

روش ***HOPE*** برای تعیین شباهت بین دو گره از معیارهای شباهتی همچون هم‌پوشانی‌همسایگی‌جاکارد[1] استفاده می‌کند. این روش بین «شباهت مرتبه اول»، که در آن S_G وجود یال مستقیم بین گره‌ها را درنظر می‌گیرد و «شباهت‌های مرتبه بالاتر» که S_G مفهوم کلی‌تری از همسایگی بین گره‌ها را برمی‌گرداند، تعادل برقرار می‌کند.
دلیل اینکه این روش‌ها به‌عنوان روش‌های فاکتورگیری ماتریس شناخته می‌شوند این است که درواقع ماتریس شباهت (A) را به دو ماتریس Y و Y^T تجزیه یا فاکتورگیری می‌کنیم. هدف از این روش‌ها صرفاً یادگیری تعبیه برای هر گره به‌گونه‌ای است که ضرب داخلی بین بردارهای تعبیه آموخته‌شده، برخی از معیارهای شباهت بین جفت گره‌ها در گراف اصلی را تقریب بزند.

۲-۴- روش‌های مبتنی‌بر قدم‌زنی تصادفی

بسیاری از روش‌های موفق اخیر تعبیه گره را از طریق اطلاعات به‌دست‌آمده از قدم‌زنی‌تصادفی یاد می‌گیرند. نوآوری اصلی این روش‌ها بهینه‌سازی تعبیه گره‌ها به‌گونه‌ای است که گره‌هایی که در یک قدم‌زنی‌تصادفی کوتاه برروی گراف، هم‌رخداد می‌شوند، تعبیه مشابهی داشته باشند (شکل ۲-۲). بنابراین، به‌جای استفاده از معیارهای شباهتی که قبلا بحث شد (منظور معیارهای شباهت در گراف اصلی است)، روش‌های قدم‌زنی‌تصادفی از یک معیار شباهت انعطاف‌پذیر و همچنین تصادفی استفاده می‌کنند [۶].
از قدم‌زنی‌تصادفی برای تقریب بسیاری از ویژگی‌های گراف از جمله مرکزیت گره[2] و شباهت استفاده شده‌است. حالتی را تصور کنید که گراف پاره‌ای مشاهده پذیر است، به این معنا که نمی‌توان به کل گره‌ها و یال‌های گراف دسترسی داشت یا حالتی که گراف مورد مطالعه بسیار بزرگ باشد، اینجاست که کارآمدی این روش‌ها مشخص می‌شود. روش‌های مختلفی برای تعبیه با استفاده از قدم‌زنی تصادفی بر روی گراف‌ها جهت به‌دست‌آوردن بازنمایی گره ارائه شده‌است که روش‌های $DeepWalk$ و $Node2Vec$ دو مثال بارز از این دسته هستند [۱۱] [۱۴].

۲-۴-۱- مدل *DeepWalk*

$DeepWalk$ یک روش دو مرحله‌ای است. این روش در مرحله اول، شبکه را با قدم‌زنی تصادفی طی می‌کند تا ساختارهای محلی را ازطریق روابط همسایگی استنباط کند. در مرحله دوم، از الگوریتمی به‌نام $SkipGram$ برای یادگیری و بهینه‌سازی تعبیه‌های ساختارهای به‌دست‌آمده از مرحله قبل استفاده می‌کند.
مرحله ۱- کشف ساختار محلی: هدف از این مرحله، کشف همسایگی‌های هر گره در شبکه است. برای انجام این کار، تعداد ثابت (k) قدم‌زنی تصادفی را که از هر گره به‌صورت مجزا شروع شده، ایجاد می‌شود. طول هر قدم‌زنی

[1] Jaccard Neighborhood Overlaps
[2] Node Centrality

(l) از قبل تعیین شده است. بنابراین، هنگامی که این مرحله به پایان رسید، k دنباله گره[1] به طول l به‌دست می‌آید. اصل تولید قدم‌زنی تصادفی این فرض است: گره های مجاور، مشابه هستند و باید تعبیه‌های مشابهی نیز داشته باشند. براساس این فرض، گره‌هایی را که هم‌زمان در یک مسیر وجود دارند[2] به‌عنوان گره‌های مشابه تعبیر و پذیرفته می‌شوند. این بدان معنی است که میزان فراوانی همزمانی دو گره در قدم‌زنی تصادفی، نشانگر شباهت آن دو گره است. درواقع این یک فرض معقول است چراکه یال‌ها، بیش‌تر نشان‌دهنده گره‌های مشابه یا گره‌های درحال تعامل در یک شبکه هستند. مرحله ۲- $SkipGram$: الگوریتم $SkipGram$ یک تکنیک محبوب برای یادگیری تعبیه کلمات است. باتوجه به کوله کلمات[3] و اندازه پنجره[4] هدف $SkipGram$ این است که شباهت تعبیه کلماتی را که در همان پنجره اتفاق می‌افتند، به حداکثر برساند. این ایده بر این فرض استوار است که: کلماتی که در یک پنجره وجود دارند، معمولاً معانی نزدیکی دارند. بنابراین تعبیه آن‌ها نیز باید نزدیک به یکدیگر باشد. برای تطبیق $SkipGram$ با شبکه‌ها، باید مفهوم پنجره (یا زمینه) مطابق با دنیای شبکه‌ها را تعیین کرد. اینجاست که قدم‌زنی‌های تصادفی وارد عمل می‌شوند.

هدف از مدل سازی زبانی، تخمین احتمال ظاهر شدن یک توالی خاص از کلمات، در یک کوله کلمات است. به‌طور رسمی‌تر، با توجه به دنباله‌ای از کلمات:

(۱۲,۲)

$$W_1^n = (w_0, w_1, \dots, w_n)$$

که در آن w_i کلمات موجود در کوله کلمات و W_1^n دنباله‌ای از این کلمات هستند، هدف، ماکزیمم کردن احتمال زیر (احتمال مشاهده کلمه nام با وجود مشاهده n کلمه قبلی) برای تمامی کلمات موجود کوله کلمات می‌باشد:

(۱۳,۲)

$$Pr\,(w_n | w_0, \dots, w_{n-1})$$

حال در این روش، یک تعمیم از مدل‌سازی زبانی، جهت اکتشاف گراف از طریق جریانی از قدم‌زنی‌های تصادفی کوتاه ارائه می‌شود. قدم‌زنی‌ها در یک شبکه را می‌توان معادل جملات و عبارات کوتاه در یک متن بلند درنظر گرفت. با این معادل‌سازی باید گفت در اینجا نیز هدف، این است که احتمال مشاهده رأس v_i را با توجه به تمام رئوس قبلی که تاکنون در مسیر قدم‌زنی تصادفی بازدید شده‌است، تخمین زد.

[1] Degree Sequence
[2] Co-Occurrence Nodes
[3] Bag of Words (BoW)
[4] Window Size or Context Size

(۱۴,۲)

$$Pr\ (v_i|(v_0, \dots, v_{i-1}))$$

اما ازآنجایی که هدف اصلی در اینجا یادگیری تعبیه‌هاست، بنابراین این معادله را باید به‌صورت زیر بازنویسی کرد:

(۱۵,۲)

$$Pr\ (v_i|(Y_0, \dots, Y_{i-1}))$$

و این یعنی احتمال مشاهده رأس v_i، به‌شرط مشاهده تعبیه تمام رئوس مشاهده‌شده قبلی در قدم‌زنی تصادفی. این هدف نیز دو اشکال اساسی دارد. اول اینکه درصورت طولانی شدن طول قدم‌زنی، قسمت شرطی معادله بالا بزرگ‌تر شده و عملاً محاسبه آن غیرممکن می‌شود. همچنین باتوجه به اینکه در فضای زبانی هم، کلمات قبل و بعد از یک کلمه خاص مهم هستند، با ایده گرفتن از آن در اینجا هم، همسایه‌های قبل و بعد از گره v_i را در محاسبات لحاظ می‌کنیم، بنابراین طول قدم‌زنی تصادفی به $2k+1$ افزایش می‌یابد. لذا برای حل این مشکل و رفع پیچیدگی زمانی یادشده، هدف جایگزین، ماکزیمم کردن تابع هدف زیر تعریف می‌شود:

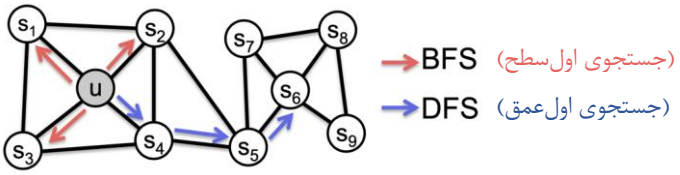
(۱۶,۲)

$$log\ Pr(v_{i-k}, \dots, v_{i-1}, v_{i+1}, \dots, v_{i+k}|Y_i)$$

برای یادگیری تعبیه‌ها از طریق $SkipGram$، بردارهای d-بُعدی، ابتدا به‌صورت تصادفی برای هر گره ایجاد می‌شود، سپس، مدل باتوجه به تابع هدف بالا، تعبیه‌ها را به گونه‌ای بهینه‌سازی می‌کند که احتمال مشاهده همسایه‌های یک گره (گره‌های هم‌رخداد در یک قدم‌زنی تصادفی) درصورت دیدار آن گره، افزایش یابد. درواقع این‌کار موجب می‌شود گره‌هایی که همسایگان مشابهی دارند، تعبیه‌های مشابهی نیز برای آن‌ها محاسبه شود.

۲-۴-۲- مدل $Node2Vec$

این مدل دراصل، ارتقاءیافته مدل $DeepWalk$ با انجام برخی تغییرات در بخش قدم‌زنی تصادفی آن می‌باشد [۱۴]. در روش $DeepWalk$، فاکتور طول ثابت (k) اجازه نمی‌داد که قدم‌زن گره‌های دورتر را مشاهده و کشف کند، فرض کنید دو گره شبیه به‌هم وجود داشته باشند اما فاصله آن‌ها بیش‌تر از k باشد، در این صورت $DeepWalk$ قادر به کشف این شباهت نخواهد بود. در واقع، تفاوت بسیار مهم $DeepWalk$ و این روش در این است



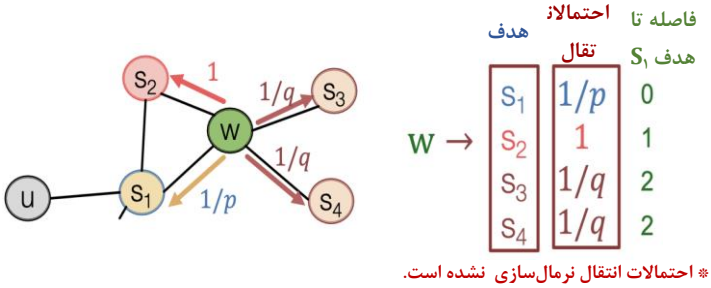
همسایه‌های کشف شده توسط جستجوی اول‌سطح: $\{S_1, S_2, S_3\} = N_{BFS}(u)$

همسایه‌های کشف شده توسط جستجوی اول‌عمق: $\{S_4, S_5, S_6\} = N_{DFS}(u)$

شکل۲- ۱. تفاوت بین قدم زنی‌های تصادفی براساس جستجوی اول‌سطح و جستجوی اول‌عمق [۶۳]

که این روش از قدم‌زنی تصادفی مغرضانه[1]استفاده می‌کند، طوری‌که بین جستجوی گراف اول‌سطح (BFS) و اول‌عمق (DFS[3]) یک تعادل و توازنی ایجاد کند، و از این رو تعبیه‌هایی با کیفیت بالاتر و حاوی اطلاعات مفیدتری نسبت به مدل سابق تولید می‌کند. داشتن رویکرد جستجوی اول‌عمق به ما این امکان را می‌دهد تا گره‌هایی با فاصله دورتر را کشف کنیم.

انتخاب تعادل مناسب بین رویکرد محلی (جستجوی اول‌سطح) و رویکرد سراسری (جستجوی اول‌عمق)، این روش را قادر می‌سازد تا ساختار اجتماع‌ها[4] و همچنین تعادل ساختاری بین گره‌ها را بهتر حفظ کند.



شکل۲- ۲. نمایی از چگونگی ایجاد تعادل در قدم‌زنی تصادفی با استفاده از پارامترهای p و q در روش $Node2Vec$. با فرض اینکه قدم زن از S_1 به W منتقل شده است ، برچسب یال‌ها، با احتمال انتخاب‌شدن آن گره به‌عنوان گره بعدی برای حرکت بعدی قدم‌زن متناسب هستند [۲۰].

به‌منظور حفظ تعادل بین جستجوی اول‌سطح و جستجوی اول‌عمق از دو پارامتر برای تولید و کشف همسایه‌ها استفاده می‌کنند: پارامتر p، که پارامتر برگشت به گره قبلی است و پارامتر q، نسبت جستجوی اول‌سطح به اول‌عمق می‌باشد. برای توضیح بیش‌تر و قابل لمس بودن این پارامترها در زیر شکلی آورده شده است تا با یک مثال ساده به توضیح این دو پارامتر بپردازیم. فرض کنیم که قدم زن[5] از S_1 حرکت کرده و رسیده به W و الان در W قرار دارد، همانطور که می‌بینیم دیگر احتمال‌های انتقال باهم برابر نیستند. بنابراین با احتمال ۱/p به گره قبلی (S_1) برمی‌گردد، با احتمال ۱/q گره‌های جدید را کشف می‌کند (S_3, S_4) و با احتمال ۱ به S_2 می‌رود، احتمال رفتن به S_2 برابر ۱ شده است، چراکه با رفتن به این گره، فاصله از گره مبدأ یعنی S_1 تغییری نمی‌کند. درضمن احتمالات نوشته شده نرمال‌سازی نشده‌اند[6] و باید تقسیم‌بر مجموع احتمالات شوند تا نرمال‌سازی انجام و مجموع احتمالات یک شود. حال پیداست که هرچه مقدار p کم‌تر باشد قدم‌زن تصادفی ما بیش‌تر به

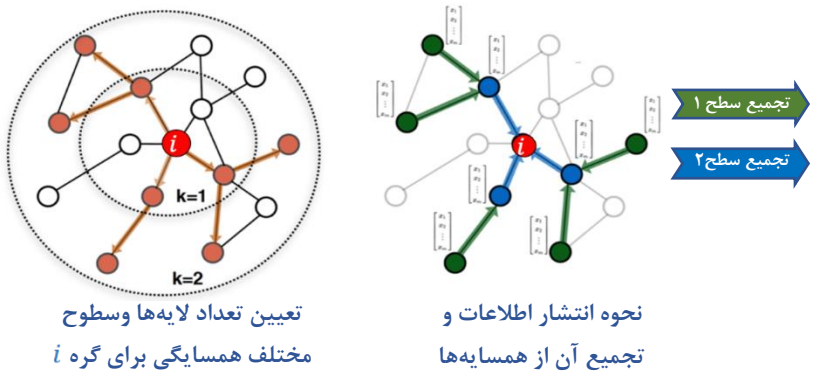
[1] Biased-Random Walks
[2] Breadth First Search
[3] Depth First Search
[4] Community
[5] Walker
[6] Unnormalized

جستجوی اول‌سطح متمایل خواهد بود، از طرفی هرچه مقدار q کم‌تر باشد تمایل به جستجوی اول‌عمق بیش‌تر خواهد بود.عمده اشکال هر دو روش معرفی‌شده، یعنی $DeepWalk$ و $Node2Vec$ این است که، تعبیه گره را به‌طور تصادفی برای آموزش مدل‌ها آغاز می‌کنند. از آنجایی که تابع هدف آن‌ها غیر محدب[1]است، این چنین شروع‌های تصادفی می‌تواند در بهینه‌های محلی[2]گیر کند. حال برای رفع این محدودیت‌ها مدل‌های نوین‌تری ارائه شدند که در ادامه به آن می‌پردازیم.

۵-۲- روش‌های مبتنی‌بر یادگیری‌عمیق

رشد تحقیقات برروی یادگیری‌عمیق باعث شده تا روش‌های مبتنی‌بر شبکه‌های عصبی عمیق بر گراف‌ها اعمال شوند [۹] [۱۰] [۲۵]. این روش‌ها، به‌دلیل توانایی آن‌ها در مدل‌سازی ساختار غیرخطی از داده‌ها، به منظور کاهش‌بُعد مورد استفاده قرار گرفته‌اند، ازطرفی روش‌های مبتنی‌بر قدم‌زنی‌تصادفی این کاستی را داشتند که ویژگی‌های خود گره‌ها تأثیر داده نمی‌شد، هر گره ممکن است برای خود ویژگی‌هایی داشته باشد که باید در فرآیند تعبیه دخیل شوند. لذا این موارد باعث شدند تا محققان به تعبیه‌های قدرمتندتری روی بیاورند.

[1] Non-Convex
[2] Local Optima

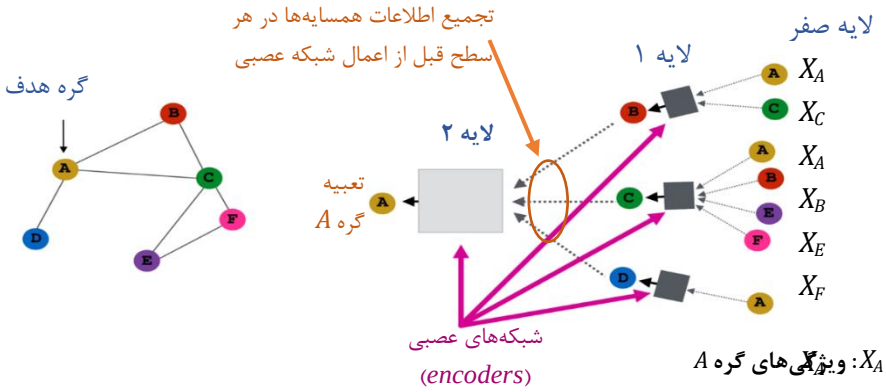


شکل۲- ۳. نحوه تجمیع پیام‌های رسیده از همسایه‌های گره i [۲۰]

۱-۵-۲- شبکه‌های کانولوشن گرافی[1](GCN)

شبکه‌های کانولوشن گرافی (GCN) [۱۹] با تعیین یک عملگرحلقوی[2] برروی گراف، به‌صورت تکراری تعبیه‌های همسایه‌ها را برای هر گره تجمیع[3] می‌کند و از یک تابع برای ترکیب تعبیه به‌دست‌آمده از همسایه‌ها و تعبیه خود آن گره که از تکرار قبلی موجود است، جهت ایجاد تعبیه جدید استفاده می‌کند. همانطور که در شکل نیز نشان داده‌شده است، مزیت این روش این است که هم، ساختار همسایگی گره‌ها را اخذ می‌کند و هم اینکه برای تعبیه یک گره، اطلاعات خود آن گره را نیز درنظر می‌گیرد. معمولاً در این روش، در انتخاب تعداد لایه‌ها (عمق) از ۶ لایه بالاتر نمی‌روند، چراکه هرچه عمیق‌تر شویم پیام‌های رسیده از همسایه‌ها اهمیتشان برای ما بیش‌تر می‌شود، درواقع باید ببینیم تا چه عمقی همسایه‌ها برای ما مهم هستند. ازطرفی، باتوجه به اینکه شبکه‌های دنیای

[1] Graph Convolutional Networks
[2] Convolutional operator
[3] Aggregate

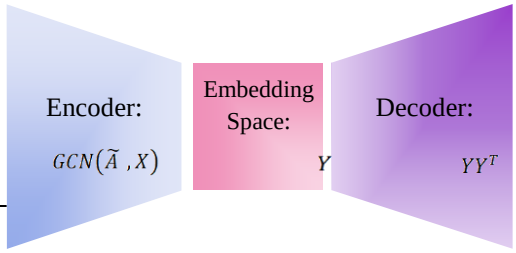


شکل۲- ۴. نمایی از نحوه رمزگذاری و تجمیع اطلاعات همسایه‌ها. جهت تولید تعبیه برای گره A، این‌مدل اطلاعات همسایه‌های محلی A (یعنیB ، C و D) را جمع‌آوری می‌کند، سپس اطلاعات دریافت‌شده از این همسایگان را به همراه اطلاعات موجود در خود A تجمیع و به رمزگذار تحویل می‌دهد. در این مثال حالت دولایه از این مدل نشان داده‌شده‌است (یعنی اطلاعات همسایگی تا دو گام دورتر از گره مبدأ جمع‌آوری می‌شوند) ، اما در اصل این روش می‌تواند از یک عمق دلخواه برخوردار باشند. در "عمق" یا "لایه" صفر اطلاعات اولیه همان ویژگی‌های خود گره‌ها هستند [۱۴].

واقعی دارای میانگین فاصله[1]و یا قطر شبکه[2]کوچک‌تر از ۶ هستند[3] بنابراین اگر تا این عمق پیش برویم تقریباً کل گره‌های موجود در شبکه را مشاهده کرده‌ایم و دیگر نیازی نیست بیش‌تر از این بر عمق بی‌افزاییم (معمولاً ۲ تا ۴ سطح بسته به نوع کاربرد برای ما کافی است). همچنین در این مدل از تابع زیان اختلاف آنتروپی دودویی جهت بهینه‌سازی مدل[4]استفاده می‌شود.

۲-۵-۲- مدل Graph Auto Encoder (GAE)

Autoencoder یک شبکه عصبی است که از یک رمزگذار و رمزگشا تشکیل شده‌است. این مدل چارچوبی را برای آموزش روی گراف‌ها به‌صورت بدون‌ناظر[5]ارائه می‌کند. رمزگذار داده‌های ورودی را به یک چکیده تبدیل می‌کند (داده‌های ورودی را فشرده‌سازی کرده و به یک فضایی با بعد کم‌تر می‌برد)، این در حالی است که رمزگشا داده‌های خروجی رمزگذار را گرفته و سعی در بازسازی آن داده‌ها دارد. درواقع *Graph Auto Encoder*



شکل۲- ۵. چارچوب کلی روش *GAE*

[1] Average Distance
[2] Network Diameter
[3] Six-Degrees of Separation Phenomena
[4] Model Optimization
[5] Unsupervised Learning

یکی از روش‌های تعبیه گراف است و بر پایه شبکه‌های عصبی طراحی‌شده که داده‌های ورودی آن، اطلاعات یک گراف می‌باشد [۲۶]. این روش نیز به‌دلیل پیچیدگی زمانی کم و توانایی بسیار خوب در کاهش‌بُعد، توجه بسیاری را در حوزه تعبیه‌گراف به‌خود جلب‌کرده‌است.

تابع رمزگذار در این مدل یک شبکه عصبی است که عمدتاً از روش تعبیه GCN در آن استفاده می‌شود. این تعبیه‌کننده (GCN) برای اینکه بتواند تعبیه گره‌ها را تولید کند به دو ماتریس نیاز دارد. (الف) ماتریس همبندی(A) گراف، (ب) ماتریس ویژگی‌گره‌ها (X). بنابراین خواهیم داشت:

$$Y = GCN(\tilde{A}, X) = ReLU(\tilde{A}\, XW_0) \quad (۱۷,۲)$$

$$A \leftarrow A + I, \quad \tilde{A} = D^{-1/2} A\, D^{-1/2} \quad (۱۸,۲)$$

که در آن D ماتریس قطری تشکیل‌یافته از درجات گره‌ها، I ماتریس همانی، Y خروجی تابع رمزگذار (GCN) و همان ماتریس گره‌های تعبیه‌شده در فضای جدید می‌باشد. برای تولید این تعبیه‌ها، GCN ماتریس همبندی و ماتریس ویژگی گره‌ها را به‌عنوان ورودی گرفته، ابتدا یک فرم نرمالایز شده از ماتریس‌همبندی ($\tilde{A}$) را ایجاد کرده و سپس با ضرب ماتریس حاصله در ماتریس ویژگی و همچنین ماتریس وزنی(ماتریس ضرایب لایه‌های GCN) و در آخر با اعمال یک تابع فعال‌ساز[1]($ReLU$)، ماتریس تعبیه گره‌ها را به‌دست می‌آورد. این GCN می‌تواند از چندین لایه تشکیل شود، به‌طور مثال در [۲۶] نویسنده از دو لایه GCN به عنوان رمزگذار استفاده کرده‌است. درصورت استفاده از دولایه GCN خواهیم داشت:

$$Y^{l2} = GCN(\tilde{A}, Y^{l1}) = ReLU\left(\tilde{A}\, Y^{l1} W_1\right) = \tilde{A}\; ReLU\left(\tilde{A}\, XW_0\right) W_1$$

Y^{l1} تعبیه‌های تولیدشده توسط لایه اول وY^{l2} تعبیه‌های تولیدشده توسط لایه دوم می‌باشد، همچنین Y^{l1}(تعبیه‌های لایه اول GCN) همان ماتریس ویژگی‌گره‌ها و ماتریس‌های W_0 و W_1 ماتریس ضرایبی است که مدل در هر لایه سعی در آموزش آن دارد.

[1] Activation Function

تابع رمزگشا: حال که تعبیه گره‌ها به‌دست آمد، مدل توسط تابع رمزگشا سعی در بازسازی[1]این تعبیه‌ها دارد به‌گونه‌ای‌که پس از بازسازی، ماتریس حاصله به گراف اصلی هرچه نزدیک‌تر و شبیه‌تر باشد. درواقع مدل تلاش دارد خطای بازسازی[2]را به حداقل برساند. یکی از متداول‌ترین روش‌ها برای بازسازی (تابع رمزگشا) استفاده از ضرب داخلی می‌باشد، به این ترتیب تابع رمزگشا با ضرب داخلی تعبیه گره‌های تولیدشده توسط رمزگذار، سپس با اعمال یک تابع فعال‌ساز مثل سیگموید[3]احتمال رخداد یال بین هر جفت گره را بازسازی می‌کند.

مدل با مقایسه احتمال یال‌های بدست‌آمده از تابع رمزگشا و سنجش صحیح یا ناصحیح بودن این پیش‌بینی‌ها با یال‌های واقعی موجود در گراف، تابع زیان را به روزرسانی کرده و سعی در بهبود تعبیه‌های تولیدشده و در نتیجه بهبود پیش‌بینی‌ها خواهدداشت. برای این منظور از تابع زیان اختلاف آنتروپی دودویی لاجستیک[4] استفاده می‌شود [۲۷].

$$\mathcal{L} = Average\{l_1, \dots, l_N\} \quad (۲۰,۲)$$

$$l_n = -w_n\,[y_n.log\sigma(x_n) + (1-y_n).log(1-\sigma(x_n))] \quad (۲۱,۲)$$

در معادله بالا، N تعداد یال‌های موجود در فاز یادگیری‌است که در اختیار مدل قرار گرفته، w_n ماتریس ضرایبی‌است که مدل سعی در یادگیری آن دارد، y_n برچسب واقعی یال‌هاست که درصورت وجود در گراف مقدار ۱ و در غیر این‌صورت مقدار ۰ می‌گیرد، x_n مقدار پیش‌بینی‌شده توسط مدل برای آن یال و σ تابع فعال‌ساز سیگموید می‌باشد، باضرب سیگموید، آن مقدار پیش‌بینی‌شده (خروجی تابع رمزگشا) تبدیل به یک احتمال می‌شود که در بازه صفر و یک قرار دارد. هرچه مقدار احتمال پیش‌بینی‌شده برای هریال به ماتریس‌همبندی گراف اصلی نزدیک‌تر باشد، مقدار این تابع زیان کوچک‌تر شده و درنهایت منجربه پیش‌بینی‌های باکیفیت‌تری خواهدشد. همچنین نویسنده پیشنهاد می‌کند، درصورتی که ماتریس ویژگی‌گره‌ها دردسترس نباشد، با جایگزینی آن با ماتریس مشخصه[5](ماتریس قطری که همه درایه‌های قطر اصلی آن یک می‌باشد) وابستگی مدل به ماتریس ویژگی از بین می‌رود [۲۶].

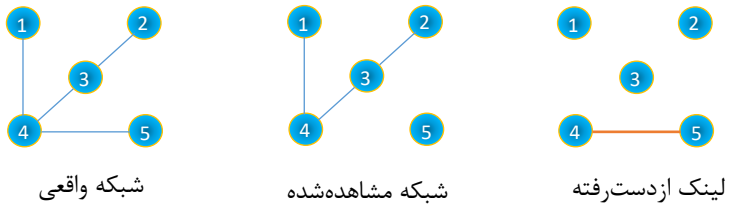
[1] Reconstruction
[2] Reconstruction Error
[3] Sigmoid Function (Logistic Function)
[4] Binary Cross Entropy with Logits Loss
[5] Identity Matrix

۲-۶- مسائل پیش‌بینی یال به‌عنوان یکی از کاربردهای تعبیه

حال که به بررسی برخی از مهم‌ترین و پرکاربردترین روش‌های تعبیه گراف پرداختیم، به دلیل استفاده از پیش‌بینی یال به‌عنوان یکی از کاربردهای تعبیه گراف، جهت سنجش و ارزیابی تعبیه پیشنهادی در فصول آتی، بهتر است نگاهی به مسائل و روش‌های پیش‌بینی یال غیرتعبیه‌ای هم بی‌اندازیم و برخی کارهای انجام‌شده اخیر در این حوزه را مرور کنیم.

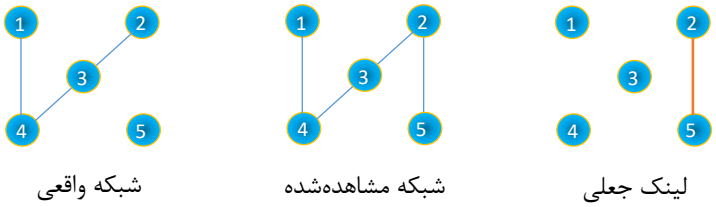
۲-۶-۱- پیش‌بینی یال‌های ازدست‌رفته[1] و جعلی[2]

معمولاً داده‌های شبکه‌های دنیای واقعی کامل نیستند و با برخی خطاها یا نواقصی همراه هستند. به‌طور مثال در بسیاری از موارد دیده می‌شود که افراد به‌دلیل برخی نگرانی‌های امنیتی یا به‌دلیل حفظ حریم‌خصوصی، دسترسی به صفحات خود در شبکه‌های‌اجتماعی را محدود می‌کنند. مسائل این‌چنینی باعث به وجود آمدن



شکل۲- ۶. نمایی از لینک‌های ازدست‌رفته

برخی نواقص در دادگان شبکه‌های موردمطالعه می‌شود و کار تحلیل و ارزیابی را با مشکل مواجه می‌کند [۲۸]. برای حل چنین مشکلی می‌توان از روش‌های پیش‌بینی یال کمک گرفت و یال‌های از دست‌رفته را بازیابی کرد. به‌طور کلی یال‌های گمشده یا ازدست‌رفته به یال‌هایی گفته می‌شود که در شبکه واقعی حضور داشته اما به هر دلیلی در داده‌های مشاهده‌شده ما وجود ندارند. در شکل زیر نمونه‌ای از این مسئله نشان داده‌شده‌است.



شکل۲- ۷. نمایی از لینک‌های جعلی

[1] Missed Links
[2] Spurious Links

یال‌های جعلی نیز در نقطه مقابل یال‌های ازدست‌رفته قرار دارند و به آن دسته از یال‌هایی گفته می‌شوند که در گراف اصلی وجود ندارند ولی به اشتباه در گراف مشاهده شده حضور داشته‌اند. شکل زیر این موضوع را نشان می‌دهد [۲۹].

۲-۶-۲- پیش‌بینی یال در شبکه‌های مختلف

ساختار و نوع اطلاعات موجود در شبکه‌های اطلاعاتی موجب به‌وجودآمدن شبکه‌های مختلفی شده، شبکه‌هایی همچون ناهمگن[۱] چندلایه و دوبخشی[۲]نمونه‌ای از انواع شبکه‌ها می‌باشند. هرکدام از این شبکه‌ها ویژگی‌ها و مشخصه‌های خاص خود را دارند برای مثال در برخی از آن‌ها گره‌ها در همه لایه‌ها یکسان بوده و در برخی دیگر نه‌تنها گره‌ها بلکه نوع ارتباطات بین آن‌ها نیز متفاوت است. لذا هنگام ارائه مدل پیش‌بینی یال، حال چه با روش تعبیه و چه روش‌های مبتنی بر شباهت و یا روش‌های مبتی بر قدم‌زنی‌تصادفی، لازم است که نوع شبکه را مورد توجه قرار داده و باتوجه به ویژگی‌های شبکه مورد بررسی روشی مناسب را اتخاذ کرد. در زیر به معرفی اجمالی این شبکه‌ها می‌پردازیم [۳۰].

- **پیش‌بینی یال در شبکه‌های ناهمگن[۳]**

پیش‌بینی یال در این شبکه‌ها با توجه به عدم وجود یال و گره‌های یکسان و تعدد انواع آن‌ها در این شبکه‌ها ویژگی‌های متفاوتی را می‌طلبد چرا که انواع مختلف یال بین انواع مختلف گره به‌طور یکنواخت و با رویکرد یکسانی شکل نخواهد گرفت، برای این منظور محققان روش‌ها و راهکارهای متفاوتی را پیشنهاد کرده‌اند. برخی از آن‌ها شبکه را بر اساس یال‌ها تقسیم‌بندی کرده و شبکه‌های مختلفی از روی آن ایجاد می‌کنند سپس پیش‌بینی یال را روی هر کدام از این زیرشبکه‌های ایجاد شده انجام داده و اطلاعات به‌دست‌آمده از هر کدام را تلفیق می‌کنند [۳۱].

- **پیش‌بینی یال در شبکه‌های دوبخشی[۴]**

نوع دیگری از شبکه‌ها وجود دارند که به‌عنوان شبکه‌های دوبخشی شناخته می‌شوند، شبکه‌های کاربر-محصول[۵] نمونه‌ای از این شبکه‌ها می‌باشند. برخی ویژگی‌ها همچون عدم شکل‌گیری مثلث‌ها[۶]و زیرگراف‌های کامل (کلیک)[۷] بر روی این شبکه‌ها موجب متمایز شدن روش‌های پیش‌بینی یال آن با شبکه‌هایی همچون تک‌بخشی[۱]

[1] Heterogeneous Networks
[2] Bipartite
[3] Heterogeneous Networks
[4] Bipartite
[5] User-Product
[6] Triangles
[7] Clique

می‌شود [۳۲]. برخی پژوهشگران برای حل مشکل، روش‌های کلاسیک پیش‌بینی یال همچون همسایگان مشترک[2]را به این شبکه‌ها تعمیم داده و یا برخی دیگر این گراف ها را به گراف‌های تک‌بخشی تصویر کرده‌اند [۳۳] [۳۴].

- **پیش‌بینی یال در شبکه‌های چندلایه**

یکی از مهم‌ترین و بارزترین تفاوت‌های شبکه‌های چندلایه با شبکه‌های ناهمگن عدم وجود گره‌های متفاوت در این شبکه هاست و همانطور که پیش‌تر نیز اشاره‌شد، نوع گره‌ها در این شبکه‌ها یکسان می‌باشد و این تفاوت نوع یال‌ها و ارتباطات بین این گره‌هاست که باعث ایجاد لایه‌های متفاوت می‌شود. هرویتس و همکاران [۳۵] شبکه توییتر-فوراسکوئر را بررسی‌کرده و تحلیل‌هایی در رابطه با همبستگی بین فعالیت‌های کاربران متناظر در هر دولایه و هم‌پوشانی آن‌ها ارائه می‌دهند. آن‌ها به این نتیجه می‌رسند که کاربرانی که در هردو شبکه باهمدیگر ارتباط دارند (متصل هستند) نسبت به گره‌هایی که فقط در یکی از شبکه‌ها به‌هم متصل‌اند تعامل بیش‌تری داشته و فعال‌ترند. برخی پژوهشگران دیگر، آنتروپی را یکی از ویژگی‌های قابل تعمیم به شبکه‌های چندلایه می‌دانند و آن را با ویژگی‌های دیگر درون‌لایه‌ای (همچون تعداد همسایگان مشترک، ضریب جاکارد و آدامیک آدار) ترکیب کرده

و ترکیب حاصل را به‌عنوان یک ویژگی جدید شبکه‌های چندلایه معرفی می‌کنند [۳۶].در پژوهش [۳۷] جلیلی و همکاران از ویژگی‌های مبتنی بر فرامسیر[3]با فرامسیرهایی به طول حداکثر ۴ به‌عنوان ویژگی‌های بین‌لایه‌ای به پیش‌بینی یال در هر لایه می‌پردازند، همچنین آن‌ها در هرلایه دسته‌بندهای مختلف ازجمله ماشین‌بردار ویژه[5]و k-نزدیک‌ترین همسایه[6]استفاده می‌کنند که درنتیجه با همین روش می‌توانند به بهترین نتیجه دست پیدا کنند.

۳-۶-۲-دسته‌بندی کلی روش‌های پیش‌بینی یال

روش‌های پیش‌بینی یال بسیاری تا به‌حال ارائه شده که می‌توان آن‌ها را به دو دسته کلی تقسیم‌بندی کرد:
(آ) رویکردهای مبتنی‌بر شباهت (ب) رویکردهای مبتنی‌بر یادگیری

---

[1] Unipartite (Monopartite)
[2] Common Neighbor
[3] Meta-Path
[4] Classifier
[5] Support Vector Machine (SVM)
[6] K-Nearest Neighbor (KNN)

- **رویکردهای مبتنی‌بر شباهت**

ویژگی‌های مختلف بسیاری مبتنی‌بر شباهت تعریف شده‌اند که این رویکردها با محاسبه این ویژگی‌ها به پیش‌بینی یال می‌پردازند. ویژگی‌های مبتنی‌بر شباهت خود به چند دسته مختلف از جمله ویژگی‌های مبتنی بر همسایگی گره (شامل همسایگان مشترک، ضریب جاکارد، معیار آدامیک آدار، اتصال ترجیحی)، ویژگی‌های مبتنی‌بر مسیر (شامل فاصله کوتاه‌ترین مسیر و معیار کاتز) همچنین ویژگی‌های مبتنی‌بر قدم‌زنی‌تصادفی (شامل زمان اصابت[1] زمان سفر[2] و رتبه‌بندی صفحه‌اصلی[3]) تقسیم‌بندی می‌شوند. رویکردهای مبتنی‌بر شباهت با محاسبه یک یا چندین معیار از معیارهای معرفی‌شده، نمراتی را به هر یال گراف تخصیص داده و سپس این نمرات را مرتب کرده و آن‌ها را رتبه‌بندی می‌کنند، جفت‌گره‌هایی که نمرات شباهت بالاتری اخذ کرده‌اند به احتمال بیش‌تری باهم در ارتباط خواهند بود و لینک بینشان برقرار است.

- **رویکردهای مبتنی‌بر یادگیری**

نقطه مقابل رویکردهای مبتنی بر شباهت، رویکردهای مبتنی‌بر یادگیری هستند که هرکدام از معیارهای معرفی‌شده در بخش قبلی و یا معیارهای جدید همگی به عنوان یک ویژگی در اختیار یک مدل یادگیری ماشینی قرار می‌گیرند (مانند آنچه که در تعبیه گراف‌ها شاهد آن هستیم) و سپس مدل نمره مثبت را به یال‌های واقعی (موجود در گراف) و نمره منفی را به یال‌های غایب اطلاق می‌کند و در ادامه با یادگیری و بهینه‌شدن مدل به پیش‌بینی یال‌های از دست‌رفته می‌پردازد [۳۸].

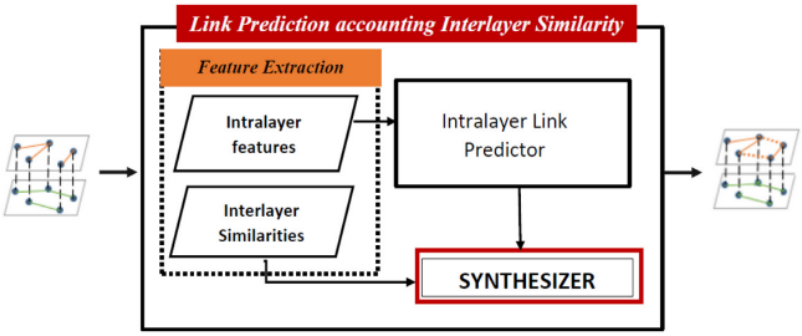
نکته قابل توجه اینکه هیچ کدام از روش‌های یاد شده را نمی‌توان به طور قطع برتر بر دیگری دانست و هرکدام از روش‌ها در کاربردهای مختلف و یا بر روی شبکه‌های مختلف عملکرد متفاوتی از خود نشان خواهند داد، اما با این حال، رویکردهای مبتنی بر یادگیری اخیراً مورد توجه بسیاری از محققان قرار گرفته و در مجموع پیش‌بینی‌های دقیق‌تر و دقت‌های نسبتاً بالاتری را به نسبت سایر رویکردها ارائه می‌کنند.

در ادامه به معرفی چندین روش پیش‌بینی یال می‌پردازیم که برخلاف روش‌های قبلی که همگی برپایه یادگیری بودند، این روش‌ها مبتنی‌بر شباهت هستند که روش‌های جدیدی محسوب می‌شوند و به دقت‌های نسبتاً خوبی در پیش‌بینی یال دست یافته‌اند.

[1] Hitting Time
[2] Commute Time
[3] Rooted PageRank

۲-۷- مدل *LPIS*[1]

این مدل از جمله روش‌های مبتنی‌بر شباهت است که با درنظرگرفتن اطلاعات درون‌لایه‌ای و بین‌لایه‌ای اقدام به پیش‌بینی یال در شبکه‌های چندلایه می‌کند. چارچوب کلی این روش در شکل زیر آمده‌است.



شکل۲- ۸. چارچوب کلی مدل *LPIS* [۳۰]

در این روش ابتدا ویژگی‌های درون لایه‌ای توسط معیارهای مبتنی بر شباهت اشاره‌شده در بخش قبل (همچون همسایگان مشترک، ضریب جاکارد و ...) محاسبه شده و توسط دسته‌بند لاجستیک‌رگرسیون[2] یک احتمال تشکیل یال برای هرجفت گره درون هرلایه اختصاص می‌یابد. سپس با استفاده از معیارهای شباهت بین‌لایه‌ای (همچون همبستگی درجه به درجه[3]) شباهت دولایه محاسبه می‌شود، حال با در دست داشتن شباهت بین لایه‌ای و همچنین

احتمال تشکیل یال درون‌لایه‌ای، از یک تابع ترکیب‌کننده برای پیش‌بینی احتمال تشکیل یال در لایه مقصد استفاده می‌شود. توجیه استفاده از این تابع این است که اگر به‌عنوان مثال یالی بین گره i و گره j در لایه ۱ وجود داشت و در صورت شبیه بودن لایه ۱ به لایه ۲، احتمال اینکه همان یال در لایه ۲ نیز تکرار شود وجود دارد. مشخص است که هرچه شباهت بین دولایه و احتمال تشکیل یال در لایه اول بالاتر باشد احتمال تشکیل همان یال در لایه بعدی هم بیش‌تر می‌شود [۳۰].

۲-۸- مدل *EMLP*

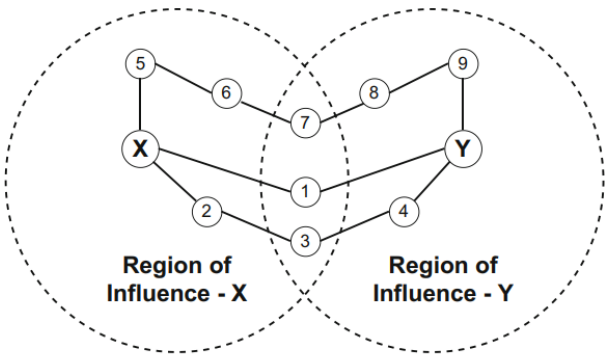
در این پژوهش، یک روش مبتنی‌بر شباهت جدید برای پرداختن به مسئله پیش‌بینی یال در شبکه‌های چندلایه با استفاده از نظریه شواهد[4] ارائه شده‌است. نظریه شواهد، می‌تواند به طور دقیق اطلاعات نامشخص و غیرقطعی را

[1] Link Prediction Accounting Interlayer Similarity
[2] Logistic Regression Classifier
[3] Degree-Degree Correlation (DDC)
[4] Evidence Theory

از منابع مختلف اخذ کرده و آن‌ها را برای تصمیم‌گیری مناسب ترکیب کند. هدف از این کار استفاده از اطلاعات موجود در لایه‌های دیگر برای پیش‌بینی یال در لایه مقصد است. برای پیش‌بینی پیوندها در لایه هدف، روش پیشنهادی اطلاعات همه لایه‌ها را ترکیب می‌کند. بنابراین، هر لایه یک منبع شواهد درنظر گرفته می‌شود. برای پیوندهای بالقوه در لایه هدف، نمرات شباهت آن‌ها که از لایه‌های مختلف به دست می‌آید به‌عنوان شواهد متفاوت که از منابع متفاوت به‌دست‌آمده‌اند تلقی می‌شود. درواقع نظریه شواهد شباهت‌های به‌دست‌آمده از هر لایه را به‌عنوان یک منبع شواهد در نظر گرفته و با ترکیب این منابع (ترکیب اطلاعات به‌دست‌آمده از لایه‌های مختلف) سعی در پیش‌بینی پیوند در لایه مقصد دارد. همچنین در این مدل، برای محاسبه شباهت یک جفت‌گره درون هر لایه، یک معیار شباهت جدید به‌نام ECM[1] با الهام از مدل هدایت‌الکترونیکی موجود در مدارهای الکترونیکی استفاده می‌شود. در حوزه الکترونیک، رسانایی الکتریکی، که رسانایی یک قطعه رسانا را اندازه‌گیری می‌کند، به‌عنوان معکوس مقاومت آن تعریف می‌شود. هر چه رسانایی آن قطعه بیش‌تر باشد، عبور جریان از آن آسان‌تر خواهدبود. لذا با الهام از این پدیده فیزیکی، یک شاخص برای محاسبه شباهت دو گره درون هر لایه پیشنهاد شده‌است. برای این منظور، یک گره در شبکه را به عنوان رسانا و انتقال‌دهنده جریان، همچنین درجه آن گره به‌عنوان مقاومت آن در نظر گرفته شده‌است. حال با این تعریف، معکوس درجه هر گره نشانگر رسانایی آن خواهدبود. در این سناریو می‌توان شباهت دو گره را به‌عنوان مجموع مقادیر رسانایی گره‌ها در تمام مسیرهای منتهی به آن‌ها در شبکه تعریف کرد. با این حال، از آنجایی که پیمودن تمام مسیرها بین دو گره زمان‌بر است، در این پژوهش فقط مسیرهایی به طول ۲ و ۳ در نظر گرفته شده‌است. همچنین برای اندازه‌گیری شباهت بین لایه‌ها از ویژگی‌های شباهت بین‌لایه‌ای همچون ضریب خوشه‌بندی استفاده شده‌است [۳۹].

۹-۲- مدل $HOPLP$

هدف از این روش هم همچون روش پیشین استفاده از لایه‌های دیگر جهت پیش‌بینی یال در لایه مقصد است. در این پژوهش از یک مدل تجمیع استفاده می‌شود، این مدل تجمیع اطلاعات لایه‌های مختلف را در یک شبکه واحد وزن‌دار خلاصه می‌کند. انگیزه اصلی در پس این روش، استفاده از مسیرهای مرتبه بالاتر برای تخمین بهتر شباهت‌های همسایگی است. با توجه به ویژگی «سه درجه اثرگذاری»[2] اشاره شده در مقاله [۴۰] می‌توان نتیجه گرفت که هر گره می‌تواند تا همسایه‌های درجه سه خود (همسایه‌هایی با طول ۳) اثربخش باشد.



شکل۲- ۹. ناحیه‌های اثربخشی گره‌های X و Y [۵۵]

[1] Electronic Conductance Me
[2] Three Degrees of Influence

همانطور که در شکل بالا آمده به وضوح قابل مشاهده است که اگر ناحیه اثرگذاری گره‌ها را به فاصله یک گام[1]از گره مبدأ فرض کنیم، تنها مسیر مرتبط بین گره‌های X و Y مسیر $X-1-Y$ خواهد بود. اگر فرض کنیم که ناحیه اثرگذاری یک گره از نقطه مبدأ، به دو گام افزایش یابد، مسیرهای مربوطه برای پیش‌بینی پیوند $X-1-Y$ و $X-2-3-4-Y$ خواهند بود. در نهایت با استفاده از پدیده ۳ درجه اثرگذاری، می‌توانیم فرض کنیم که ناحیه نفوذ یک گره تا ۳ گام دورتر از آن گسترش می‌یابد. با استفاده از این فرض، می‌توانیم ببینیم که برای محاسبه احتمال پیوند بین X وY، باید یک فرامسیر متشکل از تمامی مسیرهای ممکن اعم از مسیرهای کوتاه‌تر ($X-1-Y$ و $X-2-3-4-Y$) و همچنین مسیرهای طولانی‌تر (یعنی $X-5-6-7-8-9-Y$) را برای پیش‌بینی دقیق‌تر یال استفاده کنیم.

درواقع با این رویکرد به جای همسایگی‌های محلی (همسایگان اصلی هر گره) همسایگان سراسری آن نیز پوشش داده می‌شود که به عقیده نویسندگان این مقاله، رویکرد مناسب‌تری است. این روند محسابه احتمال پیوند، بسیار شبیه‌به روش محاسبه شباهت مبتنی‌بر تخصیص‌منابع[2] می‌باشد با این تفاوت که روش تخصیص‌منابع تنها مسیرهای با طول دو را در نظر می‌گیرد در حالی که در این روش‌های مسیرهایی با طول بیش‌تر نیز مورد توجه است.

۱۰-۲-جمع‌بندی

در جدول مروری که در ادامه آمده ، مروری کلی بر پیشینه مسائل تعبیه گراف و پیش‌بینی یال داریم.

[1] Hop
[2] Resource Allocation (RA)

جدول ۲-۱. دسته‌بندی انواع روش‌های تعبیه گراف و پیش‌بینی‌یال

دسته بندی		سال	مدل	پیچیدگی زمانی	ویژگی‌ها
روش‌های تعبیه گراف	تجزیه‌ماتریس *Matrix Factorization*	۲۰۰۱	*Laplacian Eigenmaps*	$O(\|E\|d^2)$	جزو اولین روش‌های تجزیه گراف هستند و به‌دلیل استفاده از ماتریس همبندی به‌عنوان شباهت بین گره‌ها، شباهت‌های مرتبه بالاتر بین گره‌ها حفظ نمی‌شوند (گره‌های دورتر از همسایگی اول، تأثیری در محاسبه تعبیه‌ها ندارند). تعبیه گره‌ها از طریق فاکتورگیری ماتریس همبندی گراف حاصل می‌شود، به‌گونه‌ای که گره‌های متصل‌به‌هم، تعبیه مشابهی داشته باشند.
		۲۰۱۳	*Graph Factorization*	$O(\|E\|d)$	
		۲۰۱۵	*GraRep*	$O(\|V\|^3)$	
		۲۰۱۶	*HOPE*	$O(\|E\|d^2)$	
	قدم‌زنی تصادفی *Random Walk*	۲۰۱۴	*DeepWalk*	$O(\|V\|d)$	به‌جای استفاده از یک معیار ثابت جهت اندازه‌گیری شباهت بین گره‌ها (همچون ماتریس همبندی که در روش‌های تجزیه گراف استفاده می‌شد) از قدم‌زنی تصادفی برای حصول این شباهت استفاده می‌شود. این موجب حفظ شباهت‌های مرتبه‌بالاتر بین گره‌ها و افزایش دقت مدل در پیش‌بینی‌ها می‌شود. گره‌هایی که در یک قدم‌زنی تصادفی کوتاه، هم‌رخداد بوده‌اند، تعبیه‌های مشابهی خواهند داشت.
		۲۰۱۶	*Node2Vec*	$O(\|V\|d)$	
	یادگیری‌عمیق *Deep Learning*	۲۰۱۶	*GCN*	$O(\|E\|d^2)$	برخلاف روش‌های قبلی، اطلاعات و ویژگی خود گره‌ها نیز در محاسبه تعبیه آن‌ها تأثیرگذار است، پیچیدگی زمانی بسیار پایینی دارند و دقت پیش‌بینی مناسبی ارائه می‌دهند. برای شبکه‌های تک‌لایه طراحی شده‌اند و درصورت دردسترس نبودن ویژگی گره‌ها دقت پیش‌بینی آن‌ها کاهش می‌یابد، همچنین دارای طراحی استنتاجی هستند، به‌گونه‌ای که تعبیه گره‌های مشاهده‌نشده را می‌توان ازروی پارامترهای آموخته‌شده بدون نیاز به بازمحاسبه تعبیه کل گره‌ها به دست آورد.
		۲۰۱۷	*GAE*	$O(\|E\|d^2)$	
روش‌های پیش‌بینی یال	مبتنی‌بر شباهت (همسایگی گره)	۲۰۱۹	*LPIS*	$O(\|E\|\|V\|L)$	روش‌های نوین پیش‌بینی یال در شبکه‌های چندلایه، غیرتعبیه‌ای و غیر یادگیرنده هستند که از شباهت‌های درون‌لایه‌ای و بین‌لایه‌ای از پیش تعریف‌شده توسط کاربر، جهت اندازه‌گیری شباهت بین دوگره و درنتیجه احتمال وجود یال بین آن‌ها استفاده می‌کنند. به‌جهت اینکه برای شبکه‌های چندلایه طراحی شده‌اند، دقت‌های خوبی ارائه می‌دهند، ولی دارای مشکل مقیاس‌پذیری به‌دلیل پیچیدگی زمانی نسبتاً بالا می‌باشند.
		۲۰۲۲	*EMLP*	$O(\|V\|^2+L\|V\|)$	
	مبتنی‌بر شباهت (مسیر)	۲۰۲۲	*HOPLP*	$O(\|V\|^2\langle k\rangle)$	

۳- برنامه آتی

نحوه اجرا

بطور مشخص برای ادامه کار، برنامه زیر در نظر گرفته شده‌است:

- در قدم اول به مطالعه و بررسی روشهای ارائه شده مرتبط با موضوع طرح می‌پردازیم.
- پس از بررسی مدل‌های موجود، استخراج ایده اولیه و تحلیل نظری آن در دستور کار قرار می‌گیرد.
- در ادامه به آماده‌سازی و پیاده‌سازی روش پیشنهادی خواهیم پرداخت.
- پس از پیاده‌سازی روش پیشنهادی به ارزیابی آن و مقایسه با روش‌های ارائه شده دیگر خواهیم پرداخت.
- در نهایت مقاله مرتبط با روش پیشنهادی را آماده می‌کنیم.

زمانبندی طرح

برنامه زمانی اجرای طرح در جدول زیر آورده شده است.

جدول ۳-۱ برنامه زمانی اجراء طرح

زمان(ماه) / فعالیت	۱-۲	۳-۴	۵-۶	۷-۸	۹-۱۰	۱۱-۱۲
تکمیل مطالعه مقالات مرتبط با روش‌های تعبیه گراف						
تکمیل مطالعه روش‌های پیش‌بینی یال در شبکه‌های چندلایه						
ارائه ایده اولیه حل مسئله						
انتخاب، جمع‌آوری و آماده‌سازی مجموعه‌داده						
پیاده‌سازی ابتدایی روش پیشنهادی						
استخراج و ارزیابی نتایج اولیه						
اصلاح و بهبود روش پیشنهادی باتوجه به بازخوردها						
استخراج و ارزیابی نتایج نهایی و مقایسه با روش‌های پیشین						
نگارش مقاله						

همکاران طرح

۱) سعید پیروزفر، دانشجوی کارشناسی ارشد، دانشگاه تهران

آدرس: تهران، خ کارگر شمالی، جنب دانشکده فنی دانشگاه تهران، دانشکده علوم و فنون نوین

۲) شقایق نجاری، دانشجوی دکتری، دانشگاه تهران

آدرس: تهران، خ کارگر شمالی، جنب دانشکده فنی دانشگاه تهران، دانشکده علوم و فنون نوین

مراجع

[1] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *Knowledge-Based Systems*, p. 19, 2018.

[2] A.-L. Barabási, "Network science," *Cambridge university press*, 2016.

[3] Y. Sun, J. Han, X. Yan, P. S. Yu and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, pp. 992-1003, 2011.

[4] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno and M. A., "Multilayer networks," *Journal of complex networks,*, vol. 2, no. 3, pp. 203-271, 2014.

[5] X. Wang, D. Bo, C. Shi, S. Fan, Y. Ye and P. S. Yu, "A Survey on Heterogeneous Graph Embedding: Methods, Techniques, Applications and Sources," *IEEE Transactions on Big Data*, 2022.

[6] W. L. Hamilton, R. Ying and J. Leskovec, "Representation Learning on Graphs: Methods and Applications," *arXiv:1709.05584v3 [cs.SI]*, 2018.

[7] M. Xu, "Understanding graph embedding methods and their applications," *SIAM Review*, vol. 63.4, pp. 825-853, 2021.

[8] P. Goyal, S. R. Chhetri and A. Canedo, "dyngraph2vec: Capturing network dynamics using dynamic graph representation learning," *Knowledge-Based Systems*, vol. 187, no. 104816, 2020.

[9] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan and Q. Mei, "Line: Large- scale information network embedding," *Proceedings 24th Interna- tional Conference on World Wide Web*, p. 1067–1077, 2015.

[10] D. Wang, P. Cui and W. Zhu, "Structural deep network embedding," *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining, ACM*, p. 1225–1234, 2016.

[11] B. Perozzi, R. Al-Rfou and S. Skiena, "Deepwalk: Online learning of social representations," *Proceedings 20th international conference on Knowledge discovery and data mining*, p. 701–710., 2014.

[12] C. H. Ding, X. He, H. Zha, M. Gu and H. D. Simon, "A min-max cut algorithm for graph partitioning and data clustering," *International Conference on Data Mining, IEEE*, p. 107–114., 2001.

[13] S.Bhagat, G.Cormode and S.Muthukrishnan, "Node classification in social networks," *Social network data analytics, Springer*, p. 115– 148., 2011.

[14] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining, ACM,* p. 855–864, 2016.

[15] E. Omodei, M. D. D. Domenico and A. Arenas, "“Characterizing interactions in online social networks during exceptional events," *Frontiers in Physics,* p. 59, 2015.

[16] M. Gustafsson, C. E. Nestor, H. Zhang, A.-L. Barabási, S. Baranzini and S. Brunak, "Modules, networks and systems medicine for understanding disease and aiding diagnosis," *Genome medicine,* vol. 6, no. 10, p. 82, 2014.

[17] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science 290 (5500),* p. 2323–2326, 2000.

[18] T. Kipf and M. Welling, "Variational graph auto-encoders," *NIPS Workshop on Bayesian Deep Learning,* 2016.

[19] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907.,* 2016.

[20] W. Hamilton, R. Ying and J. Leskovec, "Inductive representation learning on large graphs," *arXiv preprint, arXiv:1603.04467,* 2017.

[21] M.Belkin and P.Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," *NIPS,* 2002.

[22] A.Ahmed, N.Shervashidze, S.Narayanamurthy, V.Josifovski and A.J.Smola, "Distributed large-scale natural graph factorization," *WWW,* 2013.

[23] S.Cao, W.Lu and Q.Xu, "Grarep:Learning graph representations with global structural information," *KDD,* 2015.

[24] M. Ou , P. Cui, J. Pei, Z. Zhang and W. Zhu, "Asymmetric transitivity preserving graph embedding," *KDD,* 2016.

[25] M. Niepert, M. Ahmed and K. Kutzkov, "Learning convolutional neural networks for graphs," *Proceedings of the 33rd annual international conference on machine learning. ACM,* 2016.

[26] T. N. Kipf and M. Welling, "Variational graph auto-encoders," *arXiv preprint arXiv,* vol. 1611.07308, 2016.

[27] Y. Wang, B. Xu, M. Kwak and X. Zeng, "A simple training strategy for graph autoencoder.," *Proceedings of the 2020 12th International Conference on Machine Learning and Computing,* pp. 341-345, 2020.

[28] L. Pan, T. Zhou, L. Lü and C.-K. Hu, "“Predicting missing links and identifying spurious links via likelihood analysis," *Scientific reports,,* vol. 6, p. 22955, 2016.

[29] G.-P. G., R. Aliakbarisani, A. Ghasemi and M. Á. Serrano, "Precision as a measure of predictability of missing links in real networks," *Physical Review E,* vol. 101, p. 5, 2020.

[30] S. Najari, M. Salehi, V. Ranjbar and M. Jalili, "Link prediction in multiplex networks based on interlayer similarity," *Physica A: Statistical Mechanics and its Applications,* vol. 536, no. 120978, 2019.

[31] J. Zhang, X. Kong and P. S. Yu, "Transferring heterogeneous links across location-based social networks," *Proceedings of the 7th ACM international conference on Web search and data mining,* pp. 303-312, 2014.

[32] P. Kumar and D. Sharma, "A novel similarity measure for the link prediction in unipartite and bipartite networks," *Social Network Analysis and Mining,* vol. 11, no. 1, pp. 1-14, 2021.

[33] O. Allali, C. Magnien and M. Latapy, "“Link prediction in bipartite graphs using internal links and weighted projection," *Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 936-941, 2011.

[34] S. Xia, B. Dai, E.-P. Lim, Y. Zhang and C. Xing, "Link prediction for bipartite social networks: the role of structural holes," *Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 143-157, 2012.

[35] D. Hristova, A. Noulas, C. Brown, M. Musolesi and C. Mascolo, "A multilayer approach to multiplexity and link prediction in online geo-social networks," *EPJ Data Science*, vol. 5, no. 2, p. 24, 2016.

[36] Kanawati, M. Pujari and R., "Link prediction in multiplex networks," *NHM*, vol. 10, no. 1, pp. 17-35, 2015.

[37] M. Jalili, Y. Orouskhani, M. Asgari, N. Alipourfard and M. Perc, "Link prediction in multiplex online social networks," *Royal Society open science*, vol. 14, no. 2, p. 160863, 2017.

[38] J. Janssen, M. Hurshman and N. Kalyaniwalla, "“Model selection for social networks using graphlets," *Internet Mathematics*, vol. 8, no. 4, pp. 338-363, 2012.

[39] H. Luo, L. Li, H. Dong and X. Chen, "Link prediction in multiplex networks: An evidence theory method," *Knowledge-Based Systems*, vol. 257, no. 109932, 2022.

[40] N. A. Christakis and J. H. Fowler, "Social contagion theory: examining dynamic social networks and human behavior," *Statistics in medicine*, vol. 32, no. 4, pp. 556-577, 2013.

[41] M. E. Newman, "Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality," *Physical review E*, vol. 64, no. 1, p. 016132, 2001.

[42] A.-L. Barabási, Network Science, Cambridge University Press, 2016.

[43] F. Battiston, V. Nicosia and V. Latora, "Structural measures for multiplex networks," *PHYSICAL REVIEW E*, vol. 89, p. 032804, 2014.

[44] D. Zhao, L. Li, H. Peng, Q. Luo and Y. Yang, "Multiple routes transmitted epidemics on multiplex networks," *Physics Letters A*, vol. 378, no. 10, pp. 770-776, 2014.

[45] Y. Yao, R. Zhang, F. Yang, Y. Yuan, Q. Sun, Y. Qiu and R. Hu, "Link prediction via layer relevance of multiplex networks," vol. 28, no. 08, 2017.

[46] V. Gemmetto and D. Garlaschelli, "Multiplexity versus correlation: the role of local constraints in real multiplexes," *Scientific reports*, vol. 5, no. 1, pp. 1-7, 2015.

[47] R. J. Mondragón, "Estimating degree–degree correlation and network cores from the connectivity of high–degree nodes in complex networks," *Scientific reports*, vol. 10, no. 1, pp. 1-24, 2020.

[48] S. Zhou and R. Mondragón, "Accurately modeling the Internet topology," *Phys. Rev. E*, vol. 70, no. 066108, 2004.

[49] V. Colizza, A. Flammini, M. A. Serrano and A. Vespignani, "Detecting rich-club ordering in complex networks," *Nature*, vol. 2, pp. 110-115, 2006.

[50] A. E. Brouwer and W. H. Haemers, Spectra of graphs, Springer Science & Business Media, 2011.

[51] Philip, J. Zhang and S. Y., "Link prediction across heterogeneous social networks: A survey," *USA, University of Illinois at Chicago*, 2014.

[52] M. D. Domenico, C. Granell, M. A. Porter and A. Arenas, "The physics of spreading processes in multilayer networks," *Nature Physics*, vol. 12, no. 10, p. 901, 2016.

[53] M. D. Domenico, A. Solé-Ribalta, E. Cozzo, M. Kivelä, Y. Moreno, M. A. Porter, S. Gómez and A. Arenas, "Mathematical formulation of multilayer networks," *Physical Review X*, vol. 3, no. 4,

p. 041022, 2013.

[54] M. D. Domenico, C. Granell, M. A. Porter and A. Arenas, "The physics of spreading processes in multilayer networks," *Nature Physics*, vol. 12, no. 10, p. 901, 2016.

[55] S. Mishra, S. S. Singh, A. Kumar and B. Biswas, "HOPLP− MUL: link prediction in multiplex networks based on higher order paths and layer fusion," *Applied Intelligence*, pp. 1-29, 2022.

[56] A. C. Belkina, C. O. Ciccolella, R. Anno, R. Halpert, J. Spidlen and J. E. Snyder-Cappione, "Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets," *Nature communications*, vol. 10, no. 1, pp. 1-12, 2019.

[57] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 4, no. 2, 2010.

[58] McInnes, Leland, J. Healy and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[59] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *journal of the Association for Information Science and Technology 58 (7)*, p. 1019–1031, 2007.

[60] A. M. D. D. S. G. a. A. A. Solé-Ribalta, "Random walk centrality in interconnected multilayer networks," *Physica D: Nonlinear Phenomena*, vol. 323, pp. 73-79, 2016.

[61] L.v.d.Maaten and G.Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research 9*, p. 2579–2605, 2008.

[62] A. Solé-Ribalta, M. D. Domenico, S. Gómez and A. Arenas, "Random walk centrality in interconnected multilayer networks," *Physica D: Nonlinear Phenomena*, vol. 323, pp. 73-79, 2016.

[63] A. Grover and J. Leskovec, "Scalable Feature Learning for Networks," *arXiv:1607.00653v1 [cs.SI]*, 2016.

[64] A. Cardillo, Gómez-Gardenes, Z. J., R. M., P. M., P. D., F. D. and S. Boccaletti, "Emergence of network features from multiplexity," *Scientific reports*, vol. 3, pp. 1-6, 2013.

[65] Cao, Shaosheng, W. Lu and Q. Xu, "Deep neural networks for learning graph representations," *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, AAAI Press*, p. 1145–1152, 2016.