

مرور روش‌های جاسازی^۱ گراف‌های دانش

سید مهدی وحیدی پور*، استادیار، داود دانشمند، دانشجوی دکتری، محمد علی ظریف، دانشجوی دوره دکتری

۱ و ۲ دانشکده برق و کامپیوتر - دانشگاه کاشان - کاشان - ایران - vahidipour@kashanu.ac.ir

چکیده: گراف‌های دانش به عنوان ابزارهای حیاتی و بسیار پرکاربرد در زمینه‌های داده‌کاوی، یادگیری ماشین، شبکه‌های پیچیده، و حوزه‌های مرتبط، توجه بسیاری را به خود جلب کرده‌اند. در این مقاله، ما سعی داریم تا به بررسی گسترده‌ای از این گراف‌ها و روش‌های مرتبط با آنها بپردازیم. ما روش‌های شبکه‌های BLM و TDM، RGNN و BLM را جزئی‌تر بررسی و ارزیابی می‌کنیم و نتایج به دست آمده از هر کدام را با دقت مورد بررسی قرار می‌دهیم. این مطالعه یک تجزیه و تحلیل جامع از عملکرد این روش‌ها را ارائه می‌دهد و به دنبال ارزیابی دقیق‌تر نقاط قوت و ضعف هر یک از آنها در مواجهه با چالش‌ها و کاربردهای واقعی است. امیدواریم که این بررسی گامی مفید در جهت بهبود درک ما از این فناوری مهم باشد و در توسعه‌ی آینده‌ی آن موثر باشد.

واژه‌های کلیدی: گراف دانش، Knowledge Graph، RGNN، TDM، BLM، OGB

* سید مهدی وحیدی پور، vahidipour@kashanu.ac.ir

۱. مقدمه

یکی از مسائل اساسی یادگیری ماشین در گراف دانش، پیش‌بینی سه‌گانه جدید است؛ محاسبه احتمال وجود یک سه‌گانه. به عبارت دیگر، با استفاده از سه‌گانه‌های موجود در گراف یادگیری انجام شده و سه‌گانه‌های جدیدی پیش‌بینی شوند. استفاده از جاسازی گراف دانش^۵ (KGE) به عنوان یک روش یادگیری ماشین در این مسئله توسعه یافته است [7].

KGE برای هر موجودیت و هر رابطه یک نمایش برداری کم‌بعد (جاسازی^۶) ایجاد می‌کند. حال بر اساس مجموعه‌ای از

گراف دانش^۲ (KG) نوع خاصی از ساختار گراف با موجودیت‌ها به عنوان گره و روابط به عنوان یال، هم در داده‌کاوی و هم در یادگیری ماشین کاربرد دارد [1]. به عنوان مثال در یافتن هدف در تعامل کاربر و آیتم [2]، استدلال [3]، سامانه‌های پاسخ به سؤال^۳ [4]، سامانه‌های توصیه‌گر^۴ [5] و پردازش زبان‌های طبیعی کاربرد [6] دارد.

۲.۱ گراف دانش

یک گراف دانش با چهارتایی $G = (V, E, R, T)$ تعریف می‌شود که در آن V مجموعه تمام گره‌ها، E مجموعه تمام یال‌ها، R مجموعه تمام رابطه‌ها و T مجموعه تمام انواع گره‌ها است.

در گراف دانش هر یال به صورت سه‌گانه^{۱۲} نشان داده می‌شود: (سرآیند، رابطه، تالی) یا (Tail, Relation, Head). این سه گانه به صورت خلاصه به ترتیب با h و r و t نشان داده می‌شوند. Head و Tail هر کدام می‌توانند یک نود یا موجودیت باشند و Relation نیز رابطه بین این دو موجودیت است؛ این ترکیب سه‌گانه تشکیل یک حقیقت را می‌دهد.

گراف دانش، بر اساس تعداد زیادی سه‌گانه ساخته می‌شود و هر سه‌گانه بیانگر یک حقیقت در دنیای دانش است. به عنوان مثال جمله «علی پدر حسن، راننده است» را در نظر بگیرید. واضح است که این جمله ساده دارای دو حقیقت است: «نام پدر حسن، علی است» و «علی یک راننده است». این دو حقیقت، به سه‌گانه‌های زیر تبدیل می‌شوند:

(۱) (علی، شغل، راننده)، (۲) (علی، پدر، حسن) و (۳) (حسن، پسر، علی). با این سه‌گانه‌ها گراف دانش زیر به دست خواهد آمد:

$$V = \{ \text{"راننده"}, \text{"حسن"}, \text{"علی"} \}$$

$$E = \{ (\text{"علی"}, \text{"راننده"}), (\text{"حسن"}, \text{"پدر"}, \text{"علی"}), (\text{"علی"}, \text{"پسر"}, \text{"حسن"}) \}$$

$$R = \{ \text{"پسر"}, \text{"پدر"}, \text{"شغل"} \}$$

$$T = \{ \text{"شغل"}, \text{"انسان"} \}$$

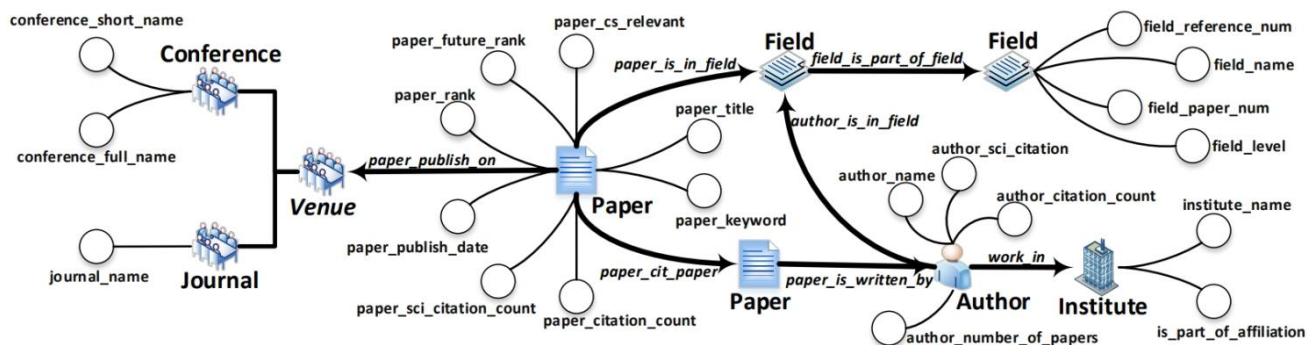
سه‌گانه‌های موجود در گراف دانش (داده‌های آموزشی)، جاسازی‌های ایجاد شده ارزیابی می‌شوند؛ از تابع امتیازدهی SF^7 برای ارزیابی استفاده می‌شود. با استفاده از این ارزیابی KGE تلاش می‌کند تا جاسازی‌های بهتری را برای موجودیت‌ها و روابط ایجاد کند (یاد بگیرد) که امتیاز بالاتری در SF دریافت کنند. این پروسه تکراری تا زمانی که عملکرد روی داده آموزشی بهبود می‌یابد ادامه خواهد داشت. همان‌طور که مشخص است یادگیری بهترین جاسازی در گراف دانش، نیازمند تابع امتیازدهی SF مناسب است.

هدف این مقاله، بررسی و توضیح روش‌های مختلف جاسازی گراف دانش امتیاز محور^۸ می‌باشد که خود دارای سه دسته کلی می‌باشد: (۱) روش‌های مبتنی بر مدل‌های فاصله‌ای^{۹، ۲} روش‌های مبتنی بر مدل‌سازی دو خطی^{۱۰} و (۳) روش‌های مبتنی بر شبکه‌های عصبی^{۱۱}. پس از آن مقایسه‌ای میان روش‌ها و دسته‌های مذکور انجام می‌دهیم.

ساختار مقاله در ادامه به صورت زیر است. در قسمت دوم به بررسی مفاهیم اولیه پرداخته می‌شود. در بخش سوم روش‌های جاسازی امتیاز محور به طور کامل مورد بحث و بررسی قرار می‌گیرند. در بخش چهارم روش‌های بحث شده در بخش سوم را بر اساس معیارها و مجموعه داده‌های مختلف مورد مقایسه قرار می‌دهیم.

۲. مفاهیم اولیه

این بخش به بحث و بررسی مفاهیم اولیه در حوزه گراف دانش می‌پردازد.



تصویر ۱: یک مثال تصویری از یک گراف دانش ناهمگون.

ب) پاسخ به سوالات: به طور سنتی، پاسخ به سؤالات با استفاده از محتوای بدون ساختار مانند پاراگراف، متن و یا صفحه ویکی پدیا انجام می‌شود. قابلیت‌های اکتشافی چنین سیستم‌هایی محدود به محتوای داده‌شده است، به عنوان مثال، با داشتن یک صفحه ویکی پدیا در مورد آلبرت انیشتین یک کاربر می‌تواند با پرسیدن سوال «آلبرت انیشتین متولد کجاست؟» محل تولد انیشتین را به دست آورد. اما پرسیدن سؤالات بیشتر در مورد این شهر (به عنوان مثال، «چند نفر در یوآام زندگی می‌کنند؟») نتایج مناسبی را بر نمی‌گردانند. در این حالت، ما به پیشینه ساختاری احتیاج داریم که گراف‌های دانش برای ما به همراه می‌آورند.

۲.۳ گراف‌های دانش در دسترس عموم

شرکت‌های بزرگ فناوری اطلاعات، گراف‌های دانش متنوعی را ارائه کرده‌اند. همچنین دانشگاه استنفورد نیز به تازگی مجموعه داده‌های متنوعی را برای تست و ارزیابی الگوریتم‌ها ارائه کرده است.^{۱۳} در ادامه به چند گراف دانش اشاره می‌شود که به صورت عمومی ارائه شده است:

الف) فری بیس^{۱۴}: این گراف دانش توسط شرکت گوگل و بر اساس داده‌های موجود در ویکی پدیا تولید شده است. اولین بار این گراف در سال ۲۰۰۷ ایجاد شده و توسعه آن در سال ۲۰۱۶ متوقف شده است. این گراف شامل بیش از ۵۰ میلیون گره، ۳۸ هزار نوع رابطه و بیش از ۳ میلیارد سه‌گانه است. با این حال بخش

در تصویر شماره ۱، یک مثال از گراف دانش در حوزه دانشگاه‌ها و تعاملات بین اعضای هیئت علمی و مقالات ارائه شده آمده است [8]. تمام اشیا (مانند مقالات، موسسات، نویسندگان) به عنوان موجودیت‌ها در AceKG نمایش داده می‌شوند. دو موجودیت می‌توانند باهم، یک رابطه داشته باشند. ویژگی‌های معمول هر موجودیت، شامل اعداد، تاریخ‌ها، رشته‌ها و دیگر متغیرهای ساده، نیز نمایش داده می‌شوند. موجودیت‌های مشابه در کلاس‌هایی گروه‌بندی می‌شوند. در مجموع، AceKG پنج کلاس از موجودیت‌های آکادمیک را تعریف می‌کند: مقالات، نویسندگان، مکان‌ها، حوزه‌های مطالعاتی و موسسات. واقعیت‌ها، شامل ویژگی‌های پرکاربرد هر موجودیت و روابط بین موجودیت‌ها، به صورت سه‌گانه‌ها در نمودار دانش توصیف می‌شوند.

۲.۲ کاربردهای گراف دانش

گراف‌های دانش دارای کاربردهای فراوانی هستند که در ادامه به دو کاربرد آن به صورت نمونه اشاره می‌شود [9]:

الف) ارائه اطلاعات: اگر گراف دانشی وجود داشته باشد که در آن نام فیلم‌ها و کارگردان‌های آن‌ها و همچنین بازیگران فیلم‌ها آمده باشد، این گراف می‌تواند برای پاسخ‌دهی به انواع کوئری‌ها قابل استفاده باشد، به عنوان مثال نام تمام فیلم‌های کارگردان فیلم «تایتانیک». اینگونه اطلاعات از طریق ساخت گراف دانش و تحلیل آن‌ها قابل ارائه هستند.

زیادی از رابطه‌ها در این گراف کامل نیستند. به عنوان مثال ۷۸ درصد از تمام اشخاصی که در این گراف ثبت شده‌اند، فاقد ویژگی «ملیت» هستند و ۹۳ درصد آن‌ها فاقد «تاریخ تولد» هستند. به همین دلیل لزوم توسعه ابزارهای تکمیل گراف، ضروری به نظر می‌رسد.

ب) ویکی‌دیتا: یک گراف دانش بسیار بزرگ و زنده که توسط مجموعه ویکی‌پدیا ارائه شده و همچنان در حال گسترش است [10].

ج) گراف‌ها BioKG^{۱۵} و WikiKG2^{۱۶}: این دو گراف دانش زیرمجموعه تمام مجموعه داده‌هایی هستند که در فریم‌ورک OGB ارائه شده‌اند [11]: برای اینکه یادگیری رابطه‌ای روی داده‌های زیست‌پزشکی، استانداردتر و قابل تکرارتر شود، یک گراف دانش بیولوژیکی جدید توسط دانشگاه استنفورد ارائه شده است که مجموعه‌ای از داده‌های رابطه‌ای انتخاب شده از پایگاه‌های بیولوژیکی باز را در قالبی یکپارچه با شناسه‌های مشترک و مرتبط به هم ارائه می‌دهد. مجموعه داده ogbl-biokg یک گراف دانش است که آقای لسکووک به همراه همکاران، با استفاده از داده‌های تعداد زیادی از مخازن داده‌های زیست پزشکی ایجاد کرده‌اند. این گراف، شامل ۵ نوع موجودیت است: بیماری‌ها (۱۰۶۸۷ گره)، پروتئین‌ها (۱۷۴۹۹)، داروها (۱۰۵۳۳ گره)، عوارض جانبی (۹۹۶۹ گره)، و عملکردهای پروتئین (۴۵۰۸۵ گره). ۵۱ نوع رابطه مستقیم وجود دارد که دو نوع موجودیت را به هم متصل می‌کند، از جمله ۳۹ نوع تداخل دارو-دارو، ۸ نوع تعامل پروتئین-پروتئین، و همچنین دارو-پروتئین، اثر جانبی دارو، دارو-پروتئین و روابط عملکرد-عملکرد. همه رابطه‌ها به عنوان لبه‌های جهت دار مدل می‌شوند، که در میان آن‌ها روابطی که انواع موجودات یکسان را به هم متصل می‌کنند (به عنوان مثال، پروتئین-پروتئین، دارو-دارو، عملکرد-عملکرد) همیشه متقارن هستند، به عنوان مثال، لبه‌ها دو

جهت هستند. این مجموعه داده مربوط به هر دو تحقیقات زیست پزشکی و بنیادی ML است. در سمت اساسی ML، مجموعه داده، چالش‌هایی را در مدیریت یک KG ناقص با مشاهدات متناقض احتمالی ارائه می‌کند. این به این دلیل است که مجموعه داده ogbl-biokg شامل برهمکنش‌های ناهمگنی است که از مقیاس مولکولی (به عنوان مثال، برهمکنش‌های پروتئین-پروتئین در یک سلول) تا کل جمعیت (مثلاً گزارش‌هایی از عوارض جانبی ناخواسته تجربه شده توسط بیماران در یک کشور خاص) را شامل می‌شود. علاوه بر این، سه‌گانه‌ها در KG از منابعی با سطوح مختلف اطمینان، از جمله بازخوانی‌های تجربی، حاشیه‌نویسی‌های تنظیم شده توسط انسان، و ابرداده‌های استخراج شده به‌طور خودکار به دست می‌آیند.

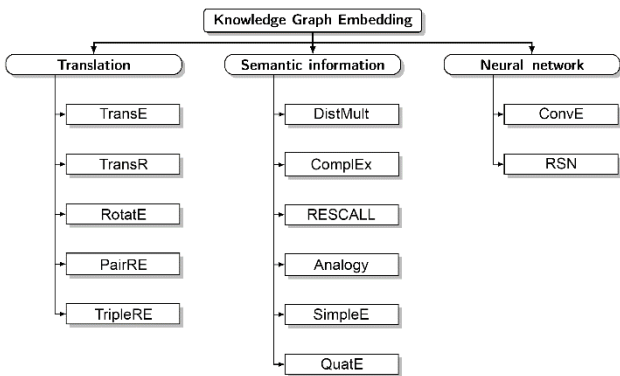
د) گراف دانش AceKG^{۱۷}: یک گراف دانش تخصصی در حوزه علمی و تحقیقاتی است که اطلاعات گسترده‌ای از مقالات، نویسندگان، مؤسسات و موضوعات تحقیقاتی را شامل می‌شود. این گراف که اولین بار در سال ۲۰۱۹ به عنوان یک پروژه تحقیقاتی معرفی شد، شامل بیش از ۲۰۰ میلیون مقاله، ۱۰ میلیون نویسنده و ۲۰ هزار مؤسسه است و روابط نویسنده-مقاله و نویسنده-مؤسسه را پوشش می‌دهد. اهداف این پروژه فراهم کردن یک گراف دانش بزرگ و جامع برای تحلیل و کاوش داده‌های علمی است. توسعه‌دهندگان این گراف از تکنیک‌های پیشرفته پردازش زبان طبیعی و یادگیری ماشین استفاده کردند تا داده‌های علمی را از منابع مختلف جمع‌آوری و ساختاردهی کنند.

بعضی از مهم‌ترین روابط در AceKG از قرار زیر هستند:

نویسنده-مقاله: این رابطه نویسندگان را به مقالاتی که نوشته‌اند متصل می‌کند و به تحلیل شبکه‌های همکاری علمی کمک می‌کند.

مقاله-کنفرانس ژورنال: این رابطه نشان می‌دهد که یک مقاله در کدام کنفرانس یا ژورنال منتشر شده است، و به بررسی تأثیرگذاری مقالات و کیفیت منابع انتشار کمک می‌کند.

به طور کلی، مدل‌های گراف دانش جاسازی (KGE) را می‌توان به سه دسته روش‌های مبتنی بر مدل‌های فاصله‌ای، روش‌های مبتنی بر مدل‌سازی دو خطی و روش‌های مبتنی بر شبکه‌های عصبی تقسیم کرد که در ادامه مقاله، هر دسته به صورت جدا مورد بررسی قرار می‌گیرد [13].



شکل ۱: نمودار درختی الگوریتم‌های جاسازی بررسی شده در مقاله

۴. TDM: روش‌های مبتنی بر مدل‌های فاصله‌ای^{۱۸}

این دسته از الگوریتم‌ها با الهام از قابلیت تغییرناپذیری نسبت به انتقال^{۱۹} که بار اول در روش word2vec ارائه شد، به وجود آمده‌اند. یک مدل انتقالی، متکی بر این اصل است که بردارهای جاسازی موجودیت‌ها پس از اعمال یک انتقال رابطه‌ای مناسب در فضای هندسی که در آن تعریف شده‌اند، به یکدیگر نزدیک خواهند بود. به عبارتی دیگر، زمانی که یک حقیقت را داریم با اضافه کردن جاسازی Head به جاسازی رابطه، نتیجه حاصله باید جاسازی Tail باشد. الگوریتم‌های متنوعی در این دسته قرار می‌گیرند که در ادامه ۵ عدد از آنها بررسی می‌شوند.

۴.۱ TransE

TransE تلاش می‌کند تا تمام گره‌ها را تبدیل به نقاطی در یک فضای d بعدی کرده و تمام رابطه‌ها را تبدیل به بردارهای انتقالی در این فضای حالت کند. به نحوی که اگر یک سرآیند h با رابطه r (که یک بردار انتقال است) جمع شود به نقطه‌ای برود که

نویسنده-مؤسسه: این رابطه، ارتباط بین نویسندگان و مؤسسه‌ای که در آن‌ها فعالیت می‌کنند را نمایش می‌دهد و به تحلیل پراکندگی جغرافیایی و سازمانی تحقیقات علمی می‌پردازد.

مقاله-موضوع تحقیقاتی: این رابطه مقالات را به موضوعات تحقیقاتی مرتبط متصل می‌کند و به شناسایی روندها و موضوعات داغ در تحقیقات علمی کمک می‌کند.

AceKG با این روابط متنوع به محققان و دانشجویان امکان می‌دهد تا به راحتی به اطلاعات علمی دسترسی پیدا کنند و شبکه‌های علمی را تحلیل کنند.

۳. تکمیل گراف با روش‌های جاسازی گراف دانش

همان‌طور که گفته شد، گراف دانش، کامل نیست و داده‌های بسیار زیادی هستند که لبه‌های مرتبط با آن‌ها در گراف دانش وجود ندارد [12]. بعضی از این لبه‌ها بر اساس لبه‌های دیگر قابل حدس هستند. به عنوان مثال اگر لبه (حسن، به دنیا آمده، تهران) در گراف دانش وجود داشته باشد و برای حسن رابطه ملیت وجود نداشته باشد، آنگاه سه‌گانه (حسن، ملیت، ایرانی) را می‌توان حدس زد و با اضافه کردن لبه متناظر با این سه‌گانه گراف دانش را تکمیل کرد.

جاسازی یک بردار ویژگی به ابعاد d است. استفاده از جاسازی مناسب گره‌ها و روابط، به تکمیل گراف از طریق پیش‌بینی لبه کمک می‌کند. در مثال قبل با داشتن بردار جاسازی گره «حسن» و بردار جاسازی رابطه «ملیت» به برداری می‌رسیم که در میان بردارهای جاسازی تمام ملیت‌های مختلف، کوتاهترین فاصله را با بردار جاسازی «ایران» دارد. بنابراین مسئله اصلی آن است که برای گره‌ها و روابط جاسازی‌هایی پیدا شود که در فضای برداری آن‌ها چنین معادلاتی برقرار باشد. روشهای KGE به دنبال یافتن (یا یاد گرفتن) چنین جاسازی‌هایی هستند.

و با استفاده از یک بهینه‌ساز به سمت کمینه‌کردن آن حرکت می‌شود.

$$L = \sum_{(h,l,t) \in S} \sum_{(h',l',t') \in S'} [\gamma + d(h+l.t) - d(h' + l'.t')]_+ \quad (2)$$

$$S'_{(h,l,t)} = \{(h'.l.t|h' \in E\} \cup \{(h.l.t')|t' \in E\} \quad (3)$$

یکی از مشکلات روش TransE هنگام مواجهه با روابط غیر «یک به یک» می‌باشد که برای حل این مشکل روش‌های متعددی ارائه شده است. یکی از این روش‌ها TransH می‌باشد [16]. این روش برای حل مشکل اختصاص دادن بردارهای روابط مختلف هنگام مواجهه با این نوع روابط، برای هر رابطه r یک ابرصفحه تعریف می‌کند و با استفاده از بردار translation و بردار نرمال w مربوط به رابطه، برای بازنمایی آن استفاده می‌کند. در مرجع [17] اثبات شده است که روش TransH عملکرد بهتری نسبت به TransE دارد. ویژگی‌های مربوط به روش جاسازی TransE توضیحات کامل در پیوست (الف) آمده است.

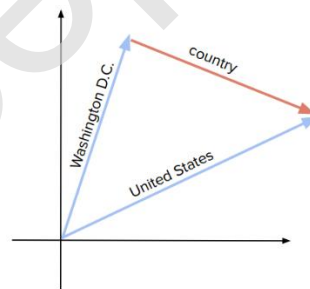
۴.۲ TransR

به دلیل بعضی مشکلات TransE، مدل TransR معرفی گردید [17]. ایده و تفاوت اصلی نسبت به TransE، آن است که به ازای هر رابطه، یک فضای حالت مجزا در نظر گرفته می‌شود. فرض کنید که گره سرآیند جاسازی h و گره تالی جاسازی t را دارد. در زمان خوانش سه‌گانه (h, r, t) ، بردار سرآیند و تالی به یک فضای دیگر نگاشت می‌شوند؛ معادله (۴) که در آن h_r و t_r بردارهای جاسازی جدید گره سرآیند و گره تالی را در فضای جاسازی جدید نشان می‌دهد. در آن فضای جدید بردار انتقالی رابطه r میان آن‌ها وجود دارد. روابط حاکم در این دو فضای جاسازی و انتقال بین آن‌ها در معادله (۵) و شکل ۳ دیده می‌شود [18] که در آن به ازای هر رابطه یک ماتریس M وجود دارد.

کمترین فاصله را با گره تالی t و همچنین بیشترین فاصله را با گره‌های دیگر داشته باشد [14]. سه‌گانه به صورت (h, r, t) نشان داده می‌شود. پس TransE به دنبال معادله $h + r \approx t$ است و لذا می‌توان تابع امتیاز آنرا در معادله (۱) تعریف کرد.

$$F_r(h, t) = -||h + r - t|| \quad (1)$$

هر چه حاصل جمع سرآیند و رابطه به تالی نزدیکتر باشد، مقدار این رابطه، بیشتر بوده و به سمت صفر میل خواهد کرد. تصویر ۲، نمایانگر یک مثال از TransE است [15]. در این مثال نود «واشنگتن دی سی» با رابطه «کشور» به نود «ایالات متحده» می‌رسد. TransE سعی می‌کند تا تمام نودها و لبه‌ها را طوی آموزش دیده و تبدیل به بردار کند که رابطه «ملیت»، تمام افراد را به ملیت حقیقی آن‌ها در گراف دانش متصل کند.



تصویر ۲: نمود بصری از یک رابطه و دو موجودیت در TransE

در زمان یادگیری، تعدادی لبه منفی را با تغییر در سرآیند و یا تالی، به عنوان لبه‌های آموزشی ایجاد می‌شوند. لبه منفی یا نمونه برداری منفی به لبه‌هایی گفته می‌شود که در حالت عادی در گراف دانش وجود ندارند ولی از آنجایی که در پروسه آموزش نیاز به داده غلط یا منفی داریم، این لبه‌ها را ایجاد می‌کنیم. نحوه ایجاد آن‌ها نیز همان‌طور که پیش‌تر گفته شد این است که به عنوان مثال یک لبه مثبت که در گراف دانش وجود دارد را انتخاب کرده و نود تالی آن را تغییر می‌دهیم. بدین ترتیب یک لبه منفی برای آموزش ایجاد می‌گردد. سپس تابع ضرر نوشته شده در رابطه (۲) باز نویسی شده

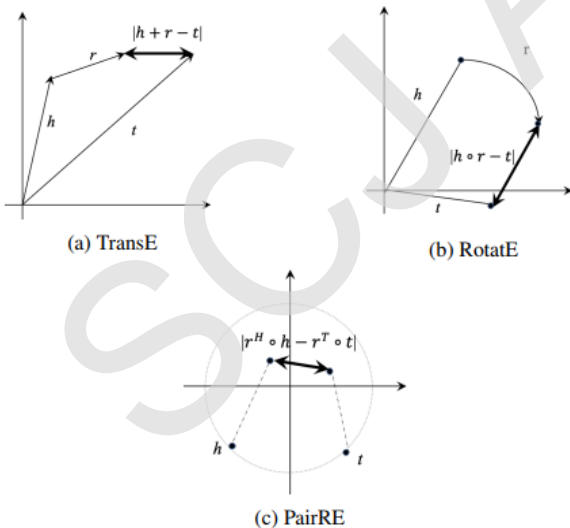
$$h_r = hM_r \quad t_r = tM_r \quad (4)$$

$$F_r(h, t) = \|h_r + r - t_r\| \quad (5)$$

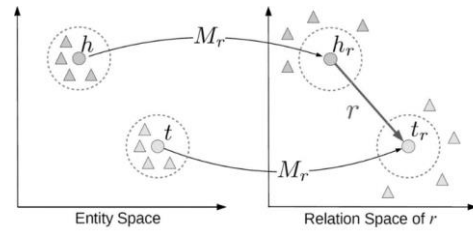
ضرب h در بردار r^H به آن می‌رسیم از نقطه‌ای که از ضرب t در بردار r^T به آن می‌رسیم، کمترین فاصله را باهم داشته باشند که این نکته را در تابع امتیاز این روش در معادله (۸) می‌توان مشاهده نمود.

$$F_r(h, t) = -\|h \circ r^H - t \circ r^T\| \quad (8)$$

در معادله ۸، بردار r^H نشان‌دهنده رابطه سرآیند و بردار r^T نشان‌دهنده رابطه تالی می‌باشد. تصویر ۴، روش‌های TransE، RotatE و PairRE را باهم مقایسه می‌کند. در این تصویر اختلاف بین نقاط به‌دست‌آمده مشاهده می‌شود. جاسازی باید این اختلاف را برای تمام نمونه‌ها به حداقل برساند. از آنجایی که تابع امتیاز روش TransR بسیار شبیه به TansE می‌باشد و تنها تفاوت این است که بردارهای سرآیند و تالی به فضای جدیدی برده می‌شوند، شکل آن در تصویر ۴ نیامده است و همانند شکل TransE خواهد بود.



تصویر ۴: مقایسه سه الگوریتم [20] TDM



تصویر ۳: نمودار بصری از نحوه تغییر فضا در الگوریتم TransR

بدین ترتیب می‌توان تابع خطا را در معادله (۶) بازنویسی کرد. برای یادگیری جاسازی‌ها و ماتریس‌های M از روش 20 SGD استفاده شده است. ویژگی‌های این روش نیز به همراه توضیحات آن در پیوست (الف) آمده است

$$L = \sum_{(h,r,t) \in \mathcal{ES}} \sum_{(h',r',t') \in \mathcal{ES}'} \max(0, f_r(h, t) + \gamma - f_r(h', t')) \quad (6)$$

۴.۳ RotatE

تابع امتیاز در RotatE در رابطه (۷) نشان داده شده است که عملگر \circ در آن «ضرب هادامارد»^{۲۱} است. این عملگر تک تک درایه‌های یکسان دو ماتریس و یا بردار هم‌سایز را در هم ضرب می‌کند. اگر ماتریس رابطه به درستی تنظیم شده باشد، می‌تواند باعث شود تا یک نقطه در فضا چرخش کرده و به یک نقطه دیگر برود. ویژگی‌های این روش نیز به همراه توضیحات آن در پیوست (الف) آمده است [18].

$$F_r(h, t) = -\|h \circ r - t\| \quad (7)$$

۴.۴ PairRE

این روش هم مشابه روش RotatE از ضرب هادامارد در تابع امتیاز خود استفاده می‌کند [19]. تفاوت این روش با روش‌های دیگر در آن است که در این روش برای هر رابطه دو بردار تعریف می‌شود که یکی از آن‌ها رابطه مربوط به h (سرآیند) و دیگری رابطه مربوط به t (تالی) می‌باشد. در نتیجه اینطور تصور می‌شود که اگر رابطه r بین h و t وجود داشته باشد، باید نقطه‌ای که از

۴.۵ TripleRE

این الگوریتم بر مبنای الگوریتم PairRE بنا شده است و نسبت به آن برتری‌هایی دارد. تابع امتیاز این روش در معادله (۹) آمده است [21].

$$F_r(h, t) = -||h \circ r^H - t \circ r^T + r^m|| \quad (9)$$

همان‌طور که در فرمول بالا مشخص است، این روش تنها در اضافه شدن بردار سوم r^m با روش PairRE متفاوت است [20]. هرچند که نسخه‌های دیگری نیز برای این الگوریتم با توابع امتیاز دیگر معرفی شده‌اند. بردار سوم، می‌تواند سبب شود که فاصله نهایی بین دو نقطه‌ای که از h و t با رابطه‌های r^H و r^T به آن‌ها رسیده‌ایم، کمتر شود. چرا که نهایتاً هرچقدر این فاصله زیاد باشد می‌تواند با یک بردار دیگر جمع زده شده و تابع امتیاز مقدار بیشتری داشته باشد.

روش‌ها و الگوریتم‌های متنوع دیگری مانند TransH در این حوزه تعریف و پیاده‌سازی شده‌اند. اثبات شده است که این دسته به طور کامل توصیف‌گر^{۲۲} نیستند و عملکرد تجربی آن‌ها نسبتاً به دیگر مدل‌ها کمتر می‌باشد [16].

۵. BLM: روش‌های مبتنی بر مدل‌سازی دو خطی^{۲۳}

این دسته از روش‌ها از یک معادله خطی برای جایگذاری^{۲۴} ارتباط بین موجودیت‌ها از طریق یک رابطه استفاده می‌کنند. در واقع بازنمایی رابطه جایگذاری شده، یک ماتریس دو بعدی می‌باشد. این مدل‌ها در طی پروسه جاسازی فقط از یک حقیقت^{۲۵} برای محاسبه بازنمایی استفاده می‌کنند و ارتباط با دیگر موجودیت‌ها و یا روابط را نادیده می‌گیرند. BLM های متفاوت از قیود مختلفی برای تنظیم سازی ماتریس R استفاده کرده‌اند تا بهتر با مجموعه داده‌های مختلف خود را انطباق دهند. در اینجا R همان ماتریس

مربعی مربوط به جاسازی رابطه می‌باشد. در ادامه سه روش از این دسته مورد بررسی قرار می‌گیرند.

۵.۱ DistMult

تابع امتیاز دهی مربوط به این روش در معادله (۱۰) آمده است. [22]

$$F_r(h, t) = \langle h, r, t \rangle = \sum_i h_i \cdot r_i \cdot t_i \quad (10)$$

در فرمول (۱۰)، r و t به ترتیب نشان دهنده تالی، رابطه و سرآیند می‌باشند و «نقطه» بین آن‌ها نیز ضرب داخلی می‌باشد. بر اساس رابطه (۱۰)، در تابع امتیازدهی DistMult برای محاسبه امتیاز وجود یک رابطه بین دو گره تالی و سرآیند، از حاصلضرب مقادیر بردار ویژگی گره‌های تالی و سرآیند در بردار ویژگی رابطه استفاده می‌شود. عمل ضرب، قابلیت جابجایی دارد، لذا مسلماً هر رابطه‌ای که بین سرآیند و تالی وجود داشته باشد، بین تالی و سرآیند هم وجود خواهد داشت. ویژگی‌های این جاسازی در پیوست (ب) مورد بررسی قرار گرفته است.

۵.۲ روش CP

روش Canonical Tensor Decomposition (CP) [23] همانند روش DistMult یک مدل تجزیه تنسور می‌باشد که چند تفاوت دارد روش CP به طور کلی یک روش ریاضیاتی برای تجزیه هر گونه تنسور می‌باشد که از آن به طرق مختلف در مسئله تکمیل گراف دانش استفاده شده. در مقاله مذکور برای بهبود عملکرد آن در گراف دانش نسبت به مقالات پیشین، ابتدا یک تنظیم ساز جدید بر پایه p -نرم‌های اتمی تنسوری ارائه شده است. سپس یک بازنمایی از مسئله ارائه می‌گردد که آن را نسبت به انتخاب دلخواه روابط تک جهته یا دو طرفه مقاوم می‌سازد. ترکیب این دو، روش ارائه شده در مقاله را نتیجه می‌دهد. یکی از تفاوت‌های این روش با DistMult این است که CP یک مدل تجزیه تنسور عمومی می‌باشد در حالی که DistMult منحصراً برای عمل تکمیل گراف

برای بازنمایی دوران در فضا استفاده کرد. در این روش هر موجودیت در گراف دانش با یک چهارگان نمایش داده می‌شود. سپس ارتباط بین موجودیت‌ها با ضرب چهارگان‌ها بازنمایی می‌گردند. روش QuatE چندین مزیت نسبت به دیگر روش‌های یادگیری جاسازی در گراف دانش دارد. مزیت اول این است که یافتن وابستگی‌های بلند مدت را دارد. دلیلش این است که از ضرب چهارگان‌ها برای بازنمایی دوران در فضا می‌توان استفاده نمود و دوران‌ها را می‌توان به یکدیگر متصل کرد تا دنباله‌های طولانی از روابط را بازنمایی نمود. دوم اینکه روش QuatE گویا تر^{۲۷} از دیگر روش‌ها می‌باشد. دلیلش نیز این است چهارگان‌ها دارای درجه آزادی بیشتری نسبت به اعداد حقیقی و یا اعداد مرکب هستند. این باعث می‌شود که این روش بتواند گستره وسیع‌تری از روابط بین موجودیت‌ها را بازنمایی کند. سوم اینکه QuatE بازدهی بهتری نسبت به دیگر روش‌ها دارد. دلیلش نیز این است که ضرب چهارگان‌ها از نظر محاسباتی یک عمل بهینه می‌باشد.

روش‌های مشابه دیگری همچون RESCALL ، Analogy^{۲۸} [26] و SimpleE [27] را نیز می‌توان در این دسته، بررسی کرد.

باید دقت شود که هر یک از روش‌های بالا و روش‌های دیگر، مزایا و معایب خود را دارد اما این مزایا و معایب کاملاً بستگی به گراف دانش دارد و باید بررسی کرد که به طور مثال آیا ویژگی تقارنی در گراف دانش وجود دارد یا خیر، اگر وجود نداشته باشد، مدل TransE، مدلی مناسب جهت جاسازی آن گراف دانش خواهد بود. [28]

۶. NNM: روش‌های مبتنی بر شبکه‌های عصبی

این دسته از روش‌ها خود دارای دسته‌های مختلفی مانند روش‌های یادگیری عمیق^{۲۹} (DNN)، روش‌های شبکه‌های عصبی کانولوشنی^{۳۰} (CNN) و روش‌های شبکه‌های بازگشتی^{۳۱} (RNN)

دانش طراحی شده است. به عبارتی روش CP به طور کلی توانایی تجزیه هر تنسوری را دارد ولی منحصراً برای تجزیه تنسورهای گراف دانش طراحی شده است که این باعث می‌شود که DistMult بتواند از ساختار خاص مربوط به گراف دانش استفاده کند. تفاوت دیگر دو روش این است که روش CP از نظر محاسباتی پرهزینه‌تر از DistMult می‌باشد. این امر بدین دلیل است که CP باید تنسور بزرگتری را تجزیه کند. اما از طرفی این اتفاق می‌تواند باعث گویاتر شدن این روش نسبت به DistMult شود و در نتیجه برخی از اوقات عملکرد بهتری نسبت به DistMult پیدا می‌کند.

۵.۳ Complex

در این روش، مقادیر بردارها می‌توانند عدد مختلط باشند. همین مسئله قدرت جاسازی را افزایش می‌دهد [24]. لذا تنها تفاوت بین تابع امتیاز Complex با DistMult این است که اعداد شرکت کننده در عملیات ضرب، از نوع مختلط هستند و در نهایت تنها قسمت حقیقی عدد مختلط به دست آمده به عنوان مقدار امتیاز تشابه برای آن سه‌گانه در نظر گرفته می‌شود. تابع امتیاز این روش در معادله (۱۱) آمده است.

$$f_r(h, t) = \langle h, r, t \rangle = \text{Re} \left(\sum_i h_i \cdot r_i \cdot t_i \right) \quad (11)$$

بدیهی است اگر مقدار موهومی در تمام بردارها را صفر در نظر بگیریم، همان روش DistMult را خواهیم داشت. ویژگی‌های مربوط به این روش در پیوست (ج) مورد بررسی قرار گرفته است.

۵.۴ QuatE

از دیگر روش‌های یادگیری جاسازی موجودیت‌ها در گراف دانش که سعی بر به دست آوردن وابستگی‌های ارتباطی بلند مدت هستند، می‌توان به روش QuatE اشاره کرد [25]. این روش بر پایه چهارگان‌ها^{۲۶} می‌باشد که نوعی عدد هستند که از آن‌ها می‌توان

دارند و باعث می‌شود که نتوانند وابستگی‌های دراز مدت بین موجودیت‌ها را به دست آورند. دلیلش این است که سه‌گانه‌ها تنها یک قدم از مسیر رابطه‌ای را بازنمایی می‌کنند در حالی که معنی یک موجودیت می‌تواند متأثر از روابطش با موجودیت‌هایی باشد که چند قدم فاصله دارند. برای حل این مشکل، نویسندگان روشی به نام شبکه‌های بازگشتی پرشی (RSNs) ارائه کردند. این شبکه‌ها نوعی شبکه عصبی بازگشتی هستند که قابلیت یادگیری از وابستگی‌های ارتباطی بلند مدت را دارند. آن‌ها ابتدا با ساخت دنباله‌ای مسیرهای ارتباطی که موجودیت‌ها را در گراف دانش به یکدیگر متصل می‌کنند، کار خود را شروع می‌نمایند. سپس RSN با توجه و در نظر گرفتن موجودیت‌های قبلی و ارتباط بین آن‌ها، پیش‌بینی موجودیت بعدی در دنباله را یاد می‌گیرد.

۷. مقایسه روش‌ها

مدل‌های انتخاب شده برای این مطالعه از دسته‌های مختلفی هستند. برای مقایسه آن‌ها، چند مجموعه داده را انتخاب کرده و تمامی روش‌ها را بر روی آن‌ها، اجرا می‌کنیم. سپس جاسازی حاصله از هر روش را با معیارهایی سنجیده و نتایج را با هم مقایسه کرده و در نهایت تفسیری از نتایج به دست آمده ارائه می‌دهیم.

۷.۱ معیارهای ارزیابی

روشهای مختلف جاسازی گراف‌های دانش، اکثراً با استفاده از تسک‌های پیش‌بینی لبه ارزیابی می‌شوند. معیارهای بسیاری برای ارزیابی جاسازی گراف دانش موجود است؛ با این حال، معمولاً معیارهای میانگین رتبه (MR)، میانگین رتبه تنظیم شده (AMR)، میانگین نرخ معکوس (MMR) و موفقیت در (Hits@K) در بررسی روش‌ها، استفاده می‌شوند. در ادامه دو معیار MMR و Hits@K را بررسی می‌کنیم.

می‌باشند. هدف مدل‌های عصبی خروجی دادن احتمال سه‌گانه‌ها بر اساس شبکه‌هایی می‌باشد که جاسازی‌های روابط و موجودیت‌ها را به عنوان ورودی دریافت می‌کنند. در روش‌های یادگیری عمیق از شبکه‌های عصبی عمیق برای یادگیری الگوها از گراف دانش که همان ورودی می‌باشد، استفاده می‌کنند. با وجود پردازش سنگین در قسمت آموزش، استفاده از شبکه‌های عمیق در جاسازی گراف دانش عملکرد خوبی از خود نشان داده است. روش MLP ارائه شده در مرجع [29] یکی از جدیدترین تلاش‌ها در استفاده از مدل‌های عصبی می‌باشد. هر دو روش از تعداد زیادی پارامتر برای ترکیب جاسازی‌های موجودیت‌ها و روابط استفاده می‌کنند. روش ConvE از شبکه عصبی کانولوشنی برای افزایش تعامل بین ابعاد مختلف جاسازی‌ها بهره می‌برد [30] یکی از روش‌های بر پایه شبکه‌های بازگشتی، RSN می‌باشد که در بخش ۶.۱ مورد بررسی قرار می‌گیرد.

با توجه به موفقیت شبکه‌های عصبی عمیق در زمینه‌های مختلف، بعضی از این مدل‌های شبکه‌های عصبی به عنوان تابع امتیاز مورد بررسی قرار گرفته‌اند. با وجود اینکه شبکه‌های عصبی توانایی توصیفی بالایی دارند ولی مدل‌های شبکه‌های عصبی در گراف‌های دانش به دلیل عدم تنظیم سازی مناسب، عملکرد خوبی ندارند. در بین توابع امتیاز موجود، توابع بر پایه بی‌لام، بر اساس نتایج عملی و تضمین تئوری برای توصیف‌گر بودن، قوی‌ترین هستند.

۶.۱ روش RSN

این روش یکی از روش‌های شبکه‌های بازگشتی می‌باشد. مزیت این دسته از روش‌ها به خاطر سپردن دنباله‌ای از حقیقت‌ها می‌باشد و مانند مابقی روش‌ها تنها به یک پدیده دقت نمی‌کنند [31]. این روش سعی بر حل مشکل روش‌های فعلی یادگیری جاسازی گراف‌های دانش دارد که معمولاً بر یادگیری از سه‌گانه‌ها تمرکز

$$Hits@K = \frac{|\{q \in Q : q < k\}|}{|Q|} \in [0,1] \quad (13)$$

مقادیر بیشتر Hits@K نشان دهنده قدرت پیش بینی کنندگی بهتر است.

۷.۲ آزمایشات

برخی از مهم‌ترین روش‌هایی که در این مقاله بررسی شده‌اند بر روی مجموعه داده‌ها و سخت افزارهای مختلف ارزیابی شده‌اند. در ادامه نتایج این آزمایشات شرح داده می‌شوند.

۷.۲.۱ مجموعه داده‌ها

OGB مجموعه‌ای از مجموعه داده‌های بنچمارک با مقیاس بزرگ، واقعی و گسترده می‌باشد که برای استفاده از یادگیری ماشین بر روی گراف‌ها به کار گرفته می‌شوند. مجموعه داده‌های OGB به صورت خودکار دانلود، پردازش و توسط OGB Data Loader تقسیم‌بندی می‌گردند. دو مجموعه داده BioKG و WIKIKG2 در این فریمورک موجودند که برای ارزیابی بهتر، روش‌های مختلف بر روی این دو مجموعه داده اجرا شدند تا بتوان نتایج را باهم مقایسه کرد.

همچنین در جداول ۳ تا ۵ [32] نیز روش‌های مختلف من جمله روش‌های بحث شده از نظر معیارهای MRR، H@1 و H@10 در مجموعه داده‌های WN18، FB15K و WN18RR مورد مقایسه قرار گرفته‌اند. نتایج بهتر پررنگ شده‌اند.

۷.۲.۲ آزمایش اول: بررسی اجرای الگوریتم‌ها بر روی یک

مجموعه داده یکسان

۷.۱.۱ معیار MRR

معیار میانگین رتبه بندی متقابل^{۳۲} یک معیار آماری برای ارزیابی هر پردازشی است که لیستی از جواب‌های محتمل، ناشی از نمونه‌ای از کوئری‌ها را به ما خروجی می‌دهد که این لیست بر اساس احتمال درست بودن مرتب شده است. رتبه دو طرفه یک جواب کوئری، وارون ضریبی اولین پاسخ صحیح می‌باشد. برای رتبه اول $\frac{1}{2}$ برای رتبه دوم $\frac{1}{3}$ برای رتبه سوم و الی آخر. میانگین رتبه بندی متقابل یا دو طرفه، میانگین رتبه‌های دو طرفه نتایج مربوط به نمونه‌ای از کوئری‌های Q می‌باشد که در فرمول (۲۶) قابل مشاهده می‌باشد.

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} \quad (12)$$

که در این رابطه $rank_i$ مربوط به رتبه اولین سند^{۳۳} مرتبط برای کوئری i ام می‌باشد.

۷.۱.۲ معیار H@10

معیار Hits@K و یا به طور مختصر H@K، یک معیار ارزیابی برای احتمال پیدا کردن پیش بینی صحیح در K پیش بینی اول

مجموعه داده	تعداد موجودیت‌ها	تعداد انواع روابط	تعداد روابط
WN18	۴۰,۹۲۳	۱۸	۱۴۱,۴۴۲
FB15k	۱۴,۹۵۱	۱۳۴۵	۴۸۴,۱۴۲
WN18RR	۴۰,۹۴۳	۱۱	۸۶,۸۳۵
FB15k237	۱۴,۵۴۱	۲۳۷	۲۷۲,۱۱۵
YAGO3-10	۱۲۳,۱۸۸	۳۷	۱,۰۷۹,۰۴۰
BioKG	۹۳,۷۷۳	۵	۵,۰۸۸,۴۳۴
WIKIKG2	۲,۵۰۰,۶۰۴	۵۳۵	۱۷,۱۳,۱۸۱

مدل می‌باشد. معمولاً مقدار K برابر ۱۰ قرار می‌گیرد. Hits@K نشان دهنده دقت یک مدل جایگذاری در پیش بینی صحیح رابطه بین دو سه‌گانه است. در فرمول (۲۷) رابطه مربوط به این معیار قابل مشاهده می‌باشد.

روش‌های مختلف را به صورت خودکار بر روی مجموعه داده بررسی کرده و بهترین روش را به عنوان پاسخ ارائه داد و سپس جاسازی‌ها را با این پاسخ تولید کرد. به همین دلیل امتیازات بهتری نسبت به بقیه روش‌ها در این روش‌ها به دست می‌آید.

روش TripleRE، مبتنی بر روش PairRE است که بهبودهایی بر روی تابع امتیازدهی آن انجام شده است. به دلیل تابع امتیاز پیچیده در این روش، پارامترهای آموزشی این روش، از بقیه روش‌ها بیشتر است و فرایند آموزش کندتر جلو می‌رود.

۷.۲.۳ آزمایش دوم: بررسی اجرای الگوریتم‌ها بر روی

سه مجموعه داده مختلف

روش‌های مختلف بر روی سه مجموعه داده WN18RR، WN18K و WN18 اجرا شده و نتایج مطابق جداول ۳ تا ۵ است.

تفسیر نتایج مختلف الگوریتم‌های امپدینگ در مجموعه داده‌های

مختلف نیاز به درک تفاوت‌های ذاتی بین این مجموعه داده‌ها و خصوصیات هر یک از الگوریتم‌ها دارد.

WN18					
دسته	مدل	MRR	H@1	H@10	
TDM	TransH	۰/۵۲۱	-	۹۴/۵	
	RotateE	۰/۹۴۹	۹۴/۴	۹۵/۹	
	ConvE	۰/۹۴۲	۹۳/۵	۹۵/۵	
NNM	RSN	۰/۹۴	۹۲/۲	۹۵/۳	
	BLM	DistMult	۰/۸۲۱	۷۱/۷	۹۵/۲
		Simple/CP	۰/۹۵	۹۴/۵	۹۵/۹
		Hole/Complex	۰/۹۵۱	۹۴/۵	۹۵/۷
Auto	Analogy	۰/۹۵	۹۴/۶	۹۵/۷	
	QuatE	۰/۹۵	۹۴/۵	۹۵/۹	
	AutoBLM	۰/۹۵۲	۹۴/۷	۹۶/۱	
	AutoBLM+	۰/۹۵۲	۹۴/۷	۹۶/۱	

در ادامه به بررسی این تفاوت‌ها و تاثیرات آن‌ها بر نتایج می‌پردازیم:

تفاوت‌های مجموعه داده‌ها:

در این آزمایش، نتایج روش‌های مختلف بر روی مجموعه داده BioKG بررسی می‌شوند. نتایج در جدول ۲ قابل مشاهده هستند [33]. مجموعه داده BioKG شامل ۵ نوع رابطه متفاوت است. مانند (پروتئین-پروتئین، دارو-دارو، غیره). همانطور که مشخص است در مجموعه داده، داده‌هایی موجود هستند که متقارن هستند. به عنوان مثال پروتئین «الف» بر روی پروتئین «ب» اثر دارد و بالعکس. از آنجایی که ماهیت تابع امتیاز دهی و روش TransE نمی‌تواند ویژگی «تقارن» را مدل کند، بالطبع داده‌های آموزشی هستند که این روش به هیچ وجه، قادر به یادگیری آنها نیست و به همین سبب کمترین امتیاز در بین تمام روش‌ها به TransE تعلق دارد. RotatE این نقص را نداشته و به همین دلیل نسبت به TransE عملکرد بهتری دارد. ComplEx و DistMult در رده‌های بعدی قرار دارند که متعلق به دسته BLM هستند. از آنجایی که ComplEx از سیستم اعداد مختلط استفاده می‌کند، برتری نسبی، نسبت به روش DistMult دارد.

روش	MRR تست	MRR اعتبارسنجی	تعداد پارامترها
۱ AutoBLM-KGBench	۰/۸۵۳۶	۰/۸۵۴۸	۱۹۲/۰۴۷/۱۰۴
۲ ComplEx-RP	۰/۸۴۹۲	۰/۸۴۹۷	۱۸۷/۷۵/۰۰۰
۳ TripleRE	۰/۸۳۴۸	۰/۸۳۶۰	۴۹۶/۶۳۰/۰۰۲
۴ AutoSF	۰/۸۳۰۹	۰/۸۳۱۷	۹۳/۸۲۴/۰۰۰
۵ PairRE	۰/۸۱۶۴	۰/۸۱۷۲	۱۸۷/۷۵/۰۰۰
۶ ComplEx	۰/۸۰۹۵	۰/۸۱۰۵	۱۸۷/۶۴۸/۰۰۰
۷ DistMult	۰/۸۰۲۳	۰/۸۰۵۵	۱۸۷/۶۴۸/۰۰۰
۸ RotatE	۰/۷۹۸۹	۰/۷۹۹۷	۱۸۷/۵۹۷/۰۰۰
۹ TransE	۰/۷۴۵۲	۰/۷۴۵۶	۱۸۷/۶۴۸/۰۰۰

به طور کلی، عملکرد روش‌های BLM نسبت به روش‌های TDM بهتر است اما روش PairER به دلیل تابع امتیازدهی بهبود یافته و دوطرفه خود، نتایج بهتری کسب کرده‌است.

روش‌های AutoBLM [34] و AutoSF [32]، روش‌های مبتنی بر جستجو هستند. از آنجایی که داده‌ها و ارتباط بین داده‌ها در مجموعه داده‌های مختلف با هم متفاوت است، لذا می‌توان انواع

۴۵/۱	-	۰/۱۸۶	TransH	TDM
۵۷/۱	۴۲/۸	۰/۴۶۶	RotateE	
۴۸	۳۹	۰/۴۶	ConvE	NNM
۵۰/۷	۴۰/۴	۰/۴۳۳	DistMult	BLM
۵۵/۱	۴۲/۴	۰/۴۶۲	Simple/CP	
۵۵/۱	۴۳	۰/۴۷۱	HolE/CompLex	
۵۵/۴	۴۲/۹	۰/۴۶۷	Analogy	
۵۸/۲	۴۳/۸	۰/۴۸۸	QuatE	Auto
۵۶/۷	۴۵/۱	۰/۴۹	AutoBLM	
۵۶/۷	۴۵/۲	۰/۴۹۲	AutoBLM+	

تأثیرات الگوریتم‌ها:

مدل‌های مبتنی بر ترجمه (TDM): این مدل‌ها مانند TransH و RotateE در مجموعه داده‌هایی با روابط پیچیده و متنوع مانند FB15K و WN18 عملکرد خوبی دارند. اما در مجموعه داده WN18RR که چالش‌های بیشتری دارد، نتایج ضعیف‌تری دارد چون نمی‌توانند به همان اندازه روابط پیچیده را مدل‌سازی کنند.

مدل‌های شبکه عصبی (NNM): مدل‌هایی مانند ConvE برای یادگیری ویژگی‌های پیچیده و روابط غیرخطی طراحی شده‌اند. این مدل‌ها در مجموعه داده‌هایی که تنوع زیادی در روابط دارند مانند FB15K و WN18 می‌توانند عملکرد خوبی داشته باشند، اما همانطور که در نتایج جدول مشخص شده است، در مجموعه داده چالش‌برانگیز WN18RR که نیاز به تعمیم‌دهی بیشتری دارد، کمتر موثر است.

مدل‌های باینری و لاجیک (BLM): مدل‌هایی مانند DistMult و ComplEx قادر به مدل‌سازی روابط پیچیده هستند. این مدل‌ها در همه‌ی مجموعه داده‌ها عملکرد خوبی دارند، به ویژه در مجموعه داده‌هایی که دارای ساختار پیچیده و چندلایه هستند.

مدل‌های خودکار (Auto): مدل‌های AutoBLM و +AutoBLM به دلیل استفاده از روش‌های خودکار و پیشرفته، توانایی بهینه‌سازی پارامترها و تعمیم‌دهی بیشتری دارند. این مدل‌ها در

WN18: این مجموعه داده شامل سه‌گانه‌هایی از یک گراف معنایی بزرگتر است. ساختار آن شامل تعداد زیادی روابط مختلف است که برای الگوریتم‌های مبتنی بر شبکه‌های عصبی و روش‌های ترجمه‌ای مناسب است. این تنوع باعث می‌شود مدل‌های مختلف عملکرد خوبی داشته باشند، به ویژه مدل‌هایی که می‌توانند پیچیدگی‌های شبکه را در نظر بگیرند.

FB15K: این مجموعه داده شامل داده‌های روابط در پایگاه داده بزرگ Freebase است و شامل تعداد زیادی از موجودیت‌ها و روابط پیچیده است. این پیچیدگی و تنوع در روابط، به الگوریتم‌هایی که می‌توانند بهتر از روابط پیچیده استفاده کنند، مانند مدل‌های خودکار و مدل‌های مبتنی بر ترجمه، کمک می‌کند تا عملکرد بهتری داشته باشند.

FB15K				
H@10	H@1	MRR	مدل	دسته
۷۶/۶	-	۰/۴۵۲	TransH	TDM
۸۸/۴	۷۴/۶	۰/۸۹۷	RotateE	
۸۹/۶	۷۶/۵	۰/۸۱۱	PairE	
۸۷/۳	۶۷	۰/۷۴۵	ConvE	NNM
۸۷/۲	۷۱/۴	۰/۸۷۵	DistMult	BLM
۹۰/۱	۷۹/۴	۰/۸۲۶	Simple/CP	
۹۰/۵	۷۹/۶	۰/۸۳۱	HolE/CompLex	
۸۹/۸	۷۸	۰/۸۱۶	Analogy	
۹۰	۷۱/۱	۰/۷۸۲	QuatE	Auto
۹۱	۸۲/۱	۰/۸۵۳	AutoBLM	
۹۱/۳	۸۳/۲	۰/۸۶۱	AutoBLM+	

WN18RR: این مجموعه داده نسخه‌ی کاهش یافته و بهبود یافته‌ی WN18 است که روابط معکوس تکراری را حذف کرده است تا از یادگیری بر مبنای میان‌برهای ساده جلوگیری کند. این ویژگی باعث می‌شود که این مجموعه داده به مراتب چالش‌برانگیزتر باشد و نیاز به مدل‌هایی داشته باشد که بتوانند روابط پیچیده‌تری را یاد بگیرند و تعمیم‌دهی بیشتری داشته باشند.

WN18RR				
H@10	H@1	MRR	مدل	دسته

مجموعه داده است. مجموعه داده‌های پیچیده‌تر و چالش‌برانگیزتر نیاز به الگوریتم‌های قوی‌تری دارند که بتوانند روابط پیچیده را مدل‌سازی و تعمیم دهند. مدل‌های خودکار به دلیل انعطاف‌پذیری و توانایی بهینه‌سازی پارامترها، در تمامی موارد عملکرد برتری داشته‌اند.

تمامی مجموعه داده‌ها عملکرد برتری داشته‌اند، زیرا قادر به انطباق با انواع مختلف داده‌ها و روابط پیچیده هستند.

تفاوت در عملکرد الگوریتم‌های امبدینگ در مجموعه داده‌های مختلف عمدتاً به دلیل تفاوت در پیچیدگی و نوع روابط در هر

Methods in Natural Language Processing, Abu Dhabi, 2020. DOI: 10.48550/arXiv.2004.14224.

مراجع

- [7] X. Zou, "A Survey on Application of Knowledge Graph," in *4th International Conference on Control Engineering and Artificial Intelligence*, Singapore, 2020. DOI: 10.1088/1742-6596/1487/1/012016.
- [8] X. Wang and L. Fu, "AceMap: Knowledge Discovery through Academic Graph," arXiv, 2024.
- [9] S. Ji, S. Pan, E. Cambria, P. Marttinen and P. S Yu, "A Survey on Knowledge Graphs: Representation, Acquisition, and Applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, 2022. DOI: 10.1109/TNNLS.2021.3070843.
- [10] "Welcome to Wikidata," 22 1 2023. [Online]. Available: https://www.wikidata.org/wiki/Wikidata:Main_Page.
- [11] W. Hu, Matthias Fey, Marinka Zitnik and Jure Leskovec, "Open Graph Benchmark: Datasets for Machine Learning on Graphs," in *34th Conference on Neural Information Processing Systems*, Vancouver, 2020. DOI: 10.48550/arXiv.2005.00687.
- [12] Z. Chen, Y. Wang, B. Zhao, J. Cheng, X. Zhao and Z. Duan, "Knowledge Graph Completion: A Review," *IEEE Access*, vol. 8, 2020. DOI: 10.1109/ACCESS.2020.3030076.
- [13] Y. Dai, S. Wan, N. N. Xiong and W. Guo, "A Survey on Knowledge Graph Embedding: Approaches, Applications and Benchmarks," mdpi.com, 2020. DOI: 10.3390/electronics9050750.
- [14] Antoine Bordes, Nicolas Usunier and Alberto Garcia-Duran, "Translating Embeddings for Modeling Multi-relational Data," *Neural Information Processing Systems*, 2013.
- [1] A. Singhal, "Introducing the Knowledge Graph: Things, not Strings," Official Google blog, 2012. [Online]. Available: <https://blog.google/products/search/introducing-knowledge-graph-things-not/>.
- [2] X. Wang, T. Huang, D. Wang, Y. Yuan, Z. Liu, X. He and T. S. Chua, "Learning Intents behind Interactions with Knowledge Graph for Recommendation," in *Association for Computing Machinery*, New York, 2021. DOI: 10.1145/3442381.3450133.
- [3] X. Chen, S. Jia and Y. Xiang, "A review: Knowledge Reasoning over Knowledge Graph," *Expert Systems with Applications*, vol. 141, 2020. DOI: 10.1016/j.eswa.2019.112948.
- [4] X. Li, F. Yin, Z. Sun, X. Li, A. Yuan, D. Chai, M. Zhou and J. Li, "Entity-Relation Extraction as Multi-Turn Question Answering," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, 2019. DOI: 10.18653/v1/P19-1129.
- [5] F. Zhang, N. Jing Yuan, D. Lian, X. Xie and W. Y. Ma, "Collaborative Knowledge Base Embedding for Recommender Systems," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, 2016. DOI: 10.1145/2939672.2939673.
- [6] T. Shen, Y. Mao, P. He, G. Long, A. Trischler and W. Chen, "Exploiting Structured Knowledge in Text via Graph-Guided Representation Learning," in *Proceedings of the 2020 Conference on Empirical*

- [24] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier and Guillaume Bouchard, "Complex Embeddings for Simple Link Prediction," in *International Conference on Machine Learning*, New York, 2016. DOI: 10.48550/arXiv.1606.06357.
- [25] S. Zhang, Y. Tay, L. Yao and Q. Liu, "Quaternion Knowledge Graph Embeddings," in *33rd Conference on Neural Information Processing Systems*, Vancouver, 2019. DOI: 10.48550/arXiv.1904.10281.
- [26] H. Liu, Y. Wu and Y. Yang, "Analogical Inference for Multi-relational Embeddings," in *Proceedings of the 34th International Conference on Machine Learning*, 2017. DOI: 10.48550/arXiv.1705.02426.
- [27] S. M. Kazemi and D. Poole, "Simple Embedding for Link Prediction in Knowledge," in *32nd Conference on Neural Information Processing Systems*, Montréal, 2018. DOI: 10.48550/arXiv.1802.04868.
- [28] A. Rossi, D. Barbosa, D. Firmani, A. Matinata and P. Merialdo, "Knowledge Graph Embedding for Link Prediction: A Comparative Analysis," *ACM Transactions on Knowledge Discovery from Data*, vol. 15, no. 2, 2021. DOI: 10.1145/3424672.
- [29] Q. Xu, K. Ren, X. Ren, S. Long and X. Li, "MlpE: Knowledge Graph Embedding with Multilayer Perceptron Networks," *2022 IEEE Smartworld, Ubiquitous Intelligence & Computing, Scalable Computing & Communications, Digital Twin, Privacy Computing, Metaverse, Autonomous & Trusted Vehicles*, 2022. DOI: 10.1109/SmartWorld-UIC-ATC-ScalCom-DigitalTwin-PriComp-Metaverse56740.2022.00130.
- [30] T. Dettmers, P. Minervini, P. Stenetorp and S. Riedel, "Convolution 2D knowledge graph embeddings," in *32nd AAAI Conference on Artificial Intelligence and Thirtieth Innovative*, 2018. DOI: 10.48550/arXiv.1707.01476.
- [31] Y. Zhang, Q. Yao and L. Chen, "Interstellar: Searching Recurrent Architecture for Knowledge Graph Embedding," *Advances in Neural Information Processing Systems*, vol. 33, 2020. DOI: 10.48550/arXiv.1911.07132.
- [32] Yongqi Zhang, Wenyuan Dai, Quanming Yao, and Lei Chen, "AutoSF: Searching Scoring Functions for
- [15] F. Bianchi, G. Rossiello, L. Costabello, M. Palmonari and P. Minervini, "Knowledge Graph Embeddings and Explainable AI," 2020. DOI: 10.3233/SSW200011.
- [16] Zhen Wang, Jianwen Zhang, Jianlin Feng and Zheng Chen, "Knowledge Graph Embedding by Translating on Hyperplanes," in *28th AAAI Conference on Artificial Intelligence*, 2014. DOI: 10.1609/aaai.v28i1.8870.
- [17] W. L. a. H. T. Xin Wang, "A Comparative Study of TransE and TransH Algorithms in Spatial Address Representation Learning," *Biomedical Journal of Scientific & Technical Research*, 2020. DOI: 10.26717/BJSTR.2020.26.004427.
- [18] Y. Lin, Z. Liu, Maosong Sun, Yang Liu and Xuan Zhu, "Learning Entity and Relation Embeddings for Knowledge Graph Completion," *29th AAAI Conference on Artificial Intelligence*, 2015. DOI: 10.1609/aaai.v29i1.9491.
- [19] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie and Jian Tang, "RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space," in *International Conference on Learning Representations (ICLR)*, 2019. DOI: 10.48550/arXiv.1902.10197.
- [20] L. Chao, Jianshan He, Taifeng Wang and Wei Chu, "PairRE: Knowledge Graph Embeddings via Paired Relation Vectors," 2021. DOI: 10.48550/arXiv.2011.03798.
- [21] L. Yu, Z. Luo, H. Liu, D. Lin, H. Li and Y. Deng, "TripleRE: Knowledge Graph Embeddings via triple Relation Vectors," arXiv, 2022. DOI: 10.48550/arXiv.2209.08271.
- [22] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao and Li Deng, "Embedding Entities and Relations for Learning and Inference in Knowledge Bases," in *International Conference on Learning Representations*, San Diego, 2015. DOI: 10.48550/arXiv.1412.6575.
- [23] F. Hitchcock, "The Expression of a Tensor or a Polyadic as a Sum of Products," 1927.

[42] Q. Wang, Z. Mao, B. Wang and L. Guo, "Knowledge Graph Embedding: A Survey of Approaches and Applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, 2017.

Knowledge Graph Embedding," in *International Conference on Data Engineering*, 2020. DOI: 10.48550/arXiv.1904.11682.

[33] "Leaderboards for Link Property Prediction," 2024. [Online]. Available: https://ogb.stanford.edu/docs/leader_linkprop/#ogbl-biogk.

[34] Y. Zhang, Q. Yao and J. Tin-Yau Kwok, "Bilinear Scoring Function Search for Knowledge Graph Learning," arXiv, 2021. DOI: 10.48550/arXiv.2107.00184.

[35] F. Hutter, L. Kotthoff and J. Vanschoren, *Automated Machine Learning: Methods, Systems, Challenges*, Springer Publishing Company, 2019.

[36] L. Guo, Z. Sun and W. Hu, "Learning to Exploit Long-term Relational Dependencies in Knowledge Graphs," in *36th International Conference on Machine Learning*, 2019.

[37] X. Luna Dong, E. Gabrilovich, G. Heitz, W. Horn and N. Lao, "Knowledge Vault: A Web-Scale Approach to Probabilistic Knowledge Fusion," *20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014. DOI: 10.48550/arXiv.1905.04914.

[38] M. Nickel, V. Tresp and H. P. Kriegel, "A three-way model for collective learning on multi-relational data," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, 2011. DOI: 10.1145/2623330.2623623.

[39] T. Shen, F. Zhang and J. Cheng, "A Comprehensive Overview of Knowledge Graph Completion," *Knowledge-Based Systems*, vol. 255, 2022. DOI: 10.5555/3104482.3104584.

[40] R. Socher, D. Chen, C. Manning and A. Ng, "Reasoning With Neural Tensor Networks for Knowledge Base Completion," *NeurIPS*, 2013. DOI: 10.1016/j.knosys.2022.109597.

[41] M. Wang, L. Qiu and X. Wang, "A Survey on Knowledge Graph Embeddings for Link Prediction," *Symmetry*, vol. 13, no. 3, 2021. DOI: 10.3390/sym13030485.

پیوست (الف)

ویژگی‌های روش TransE:

TransE با فرض اینکه جاسازی در فضای دوبعدی انجام شده باشد و نقطه الف (۱و۱) نقطه ب (۳و۳) و رابطه ج (۲و۲) امبد شده باشد، ویژگی‌های زیر را خواهد داشت:

۱. تقارن: اگر فرض کنیم که نقطه «الف» با نقطه «ب» رابطه «ج» را دارد، مسلماً اگر بخواهیم از نقطه «ب» نیز با بردار «ج» حرکت کنیم به نقطه‌ای بسیار دورتر از «الف» خواهیم رسید.

صحيح است $(1.1) + (2.2) = (3.3)$

با مختصات «الف» $(3.3) + (2.2) = (5.5)$

منطبق نیست

لذا این جاسازی، تقارن را پشتیبانی نمی‌کند. به عنوان مثال فرض شود رابطه «ج»، رابطه «همکاری» باشد، TransE نمی‌تواند جاسازی‌ها را طوری بسازد که کارمند اول با کارمند دوم رابطه همکاری داشته باشد و همزمان، کارمند دوم هم با کارمند اول، رابطه همکاری داشته باشد.

۲. همان‌طور که پیش‌تر گفته شد، هیچ‌وقت با جمع بردار انتقال رابطه و تالی، به نقطه سرآیند نمی‌رسیم. لذا این جاسازی پادتقارن را پشتیبانی می‌کند.

۳. آینه‌ای: اگر رابطه «ج» بین «الف» و «ب» وجود دارد، آیا می‌توانیم یک رابطه «د» را بین «ب» و «ج» تعریف و

۱. ویژگی‌های پادتقارن و آینه‌ای به مانند روش TransE در این روش قابل پشتیبانی هستند.

۲. یک به چند: همان‌طور که در بند اول گفته شد، می‌توان M را طوری تنظیم نمود که دو یا بیشتر نقاط در فضای مسئله با ضرب در M به فضایی بروند که همه آن‌ها در فضای جدید در یک مختصات قرار بگیرند. به همین دلیل می‌توان رابطه «یک به چند» را با روش TransR پیاده‌سازی نمود.

۳. تراگذاری: به دلیل ماهیت و روش پیاده‌سازی TransR، این ویژگی نمی‌تواند در جاسازی وجود داشته باشد، چرا که اصولاً هر رابطه در یک فضای مجزا محاسبه می‌شود و نمی‌توان دو رابطه را در یک فضای یکسان محاسبه نمود.

پیوست (ب)

با توجه به توضیحات بخش ۵ می‌توان نتیجه گرفت که قطعاً ویژگی تقارن در این جاسازی قابل پیاده‌سازی بوده و همین‌طور ویژگی پادتقارن در این جاسازی قابل پیاده‌سازی نخواهد بود. در ادامه، سایر ویژگی‌ها را بررسی می‌کنیم:

۱. رابطه یک به چند: اگر بخواهیم یک ویژگی، رابطه یک به چند را پشتیبانی کند باید بتوانیم داشته باشیم:

$$\langle h.r.t_1 \rangle = \langle h.r.t_2 \rangle$$

تساوی بالا، با تعیین درست مقادیر بردار ویژگی تمام پارامترها، برقرار خواهد بود. لذا رابطه «یک به چند» در این جاسازی قابل استفاده است.

۲. معکوس‌پذیری و تراگذاری: به دلیل ماهیت این جاسازی، واضح است که این ویژگی‌ها، توسط این جاسازی قابل پیاده‌سازی نیست. چون هر رابطه‌ای که سرآیند با تالی داشته باشد، همان رابطه نیز بین تالی و سرآیند برقرار خواهد بود.

مدل کرد؟ همان‌طور که از مثال عددی بالا مشخص است می‌توانیم بردار انتقال «د» را مساوی $(-2, -2)$ تعریف کنیم که با این تعریف می‌توانیم رابطه زیر را بازنویسی کنیم:

$$(1.1) = (-2, -2) + (3.3)$$

لذا این جاسازی، آینه‌ای را پشتیبانی می‌کند.

۴. برای بررسی تراگذاری، کافی است تصور کنیم که به ازای هر رابطه یک بردار انتقال داریم که ما را از نقطه سرآیند به نقطه تالی می‌رساند. لذا اگر تعدادی رابطه بی‌درپی داشته باشیم نهایتاً ما را به یک نقطه در فضای حالت می‌رساند که مسلماً با یک بردار انتقال می‌توانیم از نقطه سرآیند ابتدایی مستقیماً به این نقطه تالی برسیم. لذا این جاسازی تراگذاری را پشتیبانی می‌کند.

۵. شرط پشتیبانی از ویژگی «یک به همه» این است که اگر «A» رابطه «R» را با «B» داشته باشد، بتواند همین رابطه با گره دیگری به نام «C» داشته باشد. فرض کنیم که جاسازی زیر وجود داشته باشد:

$$A(1.1) + R(2.2) = B(3.3)$$

حال اگر بخواهیم همین رابطه را با یک نقطه دیگر داشته باشیم خواهیم داشت:

$$A(1.1) + R(2.2) = C(3.3)$$

که در این صورت دو نقطه «B» و «C»، کاملاً یکسان هستند و عملاً می‌بایست یک نقطه باشند که این در

گراف دانش صحیح نیست. لذا این جاسازی رابطه یک به چند را پشتیبانی نمی‌کند.

ویژگی‌های روش TransR:

تقارن: می‌توان M را طوری تعیین کرد که دو نقطه «الف» و «ب» با ضرب در M ، به یک نقطه یکسان در فضای جدید تبدیل شوند، در نتیجه در فضای جدید هر دو نقطه رابطه یکسان با یکدیگر دارند و نهایتاً می‌تواند ویژگی تقارن را پشتیبانی کند.

پیوست (ج)

۲) رسیدگی به تقسیم‌بندی مختلف: برای اینکه پارامترهای آموزش را پایدار کنیم، از جاسازی‌های با مقدار حقیقی دوبعدی برای نمایش Analogy و Simple استفاده می‌کنیم. در Analogy جاسازی‌ها به یک قسمت حقیقی $\hat{h} \in \mathbb{Z}^d$ و یک قسمتی مختلط $\check{h} \in \mathbb{Z}^d$ تقسیم‌بندی می‌شوند که می‌توان آن را به صورت بردار حقیقی به هم پیوسته $[\overline{h_{re}}, \overline{h_{im}}] \in \mathbb{Z}^d$ شبیه به Complex نشان داد. تابع امتیاز نیز به صورت زیر تقسیم می‌شود:

$$\langle \hat{h}, \hat{r}, \hat{t} \rangle = \text{Re}(\langle \check{h}, \check{t}, \text{conj}(\check{t}) \rangle)$$

در Simple، دو بردار جاسازی مستقل $\hat{h} \in \mathbb{Z}^d$ و $\check{h} \in \mathbb{Z}^d$ برای نمایش هر موجودیت و رابطه استفاده شده است. تابع امتیاز حاصله به صورت زیر است:

$$\langle \hat{h}, \hat{r}, \hat{t} \rangle + \langle \check{h}, \check{r}, \check{t} \rangle$$

۱) رسیدگی به جاسازی‌های مختلط: یک بردار مختلط $v \in \mathbb{C}^d$ با $v = v_{re} + i v_{im}$ ، از یک قسمت حقیقی $v_{re} \in \mathbb{Z}^d$ و یک قسمتی موهومی $v_{im} \in \mathbb{Z}^d$ تشکیل شده است. برای رسیدگی به جاسازی‌های مختلط، می‌توانیم از بردار حقیقی $2d$ بُعدی $[v_{re}, v_{im}]$ برای نمایش بردار مختلط d بُعدی v استفاده کنیم. فرض کنید جاسازی مختلط $h_{re}, h_{im} \in \mathbb{Z}^d$ که $h = h_{re} + i h_{im}$ (همچنین برای r, t) سپس Complex را می‌توان به صورت زیر توصیف کرد:

$$\begin{aligned} \text{Re} \langle h, r, \text{conj}(t) \rangle &= \langle h_{re}, r_{re}, t_{re} \rangle + \\ &\langle h_{im}, r_{re}, t_{im} \rangle + \\ &\langle h_{re}, r_{im}, t_{im} \rangle + \\ &\langle h_{im}, r_{im}, t_{re} \rangle \end{aligned}$$

به طور مشابه، جاسازی‌های دوبعدی را نیز می‌توان با $[v_{re} + v_{im}]$ نشان داد و به صورت دو قسمتی نمایش داد:

$$\langle h, r, t \rangle = \langle h_{re}, r_{re}, t_{re} \rangle + \langle h_{im}, r_{im}, t_{im} \rangle$$

- 17 Academic Community Exploration Knowledge Graph
- 18 TDM- Translational Distance Models
- 19 Translation invariance
- 20 SGD-Stochastic Gradient Descent
- 21 Hadamard Product
- 22 expressive
- 23 BLM-Bilinear Models
- 24 Embed
- 25 Fact
- 26 Quaternions
- 27 Expressive
- 28 Analogy
- 29 Deep Neural Networks methods
- 30 Convolutional Neural Networks methods
- 31 Recurrent Neural Networks methods
- 32 Mean Reciprocal Rank
- 33 Document

- 1 Embedding
- 2 Knowledge Graph
- 3 Question-Answering systems
- 4 Recommender system
- 5 Knowledge Graph Embedding
- 6 Embedding
- 7 Score Function
- 8 Score function based
- 9 Translational distance models
- 10 Bilinear models
- 11 Neural network based models
- 12 Triple

^{۱۳} شامل دو گراف دانش بزرگ مرتبط به ویکی‌پدیا و داده‌های بیولوژیکی

هستند که در https://ogb.stanford.edu/docs/dataset_overview/ قابل دسترسی است.

- 14 FreeBase
- 15 BIOKG
- 16 WIKIKG2