دانشگاه تهران
پردیس دانشکده های فنی
دانشکده مهندسی کامپیوتر

بهبود مدل‌های پیش‌بینی پیوند مبتنی بر ویژگی‌های پنهان با استفاده از قوانین انجمنی

پایان‌نامه جهت دریافت درجه کارشناسی ارشد

در رشته مهندسی کامپیوتر گرایش نرم‌افزار

نگارش:

مسعود هاشمیان

استاد راهنما:

دکتر ناصر یزدانی

مرداد ماه ۱۳۹۶


بسم الله الرحمن الرحيم

دانشگاه تهران

پردیس دانشکده های فنی

دانشکده مهندسی کامپیوتر

بهبود مدل‌های پیش‌بینی پیوند مبتنی بر ویژگی‌های پنهان با استفاده از قوانین انجمنی

پایان‌نامه جهت دریافت درجه کارشناسی ارشد

در رشته مهندسی کامپیوتر گرایش نرم‌افزار

نگارش:

مسعود هاشمیان

استاد راهنما:

دکتر ناصر یزدانی

مرداد ماه ۱۳۹۶

دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

گواهی دفاع از پایان‌نامه کارشناسی ارشد

هیأت داوران پایان‌نامه کار شنا سی ار شد آقای / خانم .....به شماره دانشـــجویی.....در رشـــته ..... گرایش ..... را در تاریخ ..... با عنوان .....

	به عدد	به حروف
با نمره نهایی		
و درجه		ارزیابی کرد.

ردیف	مشخصات هیئت داوران	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا مؤسسه	امضاء
۱	**استاد راهنما** **اســـتاد راهنمای دوم (حســـب مورد)**				
۲	**استاد مشاور**				
۳	**استاد داور داخلی**				
٤	**استاد مدعو**				
٥	**نـ مایـ نده کمیـ ته تحصـــیلات تکمیلی دانشکده / گروه**				

نام و نام خانوادگی معاون تحصیلات تکمیلی و پژوهشی دانشکده / گروه:

تاریخ و امضاء:

نام و نام خانوادگی معاون آموزشی و تحصیلات تکمیلی پردیس دانشکده‌های فنی:

تاریخ و امضاء:

تعهدنامه اصالت اثر

باسمه تعالی

اینجانب مسعود ها شمیان تائید می‌کنم که مطالب مندرج در این پایان‌نامه حا صل کار پژوهشی اینجانب ا ست و به د ستاوردهای پژوه شی دیگران که در این نو شته از آنها ا ستفاده شده ا ست مطابق مقررات ارجاع گردیده اســت. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم ســطح یا بالاتر ارائه نشده است.

نام و نام خانوادگی دانشجو :

امضای دانشجو :

تقدیم به پدر، مادرم و همسر عزیزم

چکیده

یادگیری ماشین برای رابطه‌های بین داده‌هایی حجیمی که روزانه توسط بازیابی اطلاعات، محاسبات زیست شناسی، پردازش زبان طبیعی و... تولید می‌شوند، به مساله‌ی سختی تبدیل شده است. روش‌های زیادی برای کار با این داده‌ها معرفی شده است که پایگاه‌های دانش بزرگ را تحلیل و روابط موجود در آن‌ها را استخراج می‌کند. یک دسته از این روش‌ها روش‌های مبتنی بر ویژگی‌های پنهان است که مساله را وارد یک فضای برداری چند بعدی کرده و پس از آن سعی می‌کنند با کاهش تعداد پارامتر، مساله را حل کنند. قدرت این روش‌ها در سادگی آموزش مدل، تعداد کم پارامترهایی که نیاز است آموزش دیده شوند و کار روی پایگاه‌های دانش با مقیاس زیاد است. این مدل‌ها روابط موجود بین موجودیت‌ها را به خوبی تشخیص داده و پس از آموزش به دسته‌بندی خوبی از این موجودیت‌ها دست پیدا می‌کنند اما در مورد پیوندهای بین موجودیت‌ها اینطور نیست و روابط بین پیوندها به‌خوبی تشخیص داده نمی‌شود. در این پژوهش قصد داریم به کمک قوانین انجمنی روابط بین پیوندها را نیز تشخیص دهیم و در آموزش مدل دخیل کنیم. به کمک راه‌حل ارائه شده در این پژوهش توانستیم روش‌های رسکال، NTN، Structured Embedding، TransE و TransH بهبود داده و این بهبود را توسط دو معیار کیفیت اثبات نموده‌ایم. به‌علاوه چارچوبی ارائه داده‌ایم که به کمک آن می‌توان نقاط ضعف و قوت روش‌های موجود را بررسی کرد. علاوه بر بهبود روش‌های موجود، این راه‌حل منجر به استخراج حقیقت‌های پنهان در پایگاه‌های دانش شده و می‌توانیم یادگیری مدل‌ها را روی پایگاه‌های دانش کامل‌تری انجام دهیم.

واژه‌های کلیدی: یادگیری ماشین، پیش‌بینی پیوند، ویژگی‌های پنهان، فضای برداری، قوانین انجمنی

فهرست مطالب

فهرست اشکال

فهرست جداول

فصل ۱: مقدمه

در سال‌های اخیر شاهد رشد بسیار زیادی در شبکه‌های اجتماعی بوده‌ایم و مطالعات زیادی روی این شبکه‌ها انجام گرفته است. داده‌های شبکه‌های اجتماعی یکی از ابزارهای محبوب برای مدل کردن رابطه و رفتار افراد و جامعه یا گروهی که در آن عضو هستند به‌شمار می‌رود. این داده‌ها معمولا به صورت گرافی نمایش داده می‌شود که در آن گره‌ها افراد و یال‌ها روابط بین این افراد می‌باشد.

هم‌چنین یادگیری ماشین مدت زیادی است که در علوم کامپیوتر جایگاه خود را پیدا کرده و به عنوان ابزار قدرمتندی برای کمک به انسان در زمینه‌های مختلف محسوب می‌شود و ماشین را بیش از پیش در خدمت انسان در آورده است.

در روش‌های اولیه یادگیری ماشین، عمدتا از داده‌ها و متن‌های خام در زمینه‌ی یادگیری استفاده می‌شد. اخیرا از طرف برخی شرکت‌ها و موسسات بزرگ، همچون گوگل، ای‌بی‌ام، مایکروسافت و... پایگاه‌های دانشی معرفی شده است که انجام راهکارهای مختلف یادگیری ماشین را ساده‌تر و کاربردی‌تر کرده است. در این پایگاه‌های دانش، اطلاعات مورد نیاز برای عملیات‌های مختلف یادگیری به صورت منظم و نیمه‌منظم موجود است و دغدغه‌ی نرمال کردن و رفع خطا و استخراج حقایق را به مقدار زیادی کم کرده است. این پایگاه‌های دانش عمدتا به صورت یک شبکه از موجودیت‌ها و روابط بین آن‌ها است که می‌توان آن را به صورت یک گراف داده نمایش داد. ساختار کلی گراف به این صورت است که گره‌ها نشانگر موجودیت‌ها و یال‌های بین گره‌ها نشانگر روابط بین آن‌ها باشند. این یال‌ها می‌توانند از یک نوع باشند یعنی گراف نشانگر یک شبکه تک-رابطه‌ای باشد (همگون) یا هر یال با یک برچسب، نشانگر نوع رابطه باشد و شبکه نشانگر یک گراف چند رابطه‌ای (ناهمگون) باشد.

یکی از مسائل بنیادی و اساسی در یادگیری ماشین روی شبکه‌های اجتماعی، پیش‌بینی پیوند در شبکه‌های همگون و ناهمگون است به این معنی که از اطلاعات موجود در گراف دانش استفاده کرده و وجود یا عدم وجود یک یال را پیش‌بینی کرد، یا به عبارت دیگر مساله‌ی پیش‌بینی پیوند این است که نمایی از یک شبکه به ما داده می شود و ما مایل هستیم که بدانیم در آینده نزدیک، احتمالا چه تراکنش‌هایی میان اعضای فعلی شبکه روی می‌دهد و یا این‌که کدام یک از تراکنش‌های موجود را از دست می‌دهیم.

این راهکار در زمینه های مختلف یادگیری ماشین مورد استفاده قرار می‌گیرد و کاربرد گسترده‌ای در زندگی انسان پیدا کرده است. برای مثال از این راهکار در سیستم‌های توصیه‌گر در فروشگاه‌های اینترنتی، سیستم‌های تشخیص پزشکی، جواب‌گویی به سوال و ... استفاده می‌شود.

اکثر مطالعات انجام شده در این زمینه روی شبکه‌های تک-رابطه‌ای بوده است. به این معنا که روابط بین موجودیت‌ها از یک نوع است و این روابط به صورت دوتایی‌های مرتب مورد استفاده قرار می‌گیرند. برای مثال اگر در یک شبکه اجتماعی رابطه را دوستی بین افراد در نظر بگیریم یال‌های گراف شبکه به صورت «الف، ب» خواهد بود به این معنی که شخص «الف» با شخص «ب» رابطه دوستی دارد.

پایگاه‌های دانشی که اخیرا معرفی شده‌اند عمدتا داده‌ها را به صورت داده‌های چند-رابطه‌ای ذخیره می‌کنند و اطلاعات بیشتری از یک رابطه دوتایی بلی یا خیر به ما می‌دهند. منظور از داده‌های چند رابطه‌ای گراف جهت‌داری است متشکل از موجودیت‌ها و روابط بین آن‌ها که بصورت «مبدا h، رابطه r، مقصد t» نمایش داده می‌شود، به این معنی که یک رابطه r بین موجودیت‌های h و t وجود دارد. برای مثال سه‌تایی «تهران، واقع در، ایران» این اطلاع را به ما می‌دهد که استان تهران داخل کشور ایران قرار دارد. در این نوع پایگاه دانش هم انواع مختلف موجودیت وجود دارد و هم انواع مختلف رابطه بین موجودیت‌ها. پایگاه‌های دانشی مانند YAGO[1]، Freebase[2]، DBpedia[3]، Google Knowledge Graph[4] و WordNet[5] وجود دارند که شامل تعداد زیادی نمونه چندرابطه‌ای می‌باشند و تعداد زیادی موجودیت و روابط فی ما بین را می‌توان در آن‌ها یافت و از آن برای یادگیری مدل استفاده کرد.

۱-۱- تعریف مسئله

روش‌های مختلفی برای حل مساله‌ی پیش‌بینی پیوند در پایگاه‌های دانش ناهمگون ارائه شده است که از رویکردهای مختلفی سعی به حل این مساله می‌کنند؛ از جمله روش‌های آماری، روش‌های ویژگی‌های پنهان، روش‌های ویژگی‌های گراف و ... [6]. تمرکز ما در این مقاله بر روی روش‌های پیش‌بینی پیوند مبتنی بر ویژگی‌های پنهان خواهد بود.

در روش‌های مبتنی بر ویژگی‌های پنهان، با استفاده از ویژگی‌هایی که در موجودیت‌ها و روابط بین آن‌ها وجود دارد سعی می شود میزان ارتباط بین موجودیت‌ها را تشخیص دهیم و به این صورت وجود یک پیوند را تایید یا رد کنیم [7]. برای مثال اگر دو شخص با هم همکار هستند، به احتمال زیادی ویژگی‌های مشترکی دارند، مثلا هر دو اهل یک شهر هستند، هر دو در یک رشته‌ی دانشگاهی تحصیل کرده‌اند، خصوصیات اخلاقی یکسانی دارند و .... در رابطه‌ی همکار بودن به هیچ یک از این ویژگی‌ها به طور مستقیم اشاره نشده است و این ویژگی‌ها به طور ضمنی در این موجودیت‌ها قرار دارند که با استفاده از آن می‌توانیم وجود یا عدم وجود رابطه‌ی همکار بودن را حدس بزنیم. پس هر موجودیت می‌تواند تعداد زیادی ویژگی پنهان داشته باشد که رابطه‌ها به این ویژگی‌ها وزن می‌دهند، مثلا در رابطه‌ی همکاری احتمال این‌که رشته‌ی تحصیلی دو شخص در همکار شدن آن‌ها تاثیرگذار باشد بیشتر از ویژگی رنگ پوست دو شخص است، پس وزن ویژگی رشته‌ی تحصیلی در این رابطه بیشتر از ویژگی رنگ پوست است.

روش‌های زیادی برای حل مساله‌ی پیش‌بینی پیوند مبتنی بر ویژگی‌های پنهان ارائه شده است. همه‌ی این روش‌ها از یک روال ثابت برای حل مساله استفاده می‌کنند و هر کدام با نوآوری‌هایی که داشته‌اند بهبودهایی در نتایج بدست‌آمده حاصل کرده‌اند. این روش‌ها در بخش ۵-۲- به‌طور کامل معرفی خواهند شد.

در این پژوهش قصد داریم که با استفاده از قوانین انجمنی موجود در پایگاه‌های دانش به دو هدف برسیم:

۱. بهبود نتایج در روش‌های معرفی شده‌ی موجود: در ادامه پنج روش از روش‌های معروف مبتنی بر ویژگی‌های پنهان که به حل مساله‌ی پیش‌بینی پیوند پرداخته‌اند را انتخاب می‌کنیم و در بخش ۱-۴- نشان می‌دهیم که استفاده از قوانین انجمنی در فرایند

آموزش این پنج روش باعث بهبود در نتایج این روش‌ها خواهد شد.

۲. ارائه‌ی چارچوبی برای سنجش نقاط ضعف و قوت روش‌های ارائه شده: هم‌چنین با دسته‌بندی قوانین انجمنی در بخش ۳-۳- نشان می‌دهیم که با اعمال جداگانه‌ی دسته های مختلف قوانین انجمنی می‌توانیم میزان قدرت و ضعف روش ها را در قانون‌های مختلف بررسی کنیم که با برطرف کردن نقاط ضعف می‌توان روش‌های قوی‌تری ارائه کرد.

۲-۱- روش انجام پژوهش

برای دست‌یابی به اهدافی که در بخش قبل مطرح شد، همان‌طور که اشاره شد از قوانین انجمنی استفاده خواهیم کرد. در روش‌های معرفی شده فقط از روابط موجود در پایگاه دانش در فرایند آموزش استفاده شده است. مثلا اگر در پایگاه دانش داشته باشیم که «علی، پدر، حسن» به این معنی که علی پدر حسن است می‌توانیم به این نتیجه برسیم که «حسن، فرزند، علی» و از آن در فرایند آموزش استفاده کنیم، اما در این روش‌ها این روابط مورد استفاده قرار نگرفته‌اند. در این پژوهش قصد داریم که این‌گونه قوانین را به کمک روش‌هایی که در بخش ۲-۳- توضیح داده می‌شوند از پایگاه دانش استخراج کنیم و به کمک معیارهای کیفیت که در بخش ۱-۳- معرفی می‌شوند قوانینی که کیفیت مناسب دارند را انتخاب کنیم و در فرایند آموزش از آن‌ها استفاده کنیم، نحوه‌ی استفاده از این قوانین در مدل‌های موجود در بخش ۳-۳- توضیح داده شده است.

۳-۱- کاربردهای پژوهش

مسائل پیش‌بینی پیوند در زمینه‌های زیادی قابل به‌کارگیری هستند و همین امر باعث شده است که در سال‌های گذشته بسیار مورد توجه و تحقیق قرار بگیرند. به عنوان مثال در ادامه تعدادی از این کاربردها را مطرح خواهیم کرد:

- پیش‌بینی پیوندهای احتمالی در شبکه‌های اجتماعی؛ برای مثال در شبکه‌های اجتماعی بین کاربران و مطالب ثبت شده، نظرات و ... بررسی شود که روابطی مانند دوستی، پسندیدن و

نپسندیدن، روابط فامیلی و... وجود دارد یا خیر.

- استفاده به‌عنوان سیستم‌های توصیه‌گر؛ برای مثال کاربران و کالاها یا اشیاء موجودیت‌ها هستند و خریدن، امتیاز دادن، بررسی کردن و ... رابطه‌ها هستند که می‌توان از بررسی این روابط و موجودیت‌ها اطلاعاتی نظیر «کاربر x کالای y را خواهد خرید» یا خیر یا این‌که «کالای y به تعداد بالا فروش خواهد رفت» یا خیر بدست آورد.

- استفاده در سیستم‌های تشخیص پزشکی؛ برای مثال شبکه‌ی بین بیماران، بیماری‌ها، داروها و... را در نظر بگیرید، با بررسی دقیق این شبکه می‌توان علل و درمان‌های بیماری‌ها را به کمک ماشین بدست آورد.

علاوه بر کاربردهایی که در بالا برای پیش‌بینی پیوند اشاره شد، از روش ارائه شده در این پژوهش می‌توان برای کامل‌تر کردن پایگاه‌های داده نیز استفاده کرد، به این صورت که روابطی که از قوانین انجمنی استخراج شده از پایگاه دانش بدست می‌آید و کیفیت لازم را دارد را نیز به پایگاه دانش اضافه و آن را کامل‌تر کنیم.

۴-۱- ساختار پایان‌نامه

ادامه‌ی این پایان‌نامه به این صورت است که در فصل ۲ پیشینه پژوهش و زمینه‌های تحقیق را شرح و توضیح می‌دهیم از چه ابزاری برای انجام این پژوهش استفاده شده است و هم‌چنین روش‌های مختلف حل این مساله را بیان می‌کنیم. در فصل ۳ مروری داریم بر قوانین انجمنی و معیارهایی برای سنجش کیفیت قوانین معرفی می‌کنیم و نحوه‌ی استخراج قانون‌ها از پایگاه‌های دانش را توضیح می‌دهیم و در انتهای این فصل پنج قانون هدف، که قصد استفاده از آن‌ها را داریم، معرفی می‌کنیم. در فصل ۴ روش پیشنهادی برای بهبود روش‌های مبتنی بر ویژگی پنهان را شرح می‌دهیم و نحوه‌ی اعمال آن بر روی همه‌ی روش‌های معرفی شده در فصل ۲ را توضیح می‌دهیم. در فصل ۵ آزمایش‌ها و نتایج آزمایش‌های انجام شده را آورده‌ایم و نهایتا در فصل ۶ نتیجه‌گیری انجام شده و کارهایی که در آینده برای بهبود بیشتر این روش‌ها قابل انجام هست توضیح داده خواهد شد.

فصل ۲: پیشینه پژوهش

در این بخش به معرفی روش‌های موجود که به حل مساله‌ی پیش‌بینی پیوند در شبکه‌های ناهمگون مربوط می‌شوند، خواهیم پرداخت. روش‌هایی با راهکارهای مختلف در حوزه‌های مختلف تلاش به حل این مساله کرده‌اند. این روش‌ها می‌توانند به سه دسته تقسیم می‌شوند: ۱- روش‌های مبتنی بر ویژگی‌های گراف [8] ۲- روش‌های مدل تصادفی مارکوف [9] ۳- روش‌های مبتنی بر ویژگی‌های پنهان [6]. در ادامه توضیح مختصری در مورد هر دسته از روش‌ها می‌دهیم.

- در روش‌های مبتنی بر ویژگی‌های گراف از روی ویژگی‌های ساختاری گراف داده‌ها استفاده می‌شود مانند دسته‌بندی گره‌ها، دسته‌بندی نوع یال‌ها، تعداد گره‌های مشابه و ...
- روش‌های مدل تصادفی مارکوف که در آن دید بالایی از گراف داده نداشته و سعی می‌کنیم مساله را به صورت محلی حل کنیم به این صورت که روابط هر موجودیت را با موجودیت‌های اطراف آن بررسی می‌کنیم و جواب‌های محلی را بدست می‌آوریم.
- روش‌های مبتنی بر ویژگی‌های پنهان که در این روش‌ها هر موجودیت و نوع رابطه بین آنها به صورت برداری از ویژگی‌های پنهان تعریف می‌شود که ویژگی‌های پنهان نام دارد. برای مثال ویژگی‌هایی که یک موجودیت می‌تواند داشته باشد، محل به وجود آمدن آن، سن آن، جاندار یا بی‌جان بودن آن و ... است.

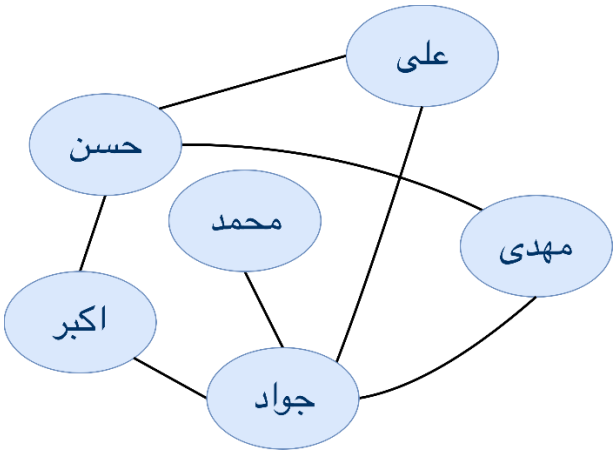
همان‌طور که در بخش ۲-۱-۲-گفتیم تمرکز ما در این تحقیق روی روش‌های مبتنی بر ویژگی‌های پنهان است. در ادامه این روش‌ها را به صورت کامل توضیح داده و تعدادی از آن‌ها را به طور مختصر معرفی کرده و نحوه‌ی کار آن‌ها و نتایج به‌دست آمده از آن‌ها را توضیح می‌دهیم.

۱-۲- زمینه‌های تحقیق

در این بخش مطالبی راجع به مباحث پایه‌ای که در ادامه تحقیق از آن‌ها استفاده شده است را مطرح خواهیم کرد.

۱-۱-۲- شبکه‌های همگون و ناهمگون

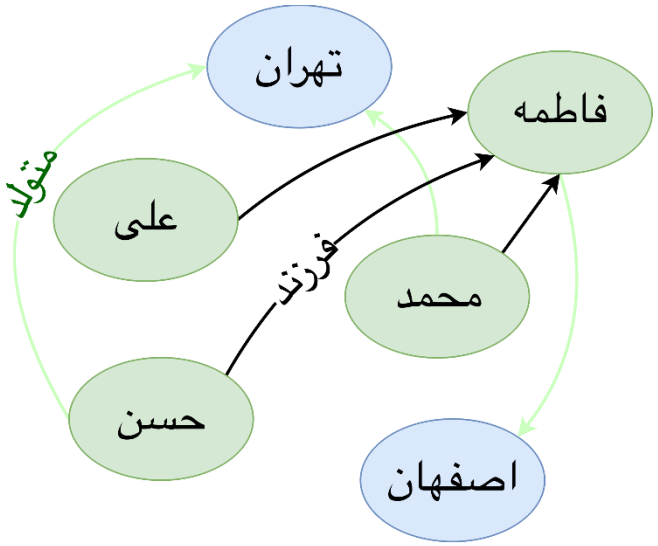
در سال‌های اخیر شبکه‌های اجتماعی پیشرفت زیادی داشته است و در زمینه‌های مختلفی شبکه‌سازی شده است. عمدتا در این شبکه‌ها روابط خاصی مدنظر و قابل استخراج است، مثل روابط دوستی، همکاری و... که اگر گراف این گونه شبکه‌ها را رسم کنیم یال‌ها از یک نوع است و نشانگر یک رابطه‌ی خاص است. مثلا گرافی هست که همه‌ی گره‌های آن انسان‌ها هستند و یال‌های بین گره‌ها نشان‌دهنده‌ی وجود یا عدم وجود دوستی بین اشخاص است. این گونه شبکه‌ها که در آن‌ها یال و گره‌ها از یک نوع است را شبکه‌های همگون می‌نامیم [10]. در شکل ۱ یک نمونه شبکه‌ی همگون که در آن اشخاص و رابطه‌ی دوستی بین آن‌ها به تصویرکشیده شده است را مشاهده می‌کنیم.



شکل ۱ شبکه‌ی همگون

اما همان‌طور که در مقدمه نیز اشاره شد، اخیرا شبکه‌های بزرگتر و پیچیده‌تری معرفی شده است که فقط یک نوع رابطه را پوشش نمی‌دهد و روابط مختلفی را بین موجودیت‌های مختلف در بر می‌گیرد، به این شبکه‌ها به علت یکسان نبودن نوع روابط و موجودیت‌ها شبکه‌های ناهمگون می‌گوییم، هم‌چنین به خاطر وجود دانشی که در این شبکه‌ها نهفته و قابل استخراج است، آن را پایگاه دانش[1] نیز می‌نامیم. در شکل ۲ یک قسمت کوچک از یک شبکه‌ی ناهمگون را مشاهده می‌کنیم که موجودیت‌های آن از دو نوع انسان و شهر هستند و روابط موجود در این شبکه از دو نوع «والد بودن» و «متولد شهر ... بودن» است.

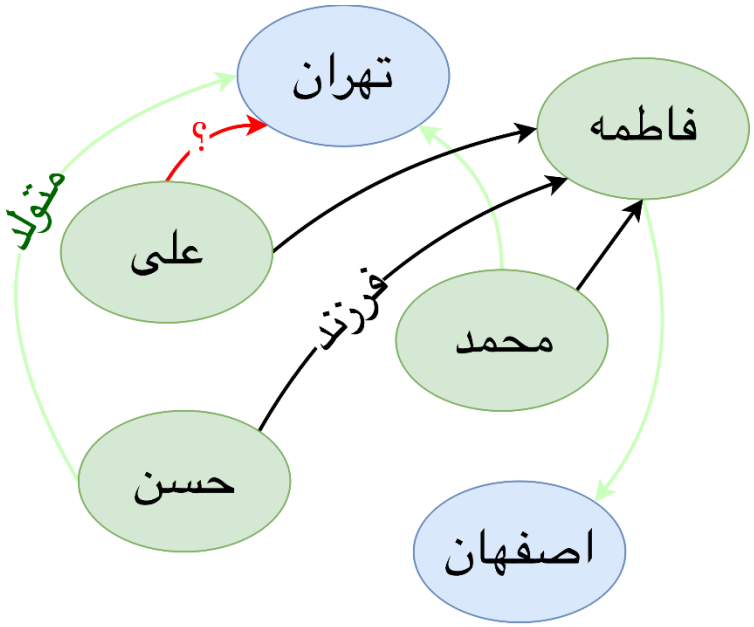
[1] Knowledgbase



شکل ۲ شبکه ناهمگون

۲-۱-۲- پیش‌بینی پیوند

یکی از روش‌های یادگیری ماشین که در زمینه‌های مختلف به کمک انسان آمده است و کارهای انسانی را تسهیل کرده است پیش‌بینی پیوند است. در پیش‌بینی پیوند، یک گراف از روابط بین موجودیت‌ها را به عنوان ورودی مساله دریافت می‌کنیم و وجود یا عدم وجود یک یال بین دو موجودیت را پیش‌بینی می‌کنیم [11]. در شکل ۳ یک گراف از یک شبکه‌ی ناهمگون را مشاهده می‌کنیم. مساله‌ی پیش‌بینی پیوند تلاش می‌کند که بررسی کند که رابطه‌ی «تولد» بین گره‌های «علی» و «تهران» قرار دارد یا خیر. این پیش‌بینی عمدتا از روی دیگر روابط بین موجودیت‌ها انجام می‌شود و با بررسی شباهت‌ها و معیارهایی که در ادامه‌ی این پژوهش توضیح خواهیم داد تصمیم می‌گیریم که این پیوند برقرار هست یا خیر.



شکل ۳ پیش‌بینی پیوند در شبکه ناهمگون

راهکارهای مختلفی برای حل این م ساله مطرح شده ا ست که به طور کلی می‌توان این راهکارها را به سه دسته‌ی ۱- روش‌های مبتنی بر ویژگی‌های گراف؛ ۲- روش‌های مدل تصادفی مارکوف؛ و ۳- روش‌های مبتنی بر ویژگی‌های پنهان تقسیم کرد که در این پژوهش تمرکز ما روی دسته‌ی سوم یعنی روش‌های مبتنی بر ویژگی‌های پنهان خواهد بود و در بخش ۴-۲- این روش‌ها را به تفصیل توضیح خواهیم داد.

۳-۱-۲- فرض جهان‌باز[1] و فرض جهان‌بسته[2]

فرض‌های جهان‌باز و جهان‌بسته در سیستم ر سمی منطق[3] برای بازنمایی دانش[4] مورد ا ستفاده قرار می‌گیرد. در فرض جهان بسته در نظر می‌گیریم که اگر داده‌ای در پایگاه دانش نبود، آن داده را غلط

[1] Open world assumption (OWA)

[2] Closed world assumption (CWA)

[3] Formal system of logic

[4] Knowledge representation

فرض می‌کنیم[12]. برای مثال اگر در پایگاه دانش حقیقت «x, r, y» که به معنی این ا ست که x با y رابطه‌ی r را دارد وجود نداشـــته باشـــد، می‌توانیم در نظر بگیریم که این حقیقت اشـــتباه اســـت و مطمئنیم که x با y رابطه‌ی r ندارد.

اما در فرض جهان‌باز این‌گونه نیســت و اگر حقیقتی در پایگاه دانش وجود نداشــته باشــد نمی‌توانیم با اطمینان بگوییم که آن حقیقت ا شتباه ا ست، و ممکن ا ست صحیح با شد [12] . وجود این فرض از کامل نبودن پایگاه‌های دانش ناشـــی می‌شـــود که وقتی نمی‌توانیم همه‌ی اطلاعات موجود در مورد موضـــوع مربوط به پایگاه دانش را جمع‌آوری و در آن قرار دهیم پس نمی‌توانیم در مورد حقیقت‌هایی که در پایگاه دانش نیست نظری بدهیم و این حقیقت را ناشناس[1] در نظر می‌گیریم.

۴-۱-۲- گرادیان نزولی تصادفی[2] (SGD)

گرادیان نزولی تصـــادفی یک راهکار ســـاده و در عین حال موثر برای مســـائل کمینه‌ســـازی یا بیشینه سازی تابع هدف ا ست که به صورت مجموع روی یک تابع مشتق‌پذیر نو شته می شود. در کل SGD سعی می‌کند نقطه‌ی کمینه یا بیشینه را به کمک تکرار پیدا کند. این روش می‌تواند به صورت دســـته‌ای روی پایگاه‌های دانش بزرگ اعمال شـــود که مشـــکل حافظه برای عملیات یادگیری روی پایگاه‌های دانش بزرگ که در حافظه‌ی ا صلی ما شین جا نمی‌گیرند را حل می‌کند. هم‌چنین د سته‌ای بودن این روش این قابلیت را به ما می‌دهد که داده‌های جدیدی که به صورت برخط به مجموعه اضافه می‌شـوند را نیز بتوانیم وارد فرایند آموزش کنیم و نیاز به اجرای مجدد همه‌ی مراحل آموزش نباشـد. نکته روش گرادیان تصادفی نزولی این است که نیاز نیست در هر تکرار کل مجموعه داده مورد بررسی قرار بگیرد و این روش با یک یا چند نمونه از مجموعه‌ی داده در هر تکرار قابل اجرا است [13].

بروزرسـانی پارامتر بصـورت زیر انجام می‌پذیرد که در آن Q تابع هدف، w پارامتری که با تغییر آن قصد داریم تابع هدف را بهینه کنیم و η نرخ یادگیری است.

$$w = w - \eta \nabla E[Q(w)] = w - \eta \sum_{i=1}^{n} \frac{\nabla Q_i(w)}{n}$$

[1] unknown

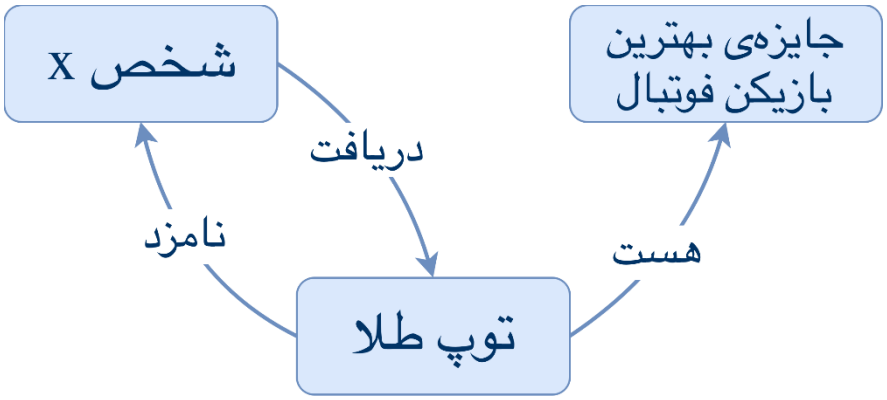
[2] Stochastic gradient descent

۲-۲- روش‌های مبتنی بر ویژگی‌های پنهان

روش‌های مبتنی بر ویژگی‌های پنهان از جدید ترین راهکارهایی است که در حوزه‌ی پیش‌بینی پیوند روی پایگاه‌های دانش موجود معرفی شده است. همان‌طور که در بخش قبل گفتیم این روش از ویژگی‌هایی که در نگاه اول از پایگاه دانش برداشت نمی‌شود استفاده می‌کند که ویژگی‌های پنهان نام دارند. در همه‌ی روش‌های مبتنی بر ویژگی پنهان این ویژگی‌ها را به صورت برداری تعریف می‌کنیم که هر مولفه از این بردار نشان دهنده‌ی یک ویژگی می‌باشد.

برای مثال در رابطه‌ی دوستی فاکتورهایی تاثیرگذار هستند و اگر در پایگاه دانش همه‌ی رابطه‌های دوستی موجود را بررسی کنیم به یک بردار از فاکتورها می‌رسیم که بردار رابطه‌ی دوستی را تشکیل می‌دهد. مثلا فاکتورهایی مانند شهر محل زندگی، سن، دانشگاه، رشته‌ی دانشگاهی، جنسیت، مذهب و ... در شکل گیری رابطه‌ی دوستی می‌تواند موثر باشد، اما این‌که هر کدام از این روابط چقدر در ایجاد رابطه‌ی دوستی تاثیر دارند و اهمیت هر کدام چقدر است و این میزان اهمیت را چگونه در تشخیص این رابطه تاثیر دهیم به مدل یادگیری بستگی دارد که در ادامه در معرفی هر یک از روش‌ها به صورت کامل توضیح داده خواهد شد.

برای مثال برای این‌که بررسی کنیم شخص X بازیکن فوتبال خوبی هست یا خیر از دیگر روابط موجود استفاده می‌کنیم و میزان ارتباط این شخص را با معیارهای بازیکن خوب فوتبال بودن بررسی می‌کنیم. در مثال شکل ۴ شخص x هم برای توپ طلا نامزد شده و هم توپ طلا را دریافت کرده و از اطلاعات دیگر پایگاه دانش می‌دانیم که توپ طلا به بهترین بازیکن فوتبال هر سال داده می شود. پس شخصی که این جایزه را دریافت کرده بازیکن فوتبال خوبی است.



شکل ۴ کاربرد قوانین انجمنی در پیش‌بینی پیوند

روش‌های زیادی برای مدل‌های مبتنی بر ویژگی‌های پنهان معرفی شـده اسـت، روش‌هایی که در این تحقیق مورد مطالعه و بررسی قرار گرفته این به صورت زیر دسته‌بندی می‌شوند.

- روش‌های خطی
- روش‌های دو خطی
  - رسکال
- روش‌های ادراک چندلایه‌ای
  - Neural Tensor Network (NTN)
- روش‌های فاصله‌ی پنهان
  - Structured Embedding
  - Translating Embedding (TransE)
  - Translating on Hyperplane (TransH)

در ادامه ابتدا پایگاه دانش Freebase که مطالعات روی آن انجام می‌گیرد و نحوه‌ی تقسیم بندی آن را توضیح خواهیم داد. سپس روش کلی آموزش مدل‌های مبتنی بر ویژگی‌های پنهان را شرح می‌دهیم و در ادامه روش‌هایی که در بالا نام برده شدند را توضیح داده و نقاط قوت و ضعف آن‌ها را بررسی می‌کنیم و نتایج بدست آمده از هریک از این روش‌ها را نیز بررسی خواهیم کرد و در فصل بعد تلاش بر بهبود این روش‌ها خواهیم کرد.

۳-۲- تقسیم‌بندی داده‌های پایگاه دانش

برای آموزش دادن مدل‌های نام برده شده در بخش قبل از پایگاه دانش Freebase که قسمتی از گراف دانش تولید شده توسط گوگل است استفاده می‌کنیم. در این پایگاه دانش ۸۰ میلیون موجودیت، ۲۰ هزار نوع رابطه مختلف و ۱,۲ میلیارد حقیقت وجود دارد [2]. حقیقت‌هایی که در این پایگاه دانش وجود دارد به صورت سه‌تایی مرتب

(subject, predicate, object)

مشخص شده‌اند به این صورت که موجودیت subject رابطه‌ی predicate دارد با موجودیت object. به این نوع ذخیره‌ی داده اصطلاحا SPO گفته می‌شود [6]. برای مثال یک نمونه حقیقت موجود در این پایگاه دانش به صورت:

(Barack Obama, place_of_birth, Hawai)

است که بیانگر حقیقت «باراک اوباما متولد هاوایی است» می‌باشد.

این پایگاه دانش شامل تعداد زیادی حقیقت است که عملیات آموزش روی آن هزینه‌ی زیادی از نظر زمان و منابع خواهد داشت. برای سادگی و تسریع کار از یک نمونه‌ی نرمال کوچک این پایگاه دانش به نام Freebase15k استفاده می‌کنیم که در آن ۱۴۹۵۱ موجودیت، ۱۳۴۵ رابطه‌ی مختلف و ۵۹۲۲۱۳ حقیقت وجود دارد [14].

جدول ۱ تقسیم‌بندی پایگاه‌های دانش

پایگاه دانش	موجودیت‌ها	رابطه‌ها	آموزش	ارزیابی	آزمون
FB15K	۱۴,۹۵۱	۱,۳۴۵	۴۸۳,۱۴۲	۵۰,۰۰۰	۵۹,۰۷۱
FB1M	۱×۱۰^{۶}	۲۳,۳۸۲	$۱۷/۵ \times ۱۰^{۶}$	۵۰,۰۰۰	۱۷۷,۴۰۴

روش‌های مبتنی بر ویژگی‌های پنهان به صورت تکراری انجام می‌شوند و نیاز است در هر تکرار بررسی کنیم که به آستانه‌ی مناسب برای قطع تکرار الگوریتم رسیده‌ایم یا خیر. هم‌چنین پس از انجام عملیات آموزش نیاز است که مدل آموزش دیده را آزمایش کنیم و میزان دقت آن را بدست آوریم.

برای انجام عملیات آموزش، بررسی کیفیت آموزش در هر مرحله و بررسی کیفیت کلی آموزش به سه دسته مختلف از داده نیاز داریم.

- داده‌های آموزش: قسمت عمده‌ی داده‌ها برای عملیات آموزش استفاده می‌شود که الگوریتم اصلی هر روش روی آن اعمال می‌شود و سعی می‌کنیم پارامترهایی که همان ویژگی‌های پنهان هستند را تنظیم کنیم.
- داده‌های ارزیابی: قسمتی از داده که داده‌های ارزیابی نام دارند برای بررسی میزان بهبود یا تخریب مدل در هر تکرار استفاده می‌شوند، این داده‌ها کاملا از داده‌های آموزش جدا هستند و در مرحله‌ی آموزش اصلا به مدل نشان داده نمی‌شود و مدل تحت تاثیر این داده‌ها قرار نمی‌گیرد. در انتهای هر مرحله به کمک این داده‌ها بررسی می‌شود که تغییراتی که روی پارامترهای این مدل انجام شده است باعث بهتر یا بدتر شدن این مدل شده است، در صورتی که بهبودی مشاهده شود تغییرات انجام شده در این مرحله نگه داشته می‌شود و به سراغ مراحل

بعدی می‌رویم اما اگه نتایج بدتر شده باشد پارامترها را به مقادیر قبلی برگردانده و مرحله‌ی بعدی را شروع می‌کنیم.

- داده‌های آزمون: قسمت دیگری از داده که داده‌های آزمون نام دارند برای بررسی کیفیت عملکرد کلی مدل به کار می‌روند. این داده‌ها نیز کاملا از داده‌های آموزش تفکیک شده‌اند و در زمان آموزش روی مدل تاثیری نمی‌گذارند و کلا در هیچ یک از مراحل آموزش استفاده نمی‌شوند و تنها پس از آموزش مدل استفاده می‌شوند تا کیفیت مدل آموزش دیده شده را بررسی کنند.

در این تحقیق پایگاه دانش Freebase15k به صورت شکل ۵ تقسیم و استفاده شده است، به این صورت که برای قسمت آموزش، از ۴۸۳,۱۴۲ حقیقت، برای قسمت ارزیابی، از ۵۰ هزار حقیقت و برای قسمت آزمون ۵۹ هزار حقیقت استفاده کرده‌ایم [14].

شکل ۵ نحوه‌ی تقسیم پایگاه دانش freebase15k به سه قسمت آموزش، ارزیابی و آزمون

۴-۲- مراحل آموزش روش‌های مبتنی بر ویژگی‌های پنهان

همان‌طور که در بخش قبل گفته شد پایگاه دانش را به سه قسمت آموزش، ارزیابی و آزمون تقسیم می‌کنیم. نحوه‌ی آموزش کلی همه‌ی روش‌های مبتنی بر ویژگی‌های پنهان در سه مرحله انجام می‌شود: ۱- مرحله‌ی آموزش ۱-مرحله‌ی ارزیابی ۳- مرحله‌ی آزمون. مراحل آموزش و ارزیابی به صورت تکراری و معمولا با تکرار بالا انجام می‌شوند و در هر تکرار بررسی می‌شود که بهبودی اتفاق افتاده است یا خیر،

اگه بهبودی داشتیم نتایج این مرحله تکرار را نگه داشته و مرحله‌ی تکرار بعدی را شروع می‌کنیم و اگر بهبودی اتفاق نیافتاده بود تنایج این مرحله را تاثیر نمی‌دهیم و مرحله‌ی تکرار بعدی را شروع می‌کنیم.

در ادامه این سه مرحله را توضیح می‌دهیم.

۱-۴-۲- مرحله‌ی آموزش

همان‌طور که قبل‌تر تو ضیح داده شد آموزش به صورت تکراری انجام می شود و در هر تکرار الگوریتم آموزش روش مورد نظر، روی قسمتی یا همه‌ی داده‌های مجموعه‌ی آموزش انجام می‌شود. در این قسمت سعی می‌شود که پارامترهای همه‌ی ویژگی‌ها جهت‌دهی شده و آموزش ببینند تا کمترین خطا در پاسخ به سوالاتی که از مدل پرسیده می‌شود را داشته باشند.

۲-۴-۲- مرحله‌ی ارزیابی

پس از هر تکرار مدل آموزش دیده شده را روی داده‌های ارزیابی اجرا می‌کنیم و معیارهایی که برای بررسی کیفیت مدل در نظر گرفته‌ایم را بدست می‌آوریم و از روی آن میزان بهبود مدل در این تکرار را بررسی می‌کنیم.

برای مثال فرض کنید که حقیقت زیر در مجموعه داده‌های ارزیابی وجود دارد و در مرحله‌ی آموزش مدل این حقیقت مشاهده نشده است:

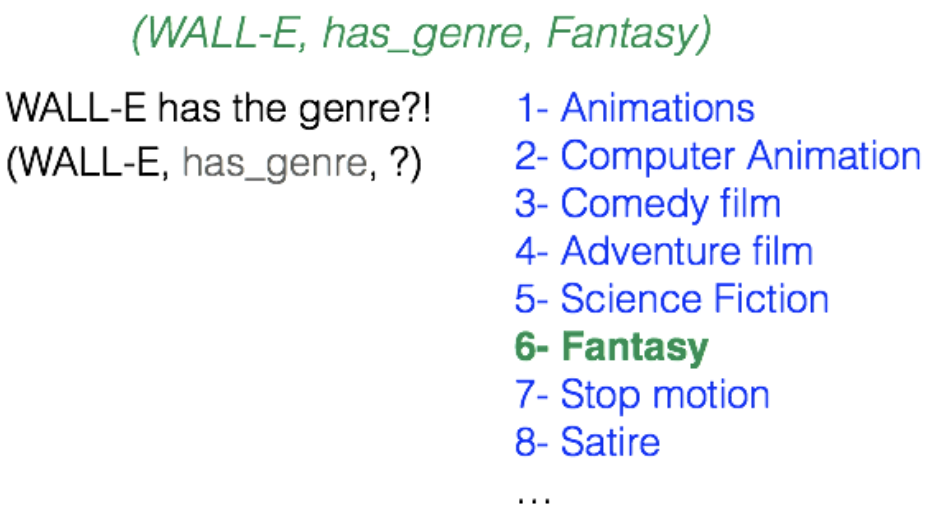
(WALL-E, has_genre, Fantasy)

این حقیقت به این معناست که «ژانر فیلم WALL-E فانتزی است». در مرحله‌ی ارزیابی موجودیت اول یا آخر این حقیقت را حذف می‌کنیم و قسمت حذف شده را از مدل سوال می‌پرسیم و انتظار داریم که قسمت حذف شده را حدس بزند. سوالی که از این مدل پرسیده می‌شود به این صورت است:

(WALL-E, has_genre, ?)

به این معنی که «ژانر فیلم وال-ای چیست؟».

در مدل‌های مبتنی بر ویژگی‌های پنهان پاسخ به این‌گونه سوالات به صورت مجموعه‌ی مرتب شده‌ی همه موجودیت‌هاست. یعنی این مدل میزان نزدیکی همه‌ی موجودیت‌ها با پاسخ این سوال را بررسی می‌کند و به ترتیب نزدیک به دور مرتب می‌کند و به عنوان پاسخ به ما می‌دهد. برای مثال پاسخ به سوال بالا به صورت شکل ۶ می‌باشد [14].



شکل ۶ نمونه‌ی سوال و پاسخ از مدل

همان‌طور که می‌بینیم مدلی که این سوال از آن پرسیده شده پاسخ درست را در ششمین حدس به ما داده است.

در مرحله‌ی ارزیابی نیاز به معیارهایی داریم که بررســـی کنیم که مدل با توجه به این معیارها بهبود داشته یا خیر. در این تحقیق از دو معیار زیر که در [14] معرفی شده است استفاده می‌کنیم:

- رتبه‌ی میانگین (Mean rank): میانگین رتبه‌ی جواب‌های درستی که مدل داده است.
- حدس زیر ۱۰: درصد سوالاتی که پاسخ درست مدل به آن زیر رتبه‌ی ۱۰ بوده است که در ادامه به آن hit@10 می‌گوییم.

همان‌طور که قبل‌تر توضیح داده شد ۵۰هزار حقیقت در دسته‌ی ارزیابی وجود دارد، ما پس از هر تکرار الگوریتم، دو معیار بالا را به دست آورده و میزان بهبود مدل را اندازه می‌گیریم. پس از پرسیدن این ۵۰ هزار سوال میانگین رتبه‌ای که جواب‌های درست داشته معیار اول را به ما می‌دهد و درصد سوالاتی که جواب درست آن زیر رتبه‌ی ۱۰ بوده است معیار دوم را به ما می‌دهد.

۳-۴-۲- مرحله‌ی آزمون:

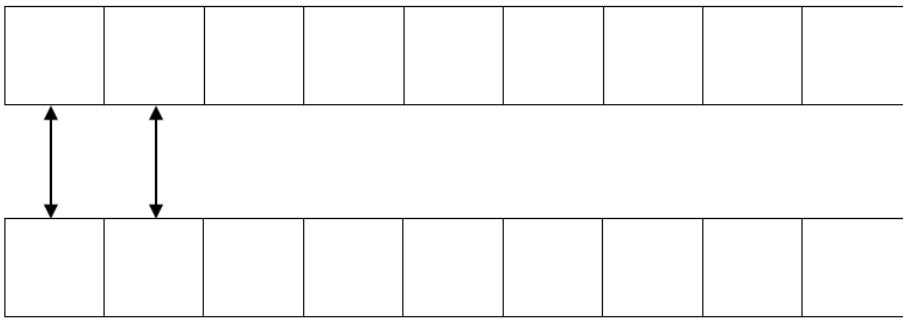
پس از انجام کامل مراحل آموزش و ارزیابی و متوقف شدن الگوریتم از داده‌های آزمون که در بخش ۳-۲- توضیح دادیم و معیارهای کیفیت که در بخش ۱-۴-۲- توضیح دادیم استفاده می‌کنیم و کیفیت مدل را بررسی می‌کنیم. در این بخش هم مانند بخش ارزیابی، داده‌ها به مدل در حال آموزش نشان داده نشده و برای مدل جدید هستند. در این مرحله هم یک قسمت از هر حقیقت موجود در داده‌های آزمون را حذف کرده و آن را از مدل سوال می‌پرسیم، دقیقا مانند مرحله‌ی ارزیابی. پس از پرسیدن سوالات دو معیار رتبه‌ی میانگین و hit@10 را بدست می‌آوریم که این دو معیار نشان دهنده‌ی میزان کیفیت و دقت روش است.

۵-۲- بررسی روش‌های موجود

در ادامه نحوه‌ی عملکرد روش‌های موجود را توضیح داده و بررسی می‌کنیم.

۱-۵-۲- روش‌های خطی

همان‌طور که در بخش ۲-۲- گفتیم در روش‌های مبتنی بر ویژگی‌های پنهان موجودیت‌ها و رابطه‌ها به صورت بردارهایی در یک فضای n بعدی تبدیل می شوند که به کمک معیارهای مختلف شباهت، رابطه‌ی بین دو موجودیت را بدست می‌آوریم. در روش‌های خطی در زمان آموزش و بررسی میزان شباهت بردارها را به صورت خطی با یکدیگر مقایسه می‌کنیم، به این صورت که هر اندیس از بردار موجودیت اول را با اندیس متناظر آن در بردار رابطه یا موجودیت دیگر بررسی می‌کنیم، شکل ۷.

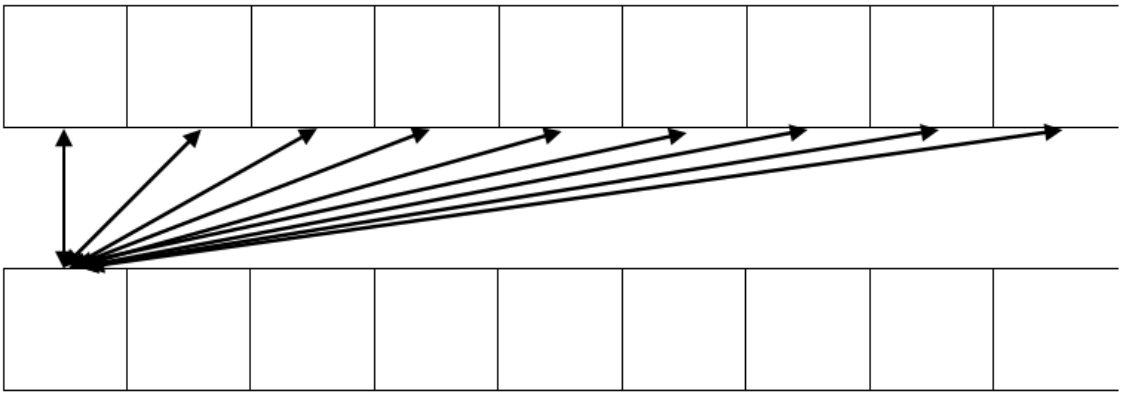


شکل ۷ نحوه‌ی مقایسه‌ی ویژگی‌ها در روش‌های خطی

با توجه به نتایج ضعیفی که روش‌های این دسته در آموزش مدل و پیش‌بینی پیوند بدست آورده اند به این روش‌ها نمی‌پردازیم و به همین معرفی اکتفا می‌کنیم.

۲-۵-۲- روش‌های دوخطی – رسکال[1]

روش رسکال [17] ,[16], [15] یکی از روش‌های ویژگی‌های پنهان رابطه‌ای[2] است که حقیقت‌های پایگاه دانش را به صورت تراکنش‌های بین جفت ویژگی‌های پنهان در نظر می‌گیرد. یعنی بر خلاف روش‌های خطی هر ویژگی پنهان از هر بردار را با همه‌ی ویژگی‌های پنهان دیگر بردار بررسی می‌کند، به صورت شکل ۸. به همین دلیل این روش را روش دوخطی نیز می‌نامیم.



شکل ۸ نحوه‌ی نگاشت ویژگی‌ها در روش‌های دوخطی

در این روش امتیاز هر سه‌تایی را از رابطه‌ی (۲-۱) بدست می‌آوریم که در آن H_e تعداد ابعاد بردار در نظر گرفته شده برای موجودیت‌ها است (تعداد ویژگی‌های پنهان هر موجودیت). W_k یک بردار وزن با ابعاد $H_e \times H_e$ است که هر اندیس w_{abk} نشانگر این است که ویژگی پنهان a و b در رابطه‌ی k چقدر با هم کنش می‌کنند.

[1] RESCAL

[2] Relational latent factor

$$f_{ijk}^{RESCAL} := e_i^T W_k e_j = \sum_{a=1}^{He} \sum_{b=1}^{He} w_{abk} e_{ia} e_{jb} \qquad (۱-۲)$$

همان‌طور که در رابطه‌ی (۲-۱) مشاهده می‌شود در این روش هر رابطه به یک ماتریس تبدیل شده و هر موجودیت به یک بردار، اگر حقیقت (a, k, b) به این معنی که موجودیت a رابطه‌ی k با b دارد را در نظر بگیریم و بخواهیم بررسی کنیم که این رابطه برقرار هست یا خیر، احتمال وجود این رابطه را از روی امتیازی که تابع امتیاز (۲-۱) به ما می‌دهد بدست می‌آوریم. این امتیاز به این صورت محاسبه می‌شود که بردار موجودیت a در ماتریس مربوط به k ضرب شده و پس از آن در بردار b ضرب می‌شود که نتیجه‌ی آن یک مقدار حقیقی است که امتیاز این حقیقت را به ما می‌دهد.

در ادامه به برخی از نکات مورد توجه این مدل به صورت موردی اشاره می‌کنیم.

آموزش رابطه‌ای[1] از طریق بازنمایی مشترک[2]: در رابطه‌ی (۲-۱) هر موجودیت به صورت یک بردار بازنمایی شده است بدون توجه به این‌که در قسمت اول حقیقت می‌آید یا قسمت دوم آن. هم‌چنین این موجودیت‌ها برای همه‌ی رابطه‌ها یک بازنمایی مشترک دارند و در هر رابطه نیاز به تعریف جدید ندارند. برای مثال موجودیت i در قسمت اول حقیقت x_{ijk} با رابطه‌ی k آمده است و همین موجودیت در رابطه‌ی x_{piq} به عنوان موجودیت دوم در رابطه‌ی q ظاهر شده است. هر دو تابع امتیاز $f_{ijk} = e_i W_k e_j$ و $f_{piq} = e_p W_q e_i$ از یک بازنمایی برای موجودیت i (e_i) استفاده می‌کنند. بنابراین همه‌ی پارامترها به

[1] Relational Learning

[2] Shared representations

صورت مشترک[1] آموزش دیده می‌شوند و این بازنمایی مشترک باعث می‌شود که اطلاعات روی همه‌ی حقیقت‌ها به وسیله‌ی بازنمایی موجودیت‌ها و ماتریس وزن‌دار رابطه‌ها پخش شوند و بتوانیم وابستگی‌های جهانی[2] در داده‌ها را تشخیص دهیم [6].

ارتباط معنایی بردارها[3]: خاصیت بازنمایی مشترک در این روش کمک می‌کند که میزان شباهت موجودیت‌ها در فضای رابطه‌ای[4] نیز بدست بیاید. برای مثال موجودیت‌هایی که با رابطه‌های مشابه به موجودیت‌های مشابه متصل هستند به یکدیگر شبیه هستند. به عنوان نمونه اگر بازنمایی e_i و e_p شبیه به هم باشد، تابع امتیاز f_{ijk} و f_{pjk} باید مقادیر نزدیک به هم داشته باشند پس موجودیت‌ها با تعداد زیادی رابطه‌ی مشترک بازنمایی یکسانی خواهند داشت. این خصوصیت می‌تواند در بازنمایی[5] و خوشه‌بندی‌های با مقیاس بالای[6] موجودیت‌ها روی داده‌های رابطه‌ای مورد استفاده قرار گیرد [6], [15], [16].

ارتباط با عامل‌بندی تنسور[7]: این روش شباهت زیادی به روش‌های استفاده شده در سیستم‌های توصیه‌گر[18] و عامل‌بندی تنسور سنتی دارد[19]. ضرب ماتریس که در معادله‌ی (۱-۲) معرفی کردیم می‌تواند به صورت $F_k = EW_kE^T$ نوشته شود که در آن $F \in R^{Ne \times Ne}$ ماتریسی است که همه‌ی امتیازات مربوط به رابطه‌ی k را در خود جا داده است و سطر i ام از ماتریس $E \in R^{Ne \times He}$ بازنمایی موجودیت e_i

[1] jointly

[2] global dependency

[3] Semantic embeddings

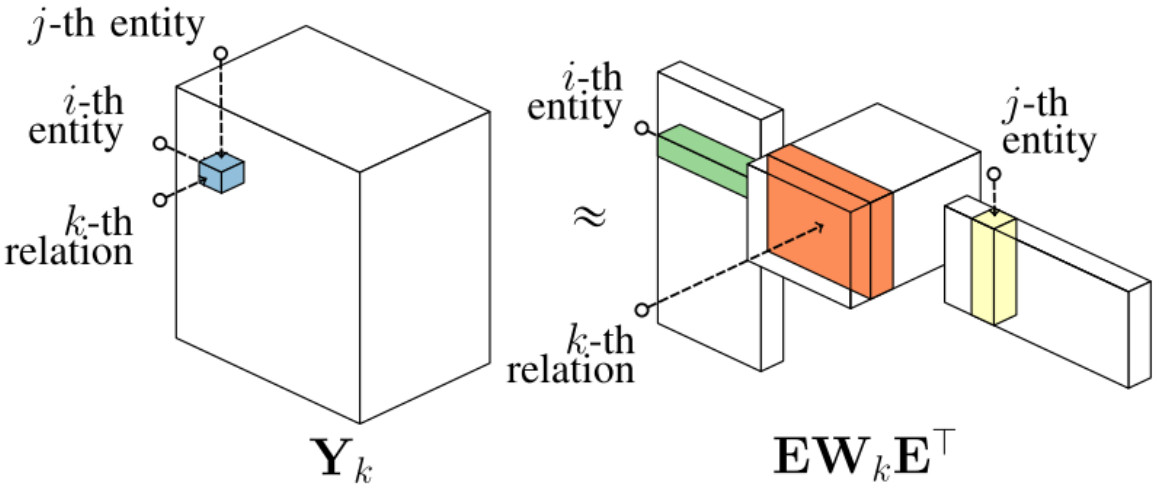
[4] relational domain

[5] resulotion

[6] large-scale hierarchical clustering

[7] Tensor factorization

است که برداری از ویژگی‌های پنهان این موجودیت است. در شکل ۹ این تبدیل ماتریس به تنسور نمایش داده شده است [6].



شکل ۹ تبدیل ماتریس به تنسور در روش رسکال

برازش مدل: اگر بخواهیم از یک مدل آماری استفاده کنیم، پارامترهای این روش می‌توانند به صورت یک مدل کمینه‌سازی برمبنای گرادیان[1] تخمین زده شوند مانند گرادیان نزولی تصادفی (SGD)[2] [20]. این روش را می‌توانیم به صورت یک روش بر مبنای امتیاز حل کرد که می‌تواند پارامترهای مدل را به صورت بسیار بهینه‌ای تخمین بزند: با توجه به ساختار تنسور که در بالا توضیح دادیم و هم‌چنین با توجه به تنک بودن داده‌های موجود، نشان می‌دهیم که روش رسکال می‌تواند به کمک توالی[3]ای از بروزرسانی‌های بسته‌ی کارآمد[4] محاسبه شود [17] ,[15]. به صورت تحلیلی می‌توان نشان داد که در

[1] gradient-based minimization

[2] stochastic gradient descent

[3] sequence

[4] efficient close-form update

این راهکار با هر بروزرسانی در E و W_k به صورت خطی با تعداد موجودیت‌ها N_e، تعداد رابطه‌ها N_r و تعداد حقیقت‌های مشاهده شده توسط مدل رشد می‌کند.[17]

پیش‌بینی مجزا: در رابطه‌ی (۲-۱) احتمال وجود یک رابطه از روی یک ضرب ماتریسی ساده از مرتبه‌ی $O(H_e^2)$ بدست می‌آید. بنابراین، زمانی که پارامترهای مدل تخمین زده شدند، پیچیدگی محاسبات برای پیش‌بینی امتیاز یک حقیقت فقط به تعداد ویژگی‌های پنهان وابسته است و مستقل از اندازه‌ی کل گراف است. با این حال به لطف بازنمایی مشترک که قبل‌تر توضیح داده شده، این مدل می‌تواند در زمان تخمین پارامترها، وابستگی‌های جهانی بین موجودیت‌ها و رابطه‌ها را فهمیده و در فرایند آموزش تاثیر دهد [6].

نتایج یادگیری رابطه‌ای: رسکال در زمینه‌های مختلف مدل‌های یادگیری مدرن[1] مورد استفاده قرار گرفته است. برای مثال در [15] نشان داده شده است که رسکال موفق شده در پیش‌بینی رابطه نتایج نزدیک یا بهتر روی چندین مجموعه داده‌ی معیار نسبت به روش‌های Markov Logic Networks [21]و the Infinite (Hidden) Relational model [22], [23] و Bayesian Clustered Tensor Factorization [24] بدست آورد. هم‌چنین رسکال برای پیش‌بینی پیوند روی کل پایگاه دانش مانند YAGOو DBpedia مورد استفاده قرار گرفته است [25] ,[17]. فارغ از پیش‌بینی پیوند، رسکال در روش‌های یادگیری رابطه‌ای تصادفی (SRL)[2] مانند بازنمایی موجودیت‌ها[3] و خوشه‌بندی بر مبنای پیوند[4] عملکرد موفقی داشته است. برای مثال این روش عملکرد موفقی در

[1] state-of-the-art

[2] stocastic relational learning

[3] entity resulotion

[4] link-based clustring

دسته‌بندی نویسندگان، ناشران و سالن‌های انتشار روی مجموعه داده‌های ناشران داشته است [15], [16]. علاوه بر این، ارتباط معنایی موجودیت‌ها که در این روش محاسبه شد در ایجاد طبقه‌بندی[1] به کمک خوشه‌بندی سلسله مراتبی[2] روی داده‌های دسته‌بندی نشده[3] به کار برده شده است [26].

۳-۵-۲- روش‌های ادراک چندلایه‌ای[4]

می‌توانیم رسکال را به صورت مدلی که برای هر حقیقت یک بازنمایی تولید می‌کند و از روی این بازنمایی‌ها وجود یا عدم وجود این حقیقت‌ها را پیش‌بینی می‌کند تفسیر کنیم. به طور خاص می‌توانیم رسکال را به صورت رابطه‌های (۲-۲) و (۲-۳) بازنویسی کنیم.

$$f_{ijk}^{RESCAL} := w_k^T \emptyset_{ij}^{RESCAL} \qquad (۲-۲)$$

$$\emptyset_{ij}^{RESCAL} = e_j \otimes e_i \qquad (۲-۳)$$

در روابط (۲-۱) ضرب داخلی e_i و e_j که در محاسبه‌ی مجموع امتیازها شرکت می‌کردند را از مجموع بیرون کشیده و به صورت ضرب خارجی نوشتیم. بنابراین رسکال بازنمایی جفت موجودیت i و j را به صورت ضرب تنسور ویژگی‌های پنهان این دو موجودیت بدست آورده است. می‌توان احتمال وجود حقیقت x_{ijk} را از روی ضرب داخلی بازنمایی جفت موجودیت‌ها (φ_{ij}) و ماتریس وزن‌دار رابطه (w_k^T)

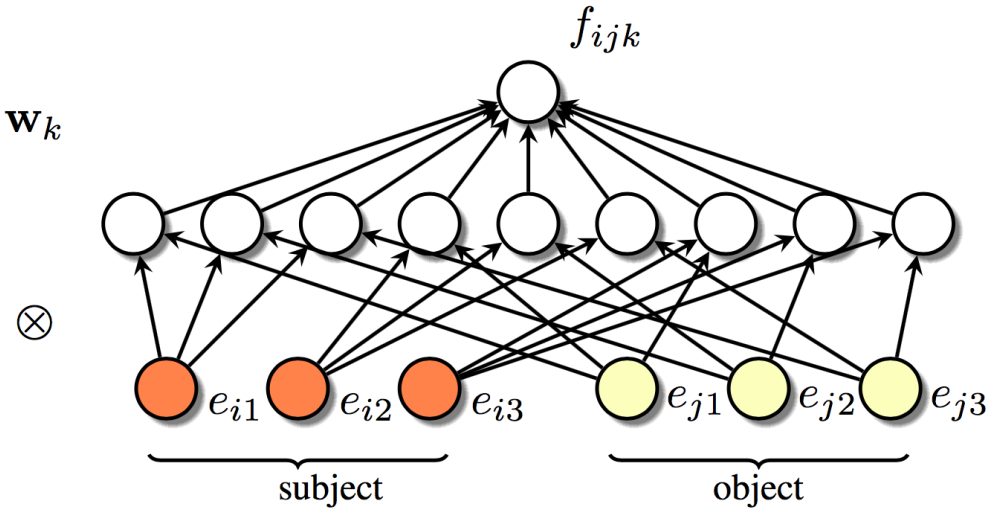
[1] taxonomies

[2] hierarchical clusterings

[3] uncategorized data via

[4] Multi-layer perceptrons

بدست آورد. این تقسیم‌بندی رابطه‌ی رسکال در شکل ۱۰ به تصویر کشیده شده است. توضیح بیشتر در مورد ایجاد بازنمایی پنهان به وسیله‌ی ضرب تنسور در [28] ,[27] آمده است.



شکل ۱۰ تقسیم رابطه‌ی رسکال به دو لایه

از آنجایی که ضرب تنسور تراکنش بین همه‌ی جفت موجودیت‌ها را مدل می‌کند، وقتی تعداد ویژگی‌های پنهان زیاد باشد، رسکال در این راهکار به تعداد زیادی پارامتر نیاز پیدا خواهد کرد. این موضوع می‌تواند باعث مشکل در مقیاس‌پذیری روش روی پایگاه‌های دانش بزرگ با تعداد زیاد رابطه شود.

در ادامه‌ی این بخش به روش‌های ادراک چندلایه‌ای که به شبکه‌های عصبی پیش‌خور[1] نیز معروف هستند می‌پردازیم. این راهکار این امکان را به ما می‌دهد که مدل‌های جایگزینی برای ساختن بازنمایی حقیقت‌ها در نظر بگیریم و هم‌چنین بتوانیم از توابع غیر خطی در پیش‌بینی وجود پیوندها استفاده کنیم.

[1] feedforward neural networks

مدل E-MLP[1]:

در ابتدا مدل Entity-MLP را معرفی می‌کنیم. تابع امتیاز این مدل به صورت معادلات (۴-۲) و (۵-۲) و (۶-۲) محاسبه می‌گردد.

$$f_{ijk}^{E-MLP} := w_k^T g(h_{ijk}^a) \quad (۴-۲)$$

$$h_{ijk}^a = A_k^T \emptyset_{ij}^{E-MLP} \quad (۵-۲)$$

$$\emptyset_{ij}^{E-MLP} = [e_i; e_j] \quad (۶-۲)$$

در روابط فوق $g(u) = [g(u_1),g(u_2),.. ..]$ یک تابع g است که بر روی تک تک المان‌های بردار u اعمال می‌شود.که می‌تواند یک تابع غیر خطی باشد، مانند $g(u) = \tanh(u)$.

در این روابط h_a یک لایه‌ی مخفی[2] اضافه شده است که ماتریس وزن‌دار دیگر روی بازنمایی موجودیت‌ها اعمال می‌کند. در اصل ما در رابطه‌ی (۲-۶) دو موجودیت e_i و e_j را ترکیب کردیم و هیچ کنشی بین این دو موجودیت محاسبه و تاثیر داده نشده است، بنابراین به یک ماتریس وزن‌دار برای محاسبه‌ی تاثیر این دو موجودیت نیاز بود و h_a وارد معادله شد.

[1] Entity-MLP

[2] hidden layer

بزرگترین تفاوتی که این روش با روش‌های ضرب تنسور مانند رسکال دارد این است که در اینجا بجای محاسبه‌ی همه‌ی تراکنش‌های ممکن بین دو موجودیت، فقط تراکنش‌های موجود در A_k مورد بررسی قرار می‌گیرد. این راهکار به طور قابل ملاحظه‌ای تعداد پارامتری که باید آموزش دیده شوند را کاهش می‌دهد.

روش ER-MLP[1]:

یکی از اشکالات روش E-MLP این است که باید برای هر رابطه‌ی ممکن، یک بردار w_k و یک ماتریس A_k تعریف شود، که تعداد پارامترها را زیاد می‌کند. برای حل این مشکل روش ER-MLP معرفی شده است که در این روش رابطه را نیز به صورت برداری در کنار موجودیت‌ها در نظر می‌گیریم، و بجای ماتریس Ak می‌توانیم از یک بردار ثابت C استفاده کنیم. بنابراین روابط ER-MLP به صورت (۲-۷) و (۲-۸) و (۲-۹) خواهند بود.

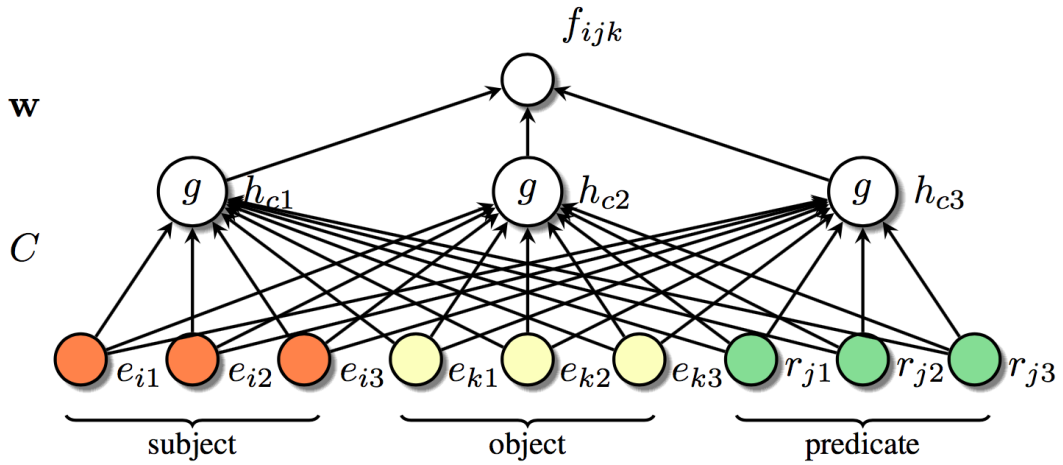
$$f_{ijk}^{ER-MLP} := w^T g(h_{ijk}^c) \quad (۲-۷)$$

$$h_{ijk}^c = C^T \emptyset_{ijk}^{ER-MLP} \quad (۲-۸)$$

$$\emptyset_{ijk}^{ER-MLP} = [e_i; e_j; r_k] \quad (۲-۹)$$

[1] Entity-Relation-MLP

توجه کنید که در این روش از یک بردار وزن‌دار جهانی برای همه روابط استفاده شده است و بردار C تعریف شده مستقل از رابطه‌ی rk است که باعث کاهش تعداد پارامترهای مساله می‌گردد. نحوه‌ی کار روش ER-MLP در شکل ۱۱ نمایش داده شده است.



شکل ۱۱ روش ER-MLP

در [29] نشان داده شده است که روش‌های MLP کلماتی که قرابت معنایی[1] دارند را به درستی نزدیک به یکدیگر تشخصی می‌دهند در حالی که برای چنین کاربردی آموزش دیده نشده اند. هم‌چنین در [30] این ویژگی نشان داده شده است، برای مثال به کمک MLP مساله‌ی نزدیک‌ترین همسایه[2] برای بازنمایی پنهان چندین رابطه‌ی انتخاب شده از پایگاه دانش Freebase را حل کرده اند. در نتایج می‌توان مشاهده کرد که روابطی که ارتباط معنایی دارند نزدیک به یکدیگر قرار گرفته‌اند.

[1] semantically similar

[2] nearst neighbors

۴-۵-۲- شبکه‌های عصبی تنسور[1]

در [31] با ترکیب روش‌های ادراک چند لایه‌ای (MLP) و روش‌های دوخطی (Bilinear) روش جدیدی به نام شبکه‌های عصبی تنسور (NTN) معرفی شده است. روابط محاسبه‌ی تابع امتیاز این روش به صورت (۲-۱۰) و (۲-۱۱) و (۲-۱۲) است.

$$f_{ijk}^{NTN} := w_k^T g([h_{ijk}^a; h_{ijk}^b]) \quad (۱۰-۲)$$

$$h_{ijk}^a := A_k^T [e_i; e_j] \quad (۱۱-۲)$$

$$h_{ijk}^b := [e_i^T B_k^1 e_j . \cdots . e_i^T B_k^{H_b} e_j] \quad (۱۲-۲)$$

در اینجا B_k یک تنسور است، که در آن میزان رابطه‌ی بین جفت موجودیت‌ها نگاشت شده است. h_{ijk}^b را یک لایه‌ی مخفی دوخطی[2] می‌نامیم، زیرا هم به صورت یک لایه‌ی مخفی از مدل‌های MLP هست و هم به کمک یک تابع وزن‌دار میزان کنش جفت موجودیت‌ها را مشخص می‌کند مانند آنچه در روش رسکال توضیح دادیم.

با توجه به معادله‌های معرفی شده در (۲-۱۰) و (۲-۱۱) و (۲-۱۲) مشخص است که این روش مشکلات مقیاس پذیری که در هر دو روش E-MLP و رسکال وجود داشت را دارد. هم‌چنین در [32] و [30]

[1] Neural tensor networks

[2] bilinear hidden layer

نشان داده شده است که این روش به بیش‌برازش[1] میل می‌کند (حداقل روی مجموعه داده‌هایی که در این مقالات استفاده شده است) [33].

۵-۵-۲- روش‌های فاصله‌ی پنهان[2]

کلاس دیگری از مدل‌ها، مدل‌های فاصله‌ی پنهان هستند (که در تحلیل شبکه‌های اجتماعی به مدل‌های فضای پنهان معروف هستند) که احتمال وجود رابطه بین موجودیت‌ها را از فاصله‌ی بین بازنمایی پنهان آن‌ها در فضا می‌سنجد به این صورت که: موجودیت‌ها با یکدیگر رابطه دارند اگر بازنمایی پنهان آن‌ها با یک معیار فاصله نزدیک به هم باشد [33].

در [34] برای داده‌های تک-رابطه‌ای[3] برای اولین بار راهکاری در این زمینه معرفی شده است. این راهکار احتمال وجود پیوند در شبکه‌های اجتماعی را با تابع امتیاز $f(e_i.e_j) = -d(e_i.e_j)$ محاسبه می‌کند، که در آن d(.,.) یک تابع اندازه‌گیری فاصله‌ی دلخواه مانند فاصله‌ی اقلیدسی[4] است. در ادامه سه روش Structured Embedding (SE)، TransE و TransH را به تفصیل توضیح می‌دهیم.

[1] overfit

[2] Latent distance models

[3] uni-relational data

[4] Euclidean distance

۶-۵-۲- مدل Structured embedding

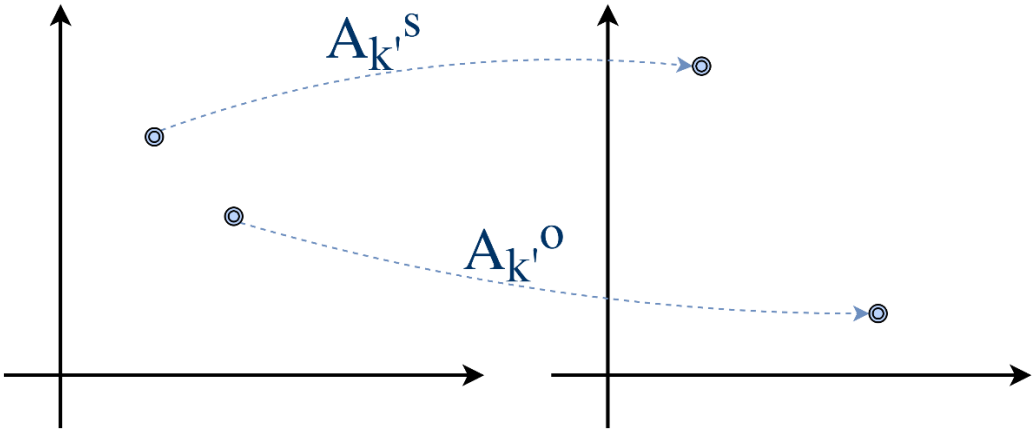
در [35] مدلی به نام Structured Embedding (SE) معرفی شده که در آن ایده‌ای که در بند قبل توضیح داده شد برای داده‌های چند-رابطه‌ای[1] گسترش داده شده است. در این روش تابع امتیاز برای حقیقت‌های x_{ijk} به صورت (۲-۱۳) مدل شده است.

$$f_{ijk}^{SE} := -\| A_k^s e_i - A_k^o e_j \|_1 = -\| h_{ijk}^a \|_1 \qquad (۱۳-۲)$$

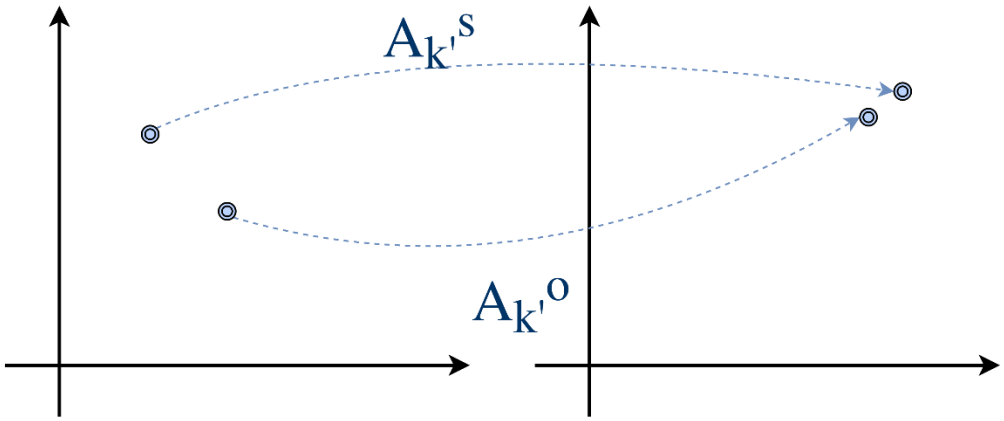
در رابطه‌ی (۲-۱۳) A_k از ماتریس‌های مربوط به موجودیت‌های اول و دوم $[A_k^s, A_k^o]$ تشکیل شده است. ماتریس‌های A_k^s و A_k^o بازنمایی ویژگی‌های پنهان موجودیت‌ها را به فضای مخصوص به رابطه‌ی k منتقل می‌کند. این انتقال به صورتی آموزش دیده می‌شود که جفت رابطه‌هایی که برقرار هستند، نسبت به جفت رابطه‌هایی که برقرار نیستند به یکدیگر نزدیکتر باشند.

در شکل ۱۲ مشاهده می‌شود که در صورتی که بین دو موجودیت i و j رابطه‌ی k برقرار باشد، پس از نگاشت این دو موجودیت توسط ماتریس‌های نگاشت A_k^s و A_k^o این دو موجودیت به یکدیگر نزدیکتر شده اند. همینطور در شکل ۱۳ مشاهده می‌شود که در صورتی که رابطه‌ی k' بین دو موجودیت i و j برقرار نباشد، پس از نگاشت این دو توسط ماتریس‌های نگاشت A_k^s و A_k^o دو موجودیت در فاصله‌ی بیشتری از یکدیگر قرار می‌گیرند.

[1] multi-relational



شکل ۱۲ افزایش فاصله‌ی دو موجودیت که رابطه‌ی k’ بین آنها برقرار نیست پس از نگاشت توسط بردار انتقال k’ در روش Structred Embedding



شکل ۱۳ کاهش فاصله‌ی دو موجودیت که رابطه‌ی k’ بین آنها برقرار است پس از نگاشت توسط بردار انتقال k’ در روش Structred Embedding

یکی از نکات منفی که در این روش به چشم می‌خورد یادگیری ماتریس‌های وزن‌دار جدا برای موجودیت‌های اول و دوم است، که باعث افزایش تعداد پارامتر مورد نیاز برای آموزش است.

۷-۵-۲- مدل TransE[1]

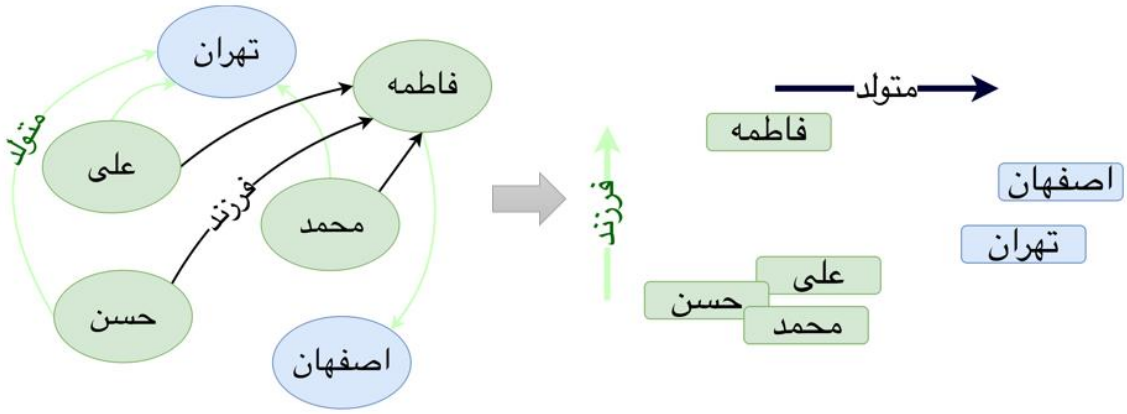
در [14] مدلی برای حل مشکل تعداد پارامتر زیاد در روش SE مطرح شده است که بجای این‌که از ماتریس‌های A_k^s و A_k^o برای تاثیر دادن رابطه‌ها استفاده شود، رابطه را یک بردار همانند بردار موجودیت‌ها در نظر گرفته شده و به عنوان یک آفست در کنار موجودیت اول در محاسبات از آن استفاده می‌کند. [6] به طور خاص امتیاز حقیقت x_{ijk} از رابطه‌ی (۲-۱۴) بدست می‌آید.

$$f_{ijk}^{TransE} := -d(e_i + r_k . e_j) \qquad (۱۴-۲)$$

در این روش رابطه‌ها به صورت یک بردار انتقال استفاده می‌شوند، به این صورت که فقط روی موجودیت اول اعمال می‌شوند و در صورت وجود رابطه بین این دو موجودیت، موجودیت اول را به موجودیت دوم نزدیک می‌کنند. در شکل ۱۴ یک گراف را مشاهده می‌کنیم که در آن روابط فرزندی و مکان تولد بین ۶ موجودیت نمایش داده شده است. در شکل ۱۴ یک مثال از اعمال روش TransE روی این گراف را مشاهده می‌کنیم. مشاهده می‌شود که موجودیت‌ها بردارهایی ثابت در نظر گرفته شده‌اند و بردارهای رابطه به صورت یک بردار انتقال به تصویر کشیده شده است که موجودیت‌ها را به هدف‌های مورد نظر نزدیک می‌کنند.

مثلا اگر «علی» را توسط بردار «فرزند» منتقل کنیم به موجودیت «فاطمه» نزدیک می‌شود که حقیقت (علی، فرزند، فاطمه) را تشکیل می‌دهد. اما اگر همین موجودیت را توسط بردار رابطه‌ی «متولد» منتقل کنیم، به موجودیت «تهران» نزدیک می‌شود که حقیقت (علی، متولد، تهران) را تشکیل می‌دهد.

[1] Translation Embedding



شکل ۱۴ بازنمایی شبکه‌ی ناهمگون در فضای برداری

روش TransE این مساله را به صورت یک مساله‌ی کمینه‌سازی حل می‌کند به این صورت که سعی می‌کند فاصله‌ی مجموع بردارهای موجودیت اول و رابطه $e_1 + r$ را با موجودیت دوم e_2 کمینه کند. همان‌طور که قبل‌تر گفتیم تابع امتیاز در این روش‌ها یک تابع فاصله است، پس در اینجا هدف کاهش فاصله بین $e_1 + r$ و e_2 است که بصورت (۲-۱۵) نوشته می‌شود.

$$d(e_1 + r.e_2) = \| e_1 + r - e_2 \| \qquad (۱۵-۲)$$

برای رسیدن به نتیجه‌ی بهینه نوآوری دیگری که در این روش معرفی شده است این است که تابع کمینه‌سازی را به گونه‌ای تغییر داده است که فاصله‌ی نمونه‌های اشتباه را نیز زیاد کرده است.

در زمانی آموزش پارامترها علاوه بر در نظر گرفتن این نکته که باید تابع فاصله‌ی نمونه‌های مثبت موجود در پایگاه دانش کمینه شود، سعی شده تا فاصله‌ی نمونه‌های منفی را نیز افزایش دهد. از آنجایی که پایگاه‌های دانش مورد استفاده از فرض جهان‌باز پیروی می‌کنند وقتی حقیقت $(e_1.r.e_2)$ در پایگاه‌دانش موجود نیست نمی‌توانیم نتیجه بگیریم که این سه‌گانه غلط است و آن را نمونه‌ی منفی تلقی کنیم.

در این روش برای ساختن نمونه‌های منفی، نمونه‌های مثبت مورد استفاده قرار گرفته‌اند به این صورت که یک‌بار موجودیت اول حذف شده و یک موجودیت تصادفی جای آن قرار داده شده است و یک‌بار موجودیت دوم حذف شده و یک موجودیت تصادفی جای آن قرار گرفته است، به این صورت به ازای هر نمونه‌ی مثبت دو نمونه‌ی منفی نیز تولید شده است (رابطه‌ی ۲-۱۶). در رابطه‌ی (۲-۱۷) مشاهده می‌شود که علاوه بر کاهش فاصله‌ی موجودیت‌های مثبت، یک جریمه هم برای نمونه‌های منفی در نظر گرفته شده است و هم‌چنین یک حاشیه γ بین نمونه‌های مثبت و منفی اضافه شده است.

$$S' = \{(\text{sub}'.\text{rel}.\text{obj})|\text{sub}' \in \varepsilon\} \cup \{(\text{sub}.\text{rel}.\text{obj}'|\text{obj}' \in \varepsilon\} \qquad (۱۶-۲)$$

$$\sum_{pos}\sum_{neq \in S'} [\gamma + \| s + r - o \|_2^2 - \| s' + r - o' \|_2^2] \qquad (۱۷-۲)$$

در این روش نیز مانند روش رسکال از SGD استفاده شده است که هم امکان آموزش به کمک نمونه‌برداری دسته‌ای[1] را فراهم می‌کند و هم از مشکل گیر کردن در نقاط بهینه‌ی[2] محلی جلوگیری می‌کند. الگوریتم کامل این روش در الگوریتم ۱ آمده است.

[1] Batch Sampling

[2] local optimom

input: Training set $S = \{(\text{sub,rel,obj})\}$, margin γ, learning rate λ
initialize $\mathbf{r} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each rel
    $\mathbf{r} \leftarrow \ell / \|\ell\|$ for each ℓ
    $\mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{j}}, \frac{6}{\sqrt{k}})$ for each entity ent(sub or obj)
loop
  $\mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$ for each entity ent
  $S_{batch} \leftarrow sample(S, b)$ //sample minibatch of size b
  $T_{batch} \leftarrow \emptyset$ //initialize set of pairs
  for (sub,rel,obj) $\in S_{batch}$ **do**
    (sub′,rel,obj′) ←sample(S'(sub,rel,obj)) //sample negative triplet
    $T_{batch} \leftarrow T_{batch} \cup \{((\text{sub,rel,obj}), (\text{sub}',\text{rel,obj}'))\}$
  end for
  Update embeddings w.r.t. $\sum_{T_{batch}} \nabla \left[\gamma + ||\mathbf{s} + \mathbf{r} - \mathbf{o}||_2^2 - ||\mathbf{s}' + \mathbf{r} - \mathbf{o}'||_2^2\right]_+$
end loop

الگوریتم ۱ الگوریتم روش TransE

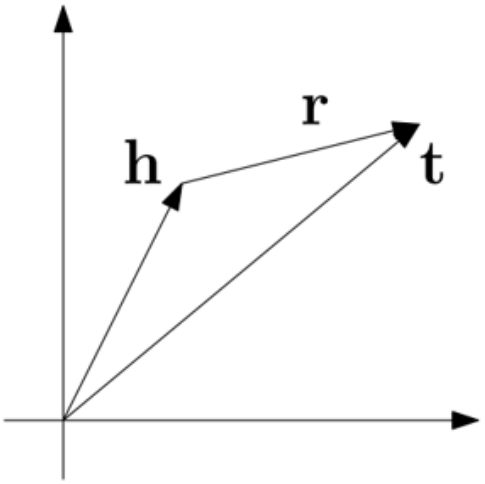
۸-۵-۲- مدل TransH[1]

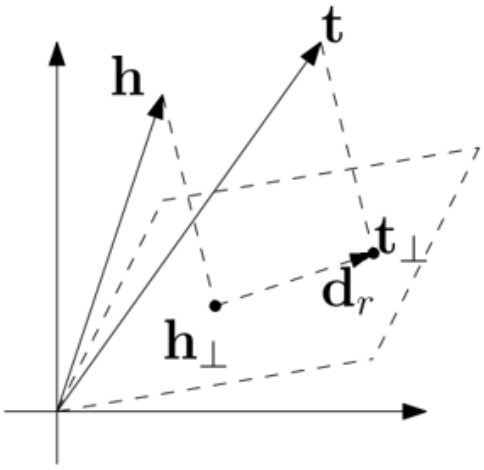
در بخش قبل مشاهده کردیم که TransE روش مناسبی را ارائه داد که با تعداد پارامتر کم و مقیاس‌پذیر قادر به آموزش مدل است. در [36] روشی معرفی شده است که به بررسی برخی از نگاشت‌ها مانند یک به چند، چند به یک، چند به چند و انعکاسی پرداخته است. در این روش رابطه‌ها به صورت یک ابرصفحه و یک بردار انتقال روی آن در نظر گرفته می‌شود، از همین رو به این روش، روش انتقال روی ابر صفحه یا به اختصار TransH گفته می‌شود.

در این روش موجودیت‌ها همچون روش قبل به صورت یک بردار از ویژگی‌های پنهان تعریف می‌شوند ولی رابطه‌ها به صورت دو بردار تعریف می‌شوند، یک بردار برای انتقال فضای مساله به ابرصفحه آن

[1] Translating on Hyperplanes

رابطه و دیگری بردار انتقالی است که در روش TransE نیز داشتیم و موجودیت‌های اول و دوم را به یکدیگر وصل می‌کرد. در روش قبل هدف این بود که دو موجودیت مشارکت کننده در یک حقیقت درست به وسیله‌ی بردار انتقال رابطه به یکدیگر وصل شوند و اگر حقیقت صحیح نیست انتظار داشتیم که این اتصال برقرار نباشد. در این روش نیز هدف به همین صورت است با این تفاوت که این انتقال برای هر رابطه روی یک ابرصفحه انجام می‌شود که نگاشتی از موجودیت‌های اول و دوم در آن قرار دارد. به کمک این روش ساده می‌توانیم نگاشت‌های یک به چند، چند به یک، چند به چند و انعکاسی را نیز در عملیات آموزش تاثیر دهیم در حالی که پیچیدگی و هزینه‌ی چندانی نسبت به روش TransE به مساله اضافه نمی‌کنیم.





شکل ۱۵-ب روش TransH

شکل ۱۶-الف روش TransE

همان‌طور که در شکل ۱۶-الف روش TransE مشاهده می‌شود روش TransE بردارهای h و t را به کمک بردار r به یکدیگر متصل می‌کرد اما در روش TransH همان‌طور که در شکل ۱۵-ب روش TransH نشان داده شده است ابتدا بردارهای h و t به صفحه‌ی مربوط به رابطه‌ی مورد نظر منتقل شده‌اند و توسط بردار dr که بردار رابطه است به یکدیگر متصل می‌شوند.

مقادیر $h_{\perp}$ و $t_{\perp}$ به صورت معادله‌ی (۲-۱۹) بدست می‌آیند که در آن w_r بردار انتقال به ابرصفحه‌ی مربوط به رابطه‌ی r است و d_r معادل بردار r در ابرصفحه‌ی رابطه است.

$$h_{\perp} = h - w_r^T h w_r. \qquad t_{\perp} = t - w_r^T t w_r \tag{۲-۱۸}$$

پس تابع امتیاز این روش به صورت زیر خواهد بود:

$$f_r(h.t) = ||(h - w_r^T h w_r) + d_r - (t - w_r^T t w_r)||_2^2 \tag{۲-۱۹}$$

در این روش نیز مانند روش TransE برای کمینه کردن خطا از حقیقت‌های صحیح و غلط و یک فاصله بین آن‌ها استفاده می‌شود که قصد دارد فاصله‌ی بین بردار h + r حقیقت‌های صحیح را با t کم و همین فاصله را در حقیقت‌های غلط زیاد کند. انتخاب حقیقت‌های غلط در این روش بمانند روش قبل است اما در این روش بجای استفاده از تابع امتیاز f_r که در معادله‌ی (۲-۱۷) مطرح کردیم از تابع امتیاز f_r که در (۲-۱۹) معرفی کردیم استفاده می‌کنیم و تابع کمینه‌سازی به صورت (۲-۲۰) خواهد شد.

$$f_r = \sum_{(h.r.t)\in\Delta} \sum_{(h'.r'.t')\in\Delta'_{(h.r.t)}} [f_r(h.t) + \gamma - f_{r'}(h'.t')]_+ \tag{۲-۲۰}$$

فصل ۳: قوانین انجمنی

در این فصل به معرفی قوانین انجمنی پرداخته و توضیح می‌دهیم چگونه از این قوانین برای بهبود مدل‌هایی که در فصل قبل معرفی شد استفاده می‌کنیم. در ابتدا برای سنجش کیفیت قوانین انجمنی معیارهای اطمینان را تعریف می‌کنیم، در ادامه نحوه‌ی استخراج این قوانین از پایگاه دانش را توضیح می‌دهیم و در انتها نیز قوانین هدفی که قصد استفاده از آن‌ها برای بهبود مدل داریم را مشخص می‌کنیم.

۳-۱- معیارهای اطمینان

برای این‌که بتوانیم به قانون‌هایی که استخراج کرده‌ایم اعتماد کنیم و از آن‌ها در آموزش مدل‌ها استفاده کنیم، نیاز به یک معیار اطمینان داریم. معیارهای رایج برای اندازه‌گیری دقت یک قانون میزان پشتیبانی قانون[1]، پوشش سر[2] و معیار اطمینان استاندارد[3] است.

[1] support rule

[2] head coverage

[3] Standard confidence

۱-۱-۳- معیار پشتیبانی قانون:

این معیار برای میزان اهمیت یک قانون استفاده می‌شود و به این صورت تعریف می‌شود: تعداد قوانین یافت شـــده در پایگاه دانش که یک حقیقت را نتیجه می‌دهند. برای مثال در (۳-۱) قانون R معرفی شده ا ست که م شخص می‌کند هر شخص x که در شهر y زندگی می‌کند، در آن شهر متولد شده اســـت. به تعداد دفعاتی که این دو حقیقت در پایگاه دانش اتفاق بیافتد و قانون R را نقض نکند، پشتیبان این قانون گفته می‌شود [37].

$$\text{R: livesIn(x, y)} \Rightarrow \text{wasBornIn(x, y)} \qquad (۱-۳)$$

این معیار به صورت رابطه‌ی (۳-۲۱) تعریف می‌شود.

$$supp(\overrightarrow{B} \Rightarrow r(x,y)) := \#(x,y) : \exists z_1, ..., z_m : \overrightarrow{B} \wedge r(x,y) \qquad (۲-۳)$$

در رابطه‌ی (۳-۲)، B مجموعه‌ای از حقیقت‌ها اســـت که حقیقت r(x,y) را نتیجه میدهد، به تعداد باری که این اتفاق می‌افتد معیار پشـــتیبانی قانون گفته می‌شـــود. در جدول ۲ نمونه‌ی پایگاه دانش متشـــکل از دو رابطه یک پایگاه دانش متشـــکل از دو رابطه و ۵ حقیقت وجود دارد. قانون (۱-۳) را در نظر بگیرید، میزان پشـــتیبان قانون R در این جدول برابر ۱ اســـت، بخاطر وجود حقیقت (Adam, LivesIn, Paris) و (Adam, wasBornIn, Paris) که از قانون R پیروی می‌کنند [AMIE].

جدول ۲ نمونه‌ی پایگاه دانش متشکل از دو رابطه

livesIn	wasBorIn
(Adam, Paris)	(Adam, Paris)
(Adam, Rome)	(Carl, Rome)
(Bob, Zurich)	

۲-۱-۳- معیار پوشش سر

معیار پشتیبان قانون یک مقدار مطلق است و برای تعریف کردن یک آستانه برای این معیار نیاز است تا اندازه‌ی پایگاه دانش را نیز بدانیم. مثلا اگر پشتیبان یک قانون در یک پایگاه دانش با ۲۰۰۰ حقیقت ۵۰۰ باشد، عدد بسیار بزرگی است اما اگر در یک پایگاه دانش با چندین میلیون حقیقت ۵۰۰ باشد عدد قابل توجهی نیست. برای حذف این وابستگی معیار پوشش سر را به صورت زیر معرفی می‌کنیم.

$$hc(\overrightarrow{B} \Rightarrow r(x,y)) := \frac{supp(\overrightarrow{B} \Rightarrow r(x,y))}{size(r)} \qquad (۳-۳)$$

$$size(r) := \#(x',y') : r(x',y') \qquad (۴-۳)$$

در این معیار بررسی می‌شود که چند درصد از r(x,y) هایی که اتفاق افتاده است بخاطر وجود زنجیره‌قانون B بوده است. در جدول ۲ مقدار معیار پوشش سر بخاطر وجود (Carl, Rome) برابر با ۰,۵ است [37].

۳-۱-۳- معیار اطمینان استاندارد

معیارهای قبل اهمیت قانون را بررسی می‌کردند و فقط پیش‌بینی‌های درست از قانون را در نظر می‌گرفتند، و پیش‌بینی‌های غلطی که این قانون تولید می‌کند را در نظر نمی‌گیرند. پس ما به معیاری نیاز داریم که کیفیت قانون را نیز بررسی کند [37].

در این معیار میزان پشتیبان هر قانون را بر تعداد دفعاتی که قسمت بدنه‌ی این قانون در پایگاه دانش دیده‌شده است تقسیم می‌کنیم. این کار باعث می‌شود که معیاری داشته باشیم برای این‌که بدانیم در چند درصد مواقع که قسمت بدنه‌ی این قانون برقرار بوده است منجر به تولید قسمت نتیجه شده است.

$$conf(\overrightarrow{B} \Rightarrow r(x,y)) := \frac{supp(\overrightarrow{B} \Rightarrow r(x,y))}{\#(x,y) : \exists z_1, ..., z_m : \overrightarrow{B}} \quad (۵-۳)$$

رابطه‌ی بالا به زبان ساده برابر است با تعداد رخداد قانون بخش بر تعداد دفعاتی که می‌توانست رخ دهد. برای مثال در پایگاه دانش معرفی شده در جدول ۲ معیار اطمینان استاندارد برابر $\frac{1}{3}$ است. زیرا wasBornIn(Adam, Paris) به عنوان نمونه‌ی مثبت[1] در نظر گرفته شده و wasBornIn(Adam, Rome) و wasBorn(Bob,Zurich) که در پایگاه دانش وجود ندارند نمونه‌ی منفی[2] در نظر گرفته شده است. که معیار اطمینان استاندارد برابر می‌شود با تعداد نمونه‌ی مثبت یعنی ۱ بخش بر تعداد کل نمونه‌ها یعنی ۳.

این معیار در پایگاه‌های دانش فرض جهان‌بسته معیار خوبی است و به خوبی دقت قانون استخراج شده را مشخص می‌کند زیرا همان‌طور که در مثال بالاتر هم مشاهده کردیم، عدم وجود یک نمونه‌ی مثبت از حقیقت در پایگاه دانش دلیلی بر غلط بودن آن حقیقت بوده است. اما همان‌طور که در بخش ۳-۱-۲- اشاره کردیم فضای این مساله فرض جهان‌باز است و حقیقت‌هایی که در این پایگاه‌های دانش وجود ندارند لزوما غلط نیستند [37]. برای درک بیشتر این موضوع به مثال زیر دقت کنید:

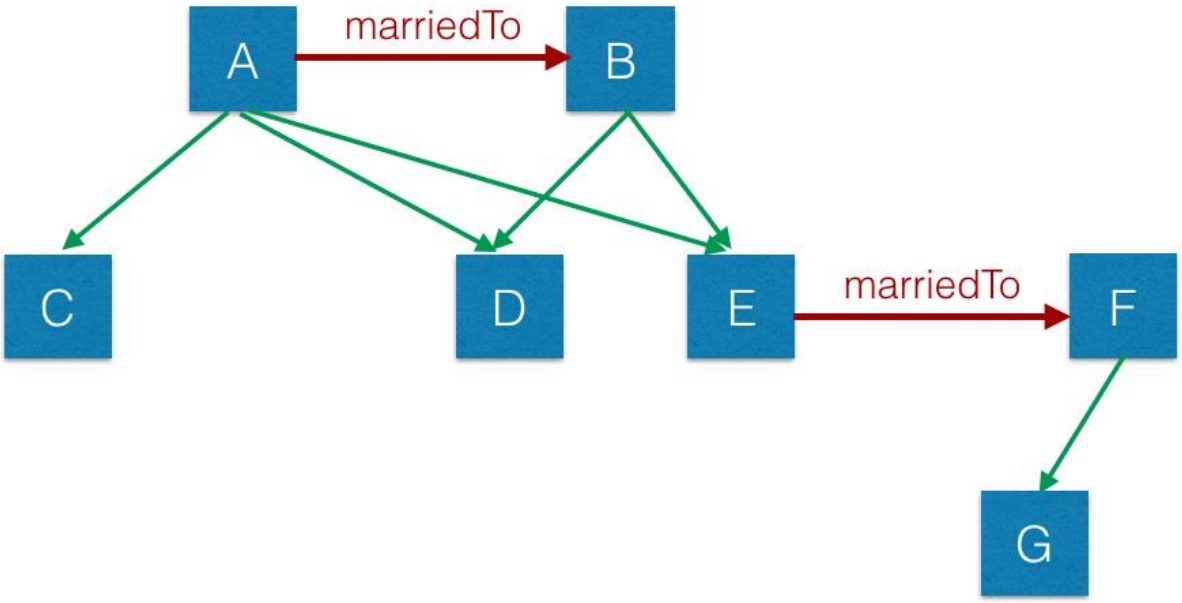
به گراف ارائه شده در شکل ۱۷ دقت کنید. در این گراف رابطه های افقی رابطه‌ی ازدواج (marriedTo) و روابط عمودی رابطه‌ی داشتن فرزند (hasChild) را مشخص می‌کنند. در این گراف قصد داریم که قانون R با تعریف زیر را بررسی کنیم:

$$hasChild(y.x).marriedTo(y.z) => hasChild(z.x) \quad (۶-۳)$$

این قانون به این معناست که اگه شخص y فرزندی به نام x داشته باشد و همین شخص با شخص دیگری به نام z ازدواج کرده باشد، می‌توان نتیجه گرفت که شخص z هم فرزندی به نام x دارد.

[1] positive example

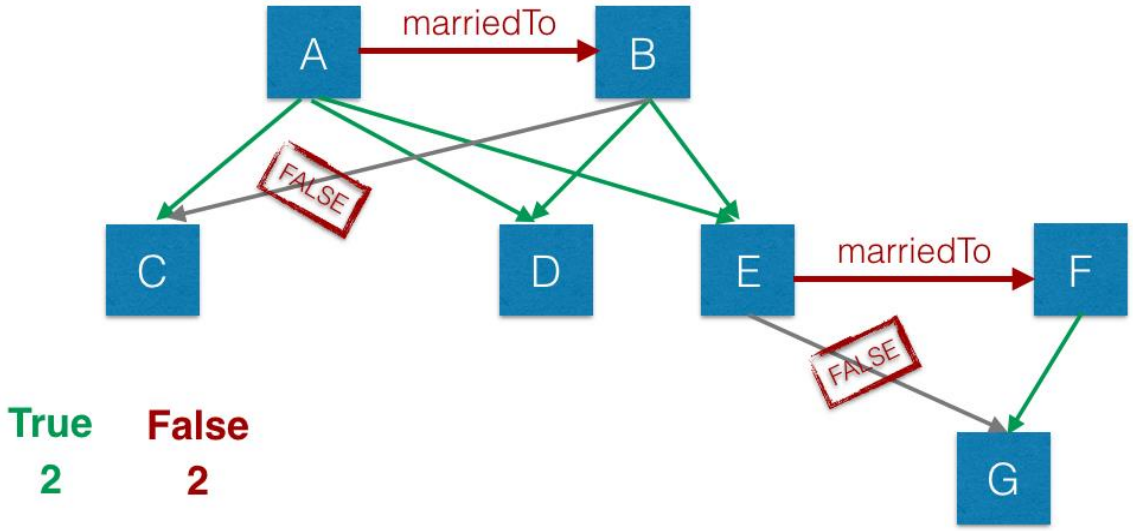
[2] negative example



شکل ۱۷ نمونه گراف ناهمگون با دو نوع رابطه

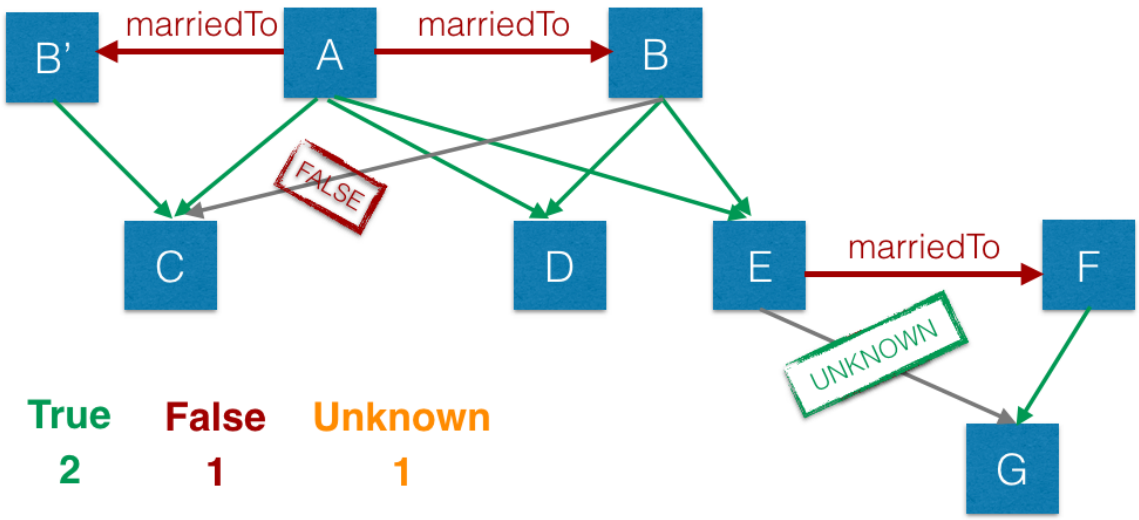
اگر بخواهیم معیار اطمینان استاندارد در این گراف را بررسی کنیم، مشاهده می‌شود که این قانون ۲بار در این گراف صدق کرده است در حالی که می‌توانست ۴ بار اتفاق بیافتد. در شکل ۱۸ دفعاتی که این قانون باید اتفاق می‌افتاده اســت و اتفاق نیفتاده اســت نمایش داده شــده و به عنوان نمونه‌ی منفی در نظر گرفته شده است. با این اوصاف معیار اطمینان استاندارد قانون R در گراف معرفی شده برابر ۰,۵ می‌شود.

$$Standard\ Confidence = \frac{\sup(B \Rightarrow r(x.y))}{\#(x.y):B} = 2 / 4 = 0.5 \qquad (۷-۳)$$



شکل ۱۸ حقیقت‌هایی که در گراف نمونه موجود نیست

اما همان‌طور که توضیح دادیم عدم وجود حقایق در پایگاه‌های دانش فرض جهان‌باز دلیل بر غلط بودن این حقیقت‌ها نیست. برای مثال فرض کنید که در گراف معرفی شده یک گره‌ی B’ داشته باشیم که رابطه‌ی ازدواج دا شته با شد با گره‌ی A و رابطه‌ی فرزندی دا شته با شد با گره‌ی C پس می‌دانیم که رابطه‌ی فرزندی بین گره‌ی B و C برقرار نیست اما در مورد وجود و عدم وجود این رابطه بین E و G اطلاعی نداشته باشیم (شکل ۱۹).



شکل ۱۹ حالت ناشناس برای پیوندهایی که اطلاعی در مورد آن نداریم

پس معیار اطمینان اســتاندارد برای پایگاه‌های دانش جهان‌باز معیار مناســبی نیســت و نیاز به تعریف معیار بهتری داریم. در [AMIE] معیار اطمینان با فرض نیمه کامل[1] (PCA-Confidence) را معرفی کرده است.

۴-۱-۳- معیار اطمینان با فرض نیمه‌کامل

در این معیار سعی شده نمونه‌هایی که در پایگاه دانش وجود ندارند و در معیار اطمینان استاندارد آن‌ها را نمونه‌ی غلط در نظر می‌گرفتیم را در اینجا بیشــتر بررســی کنیم و با احتمال بهتری غلط بودن یا نبودن آن را مشخص کنیم [37].

در این معیار اگر حقیقت r(x,y) در پایگاه دانش موجود نبود، بر سی می شود که y’ی وجود دارد که برای آن حقیقت r(x,y’) برقرار باشد یا خیر، اگر وجود داشته باشد در نظر می‌گیرد که r(x,y) غلط بوده و در غیر این صورت این حقیقت را ناشناخته فرض می‌کند.

$$conf_{pca}(\overrightarrow{B} \Rightarrow r(x,y)) := \frac{supp(\overrightarrow{B} \Rightarrow r(x,y))}{\#(x,y) : \exists z_1, ..., z_m, y' : \overrightarrow{B} \wedge r(x,y')} \qquad (۸-۳)$$

به مثال جدول ۲ برگردیم، در این جدول wasBornIn(Adam,Paris) را یک نمونه‌ی مثبت در نظر می‌گیریم و wasBornIn(Adam,Rome) را یک نمونه‌ی منفی در نظر می‌گیریم اما این که livesIn (Bob, Zurich) داریم و در مورد محل تولد آن اطلاعی نداریم دلیل نمی‌شــود که این نمونه را یک نمونه‌ی منفی در نظر بگیریم زیرا ممکن است bob اصلا متولد نشده باشد. پس در این مثال مقدار PCA-Confidence = $\frac{1}{2}$ می‌شود.

یا در گرافی که در شکل ۱۹ معرفی شده است، در معیار اطمینان استاندارد هر دو رابطه‌ی بین (B,C) و (E,G) را غلط در نظر گرفتیم، اما در اینجا داریم که C مادری به نام B’ دارد و می‌توانیم مطمئن شــویم که رابطه‌ی فرزندی بین B و C برقرار نیســت و این رابطه را غلط در نظر بگیریم. اما برای رد رابطه‌ی E و G هیچ مدرکی نداریم و این رابطه را غلط در نظر نمی‌گیریم بلکه ناشــناخته در نظر

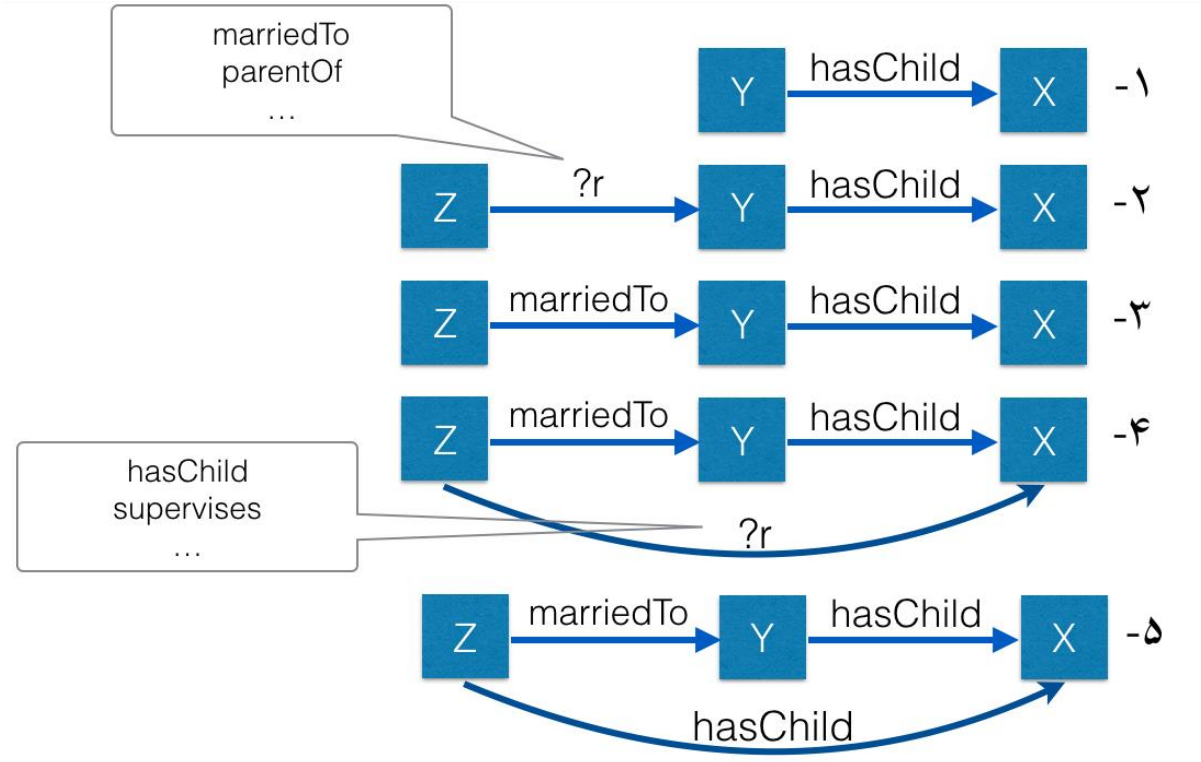
[1] Partial Completeness Assumption

می‌گیریم و از آن در فرمول PCA-confidence استفاده نمی‌کنیم. پس PCA-Confidence در این مثال برابر با X است.

۲-۳- الگوریتم‌ها

برای استخراج قوانین انجمنی از الگوریتم‌های معرفی شده در [37] و [38] استفاده شده است که در ادامه توضیح مختصری در مورد بخشی از این الگوریتم‌ها خواهیم داد.

نحوه‌ی استخراج قوانین در شکل ۲۰ نشان داده‌شده است که هر مرحله را در ادامه توضیح خواهیم داد.



شکل ۲۰ نحوه‌ی استخراج قوانین انجمنی از حقیقت‌های پایگاه دانش

به ازای هر حقیقت این روند یک بار تکرار می‌شود:

۱- حقیقت انتخاب شده را در نظر می‌گیریم. در این مثال حقیقت (Y, hasChild, X) در نظر گرفته شده است.

۲- تمامی روابطی که بین یکی از موجودیت‌ها و موجودیت دیگری وجود دارد را بررسی می‌کنیم. در این مثال روابط موجود برای موجودیت اول یعنی Y نمایش داده شده است، برای مثال روابط ازدواج و والد بودن با این موجودیت مورد بررسی قرار گرفته است.

۳- همه‌ی روابطی که در مرحله‌ی ۲ کاندید شده بودند را در کنار Y قرار می‌دهیم و یک زنجیر از قوانین را ایجاد می‌کنیم. زنجیر ایجاد شده در این قسمت به صورت رابطه‌ی ازدواج بین Z و Y و رابطه‌ی داشتن فرزند بین Y و X است.

۴- در این قسمت همه‌ی روابط بین موجودیت اضافه شده به زنجیر و موجودیت اول را بررسی می‌کنیم یعنی روابط بین Z و X. روابط کاندید در این مثال روابط «داشتن فرزند» و «ناظر بودن» است.

۵- در این مرحله روابطی که کاندید شده‌اند را بررسی می‌کنیم و معیارهای اطمینان و اهمیت که در بخش قبل معرفی کردیم را برای آن‌ها بدست می‌آوریم، در صورتی که معیارها مقدار قابل قبولی داشته باشند قانون تولید شده را ذخیره می‌کنیم و در غیر این صورت از آن رد می‌شویم.

۶- پس از بررسی همه‌ی روابط کاندید تولید شده در بخش ۴ کل زنجیر تولید شده را به مرحله‌ی ۱ ارسال می‌کنیم و با اضافه کردن یک حقیقت دیگر به ابتدا یا انتهای آن روند رشد زنجیر را تا آستانه‌ی تعریف شده ادامه می‌دهیم.

به کمک الگوریتم معرفی شده در بالا همه‌ی قانون‌های ممکن در پایگاه دانش با توجه به معیارهای کیفیت و اهمیت استخراج خواهند شد. اما با توجه به اندازه‌ی بزرگ پایگاه‌های دانش مورد استفاده، این روش برای بررسی همه‌ی حالت‌های ممکن بسیار وقت‌گیر خواهد بود، پس بهتر است در مرحله‌ی ۲ و ۴ که به انتخاب کاندید می‌پردازیم بجای بررسی همه‌ی حقیقت‌های موجود در پایگاه

داده یک عملیات هرس[1] روی کاندیدها با توجه به هم‌رخدادی با حقیقت موجود انجام شود و همه‌ی حقیقت‌ها مورد بررسی قرار نگیرند.

با اعمال این الگوریتم روی پایگاه دانش Freebase15k که شامل حدود ۵۰۰هزار حقیقت است، تعداد ۴۱۱۹۶ قانون استخراج شد. در ادامه چند مثال از این قانون‌ها را بررسی می‌کنیم.

در قانون زیر داریم که اگر b یک تیم فوتبال باشد و بازیکنی به نام a داشته باشد، می‌توانیم نتیجه بگیریم که تیم فعلی بازیکن a، تیم b است. معیار اطمینان PCA در این مثال ۰,۸۹ است که مقدار قابل اعتمادی است.

?b /sports/soccer/team/player ?a

⇩

?a /soccer/current_team/team ?b

در مثال دیگر داریم که اگر بازیگر سینمای a جایزه‌ی b را برنده شده باشد، نتیجه می‌گیریم که این بازیگر برای جایزه‌ی b نامزد شده است.

?b /award/awards_won ?a

⇩

?a /award/award_nomination/nominated_for ?b

قوانین پیچیده‌تری نیز استخراج شده است که بدنه‌ی قانون[2] از چند حقیقت تشکیل شده است. یک نمونه از این قوانین در ادامه آورده شده که در این قانون داریم اگر a یک موسسه‌ی آموزشی

[1] Pruning

[2] rule body

باشد که در مکان f قرار دارد و داشته باشیم که f در مکان b واقع شده است، می‌توانیم نتیجه بگیریم موسسه‌ی آموزشی a نیز در مکان b قرار دارد. علیرغم پیچیدگی‌ای که این قانون نسبت به دو قانون قبلی دارد، اما معیار اطمینان PCA این قانون ۰,۹۳ بوده و بسیار قابل اعتماد است و می‌توانیم از آن در آموزش مدل‌ها استفاده کنیم.

?a /educational_institution/located_in ?f

?f /location/location/containedby ?b

⇩

?a /location/location/containedby ?b

در بخش بعدی انواع این قوانین را بررسی کرده و توضیح خواهیم داد که چگونه از این قوانین در بهبود مدل‌های مبتنی بر ویژگی‌های پنهان استفاده خواهیم کرد.

۳-۳- قوانین هدف

در بخش قبل مشاهده شد که چگونه قوانین از پایگاه‌های دانش استخراج می‌شوند و چند نمونه از این قوانین را مرور کردیم. در این بخش ۵دسته‌بندی از قوانینی که استخراج می‌شوند را معرفی می‌کنیم و در ادامه بررسی‌های انجام شده روی روش‌های موجود را روی این دسته‌بندی‌ها اعمال می‌کنیم و نقاط قوت وضعف هر روش را نسبت به این دسته‌بندی‌ها می‌سنجیم [39].

۱-۳-۳- **قانون** R-subsumption

این قانون به صورت زیر تعریف می‌شود، به این معنی که اگر x و y رابطه‌ی r را باهم داشته باشند، رابطه‌ی r’ نیز بین آن‌ها برقرار است.

$$r(x, y) => r'(x, y)$$

برای مثال اگر در پایگاه دانش رابطه‌ی پدر بودن را داشته باشیم، می‌توانیم از آن رابطه‌ی والد بودن را نتیجه بگیریم، مثلا اگر حقیقت «علی، پدر، حسن» به معنی «علی پدر حسن است» را داشته باشیم، می‌توانیم نتیجه بگیریم که حقیقت «علی، والد، حسن» نیز حقیقت درستی است.

۲-۳-۳- **قانون** R-equivalence

این قانون رابطه شبیه به قانون قبلی است با این تفاوت که رابطه‌ی برگشت نیز بین دو طرف قانون برقرار است.

$$r(x, y) <=> r'(x, y)$$

این قانون در آموزش مدل‌ها کمک بسیاری می‌تواند بکند زیرا ماهیت این قانون به این صورت است که روابط یکسان (روابطی که به دو صورت در پایگاه دانش استفاده شده اند اما یک معنی می‌دهند) را شناسایی می‌کند، برای نمونه مثالی که در بخش قبل زدیم، دو رابطه‌ی located_in و contained_by در توالی یکدیگر آمده بودند که یک معنی را می‌دهند، پس می‌توانیم نتیجه بگیریم که اگر حقیقت (x, located_in, y) را داشته باشیم، حقیقت (x, contained_by, y) را نیز داریم و در عملیات آموزش روش از یکی از این دو مفهوم استفاده کنیم که باعث کاهش تعداد روابط موجود در پایگاه دانش و در نتیجه کاهش پیچیدگی مساله می‌شود.

۳-۳-۳- **قانون** 2-hope translation

این قانون، همان قانون تعدی است، به این صورت که اگر x و y رابطه‌ی r_1 را باهم داشته باشند و هم‌چنین y و z نیز رابطه‌ی r_2 را داشته باشند، می‌توانیم نتیجه بگیریم که x و z نیز با یکدیگر در ارتباطند.

$$r_1(x, y), r_2(y, z) \Rightarrow r'(x, z)$$

نمونه‌ای از این قانون را در مورد مناطق جغرافیایی در بخش قبل مشاهده کردیم. مثلا می‌دانیم که شهر تهران در کشور ایران قرار دارد، و کشور ایران در منطقه‌ی خاورمیانه قرار دارد، پس می‌توانیم نتیجه بگیریم که شهر تهران در منطقه‌ی خاورمیانه قرار دارد.

۳-۳-۴- **قانون** Triangle alignment

این قانون نیز مانند قانون قبلی در سمت بدنه‌ی قانون دو حقیقت را بررسی می‌کند و از روی آن‌ها حقیقت جدید را نتیجه می‌گیرد.

$$r(x, z), r(y, z) \Rightarrow r'(x, y)$$

مشاهده می‌شود که در این قانون اشتراک موجودیت‌های دوم مد نظر قرار گرفته است و در صورتی که دو حقیقت در موجودیت دوم مشترک باشند بررسی می‌شود که بین موجودیت‌های اول نیز رابطه‌ای برقرار است یا خیر. برای مثال اگر داشته باشیم که hasChild(x, z) و hasChild(y, z) به این معنی که هم x و هم y فرزندی به نام z داشته باشند، می‌توان نتیجه گرفت که marrid(x, y) یعنی x و y زن و شوهر هستند.

۵-۳-۳- **قانون** Specific R-subsumption

این قانون حالت کامل‌تری از قانون اول یعنی R-sub است به این صورت که بجز بررسی رابطه‌ها، ویژگی‌های موجودیت اول را نیز بررسی می‌کند. برای مثال در قانون اول داشتیم اگر شخصی پدر x باشد می‌توان نتیجه گرفت که آن شخص والد x نیز هست، اما عکس این قضیه را نمی‌توان نتیجه گرفت، یعنی نمی‌توان نتیجه گرفت که اگر شخصی والد x است پس پدر x است زیرا ممکن است مادر x باشد. در این قانون یک ویژگی از موجودیت اول نیز بررسی می‌شود.

$$r_1(x, y), r_2(x, v) \Rightarrow r'(x, y)$$

مثلا اگر داشته باشیم که شخصی والد x است و جنسیت آن شخص مذکر است، می‌توان نتیجه گرفت که آن شخص پدر x است و اگر جنسیت مونث داشت مادر x است.

در جدول ۳ قوانین معرفی شده به اختصار آمده‌اند.

جدول ۳ قوانین انجمنی هدف

Body rule		Target rule	name
r(x, y)	=>	r'(x, y)	R-subsumption
r(x, y)	<=>	r'(x, y)	R-equivalence
r1(x, y), r2(y, z)	=>	r'(x, z)	2-hope translation
r(x, z), r(y, z)	=>	r'(x, y)	Triangle alignment
r1(x, y), r2(x, v)	=>	r'(x, y)	Specific R-sub

۴-۳- جمع‌بندی

در این فصل ابتدا به معیارهای اطمینان برای سنجش کیفیت قانون‌های استخراج شده پرداختیم و ۴معیار اطمینان را معرفی کردیم:

- معیار پشیتبانی قانون (support)
- معیار پوشش سر (head cover)
- معیار اطمینان استاندارد (standard confidence)
- معیار اطمینان با فرض نیمه‌کامل (PCA Confidence)

که از بین این معیارها معیار چهارم برای پایگاه‌های دانش نیمه‌کامل مناسب بوده و در ادامه‌ی پژوهش برای سنجش کیفیت قانون‌ها از این معیار استفاده می‌کنیم.

در بخش ۲-۳- الگوریتم استخراج قوانین موجود در پایگاه دانش را توضیح دادیم و چند نمونه از قوانین استخراج شده را بررسی کردیم. در بخش ۳-۳- دسته‌بندی‌ای روی قوانین استخراج شده انجام دادیم و آنها را به ۵ دسته تقسیم کردیم که هر یک روش‌های موجود را از یک جنبه‌ی خاص بهبود می‌دادند و با این دسته‌بندی در ادامه قادر خواهیم بود که هر روش را از جنبه‌های مختلف بسنجیم و نقاط ضعف و قوت آن را بررسی کنیم.

فصل ۴: روش پیشنهادی

در این فصل عملکرد روش‌های موجود از نظر مقیاس‌پذیری و نتایج بدست آمده از هرکدام را مورد بررسی قرار خواهیم داد و در ادامه‌ی فصل شرح می‌دهیم که چگونه می‌توانیم از قوانین انجمنی جهت بهبود این روش‌ها استفاده کنیم. در فصل بعد روش‌های پیشنهادی در این فصل را در آزمایش‌های جداگانه بررسی خواهیم کرد.

۴-۱- عملکرد روش‌های موجود

در این بخش نتایج بدست آمده از روش‌های معرفی شده در بخش قبل را مورد بحث و بررسی قرار می‌دهیم. روش‌هایی که در این پایان‌نامه مورد بررسی و بهبود قرار گرفته‌اند: ۱- روش رسکال از روش‌های دوخطی ۲- روش NTN از روش‌های ادراک چندلایه‌ای ۳- روش SE ۴- روش TransE و ۵- روش TransH از روش‌های فاصله‌ی پنهان هستند.

در ادامه آزمایشاتی روی این ۵ روش انجام شده است و این روش‌ها از نظر مقیاس‌پذیری و میزان کیفیت خروجی که به کمک معیارهای رتبه‌ی میانگین و hit@10 محاسبه می‌شود مورد بررسی قرار گرفته‌اند. همه‌ی این آزمایشات روی پایگاه دانش Freebase15k انجام شده است که شامل حدود ۱۵هزار موجودیت، ۱۳۰۰ رابطه و در مجموع ۶۰۰ هزار حقیقت است. تقسیم‌بندی داده‌های آموزش، ارزیابی و آزمون به شکلی که در بخش ۳-۲- توضیح داده شد انجام شده است.

۴-۱-۱- روش رسکال

تعداد پارامترها: در این روش برای هر موجودیت یک بردار d بعدی و برای هر رابطه یک ماتریس وزن‌دار d×d بعدی در نظر گرفتیم. پس تعداد پارامترهایی که در این روش نیاز است که آموزش دهیم از مرتبه‌ی $O(n_e d + n_r d^2)$ است. در آزمایش‌ها تعداد ابعادی که بهترین نتیجه را بدست می‌دهد d=250 است. اگ بردارهای مساله را ۲۵۰ بعدی در نظر بگیر باید به تعداد ۸۸ میلیون پارامتر آموزش داده شود ($n_e d + n_r d2 = 15k\times250 + 1.3k\times250\times250 = 88m$).

کیفیت خروجی: پس از اعمال مدل آموزش دیده‌ی رسکال روی مجموعه داده‌ی آزمون ۵۰هزار حقیقتی، میزان ۴۲,۱٪ پاسخ‌های داده شده توسط این مدل زیر رتبه‌ی ۱۰ بوده (hit@10) و میانگین رتبه‌ی کل پاسخ‌هایی که داده شده (mean rank) ۶۸۳ است.

۴-۱-۲- روش NTN

تعداد پارامترها: در این روش نیز برای هر موجودیت یک بردار d بعدی در نظر گرفته شده و برای هر رابطه یک ماتریس B_k و دو بردار A_k و w_k در نظر گرفته می‌شود که در مجموع برای هر رابطه d^3 بعد در نظر گرفته می‌شود و تعداد کل پارامترهایی که باید آموزش دیده شوند از مرتبه‌ی $O(n_e d + n_r d^3)$ است. تعداد پارامتری که در آزمایشات برای این روش نتیجه‌ی مناسبی بدست می‌دهد d=50 است. برای این تعداد ویژگی پنهان تعداد کل پارامتری که باید آموزش داده شوند تقریبا ۱۶۵میلیون پارامتر می‌شود که نسبت به روش رسکال با این‌که تعداد ابعاد بردارها بسیار کمتر است، تقریبا دو برابر است ($n_e d + n_r d^2 = 15k\times50 + 1.3k\times50\times50\times50$ 163,250k).

کیفیت خروجی: مقدار hit@10 در این روش ۲۷٪ و میانگین رتبه‌ی همه‌ی پاسخ‌های این روش ۱۶۴ است. همان‌طور که در معرفی این روش گفته شد این روش به سرعت به بیش‌برازش[1] میل می‌کند و تاثیر این امر در میزان hit@10 مشخص است اما میانگین رتبه این روش به میزان خوبی از روش رسکال بهتر است، که نشان می‌دهد روش رسکال برای سوالاتی که به میزان کافی نمونه برای آموزش نداشته است، نتایج خیلی دور از ذهنی بدست می‌دهد با این حال برای ۴۲٫۱٪ حقیقت‌ها که روابط بین آن‌ها به خوبی آموزش دیده شده است، پاسخ‌های درست در ۱۰ پاسخ اول بوده است.

۳-۱-۴- روش Structured Embedding

تعداد پارامتر: در این روش برای هر موجودیت یک بردار d بعدی و برای هر رابطه دو ماتریس d×d بعدی آموزش دیده می‌شود، یکی برای انتقال موجودیت اول و دیگری برای انتقال موجودیت دوم. بنابراین تعداد پارامتر این روش از مرتبه‌ی $O(n_e d + 2n_r d^2)$ است. برای آموزش این مدل نیز بردارهایی با اندازه d=50 کفایت می‌کند و برای آموزش این مدل باید حدود ۸میلیون پارامتر آموزش دیده شوند.

$$(n_e d + 2n_r d^2 = 15k \times 50 + 2 \times 1.3k \times 50 \times 50 = 7.250k)$$

کیفیت خروجی: در این روش hit@10 برابر با ۳۹٫۸٪ است و مقدار میانگین رتبه ۱۶۲ بوده است. مشاهده می‌شود که علی‌رقم کاهش بسیار زیادی که در تعداد پارامترهای مساله نسبت به دو روش قبل داشته‌ایم، نتیجه‌ی خوبی حاصل شده است و در معیار hit@10 با روش رسکال و در معیار میانگین رتبه با روش NTN رقابت می‌کند.

[1] overfit

۴-۱-۴- روش TransE

تعداد پارامتر: در این روش تعداد پارامترها به میزان بسیار خوبی کاهش پیدا کرده است، همان‌طور که در معرفی این روش مشاهده شد، برای هر موجودیت یک بردار d بعدی در نظر گرفته شد و برای روابط از ماتریس استفاده نشده و به عنوان یک بردار انتقال در کنار موجودیت اول در نظر گرفته شده است که باعث می‌شود تعداد پارامتری که برای رابطه‌ها نیز داریم نیز d بعد باشد. پس تعداد پارامتری که در این روش باید آموزش دیده شود از مرتبه‌ی $O(n_e d + n_r d)$ است که در مجموع تعداد پارامترهای مساله را بسیار کاهش می‌دهد و همین مساله روی همین پایگاه دانش را با حدود ۸۰۰ هزار پارامتر حل می‌کند. ($n_e d + n_r d = 15k \times 50 + 1.3k \times 50 = 815k$).

کیفیت خروجی: این روش علی‌رقم کاهش چشم‌گیری که در تعداد پارامترها داشت و مقیاس‌پذیری بسیار بالایی که ایجاد می‌کند، در نتایج بدست آمده نیز بهتر از روش‌های معرفی شده‌ی قبلی کار کرده است. در این روش hit@10 برابر ۴۵٫۱٪ و میانگین رتبه برابر ۱۲۵ است که مشاهده می‌شود در این روش هم جواب‌های درست بسیار بیشتر بوده و هم داده‌های دور از انتظار بسیار کمتر شده است و جواب سوال‌ها به جواب‌های منطقی نزدیک‌تر شده است.

۴-۱-۵- روش TransH:

تعداد پارامتر: همان‌طور که در بخش ۸-۵-۲- دیدیم این روش از نظر عملیات آموزش بسیار شبیه به روش TransE است و در تعداد پارامترهایی که باید آموزش داده شود فقط یک بردار انتقال به

ابرصفحه‌ی مربوط به رابطه را بیشتر دارد که به ازای هر رابطه k پارامتر به عملیات آموزش اضافه می‌کند، پس پارامترهایی که باید آموزش دیده شوند از مرتبه‌ی $O(n_e k + 2n_r k)$ هستند. تعداد پارامتر این روش روی پایگاه دانش freebase15k حدود ۸۸۰ هزار خواهد بود:

$$(n_e d + 2n_r d = 15k \times 50 + 2 \times 1.3k \times 50 = 880k)$$

مشاهده می‌شود که این روش افزایش چشم‌گیری در تعداد پارامترهای مساله نداشته اما نتایج بسیار دقیق‌تری بدست آورده است.

کیفیت خروجی: این روش با درنظر گرفتن روابط پیچیده‌تری مانند روابط یک به چند، چند به چند، چند به یک و بازگشتی توانسته است دقت بهتری نسبت به روش‌های نام برده شده‌ی قبلی به دست آورد و با افزایش منطقی تعداد پارامترها نسبت به روش TransE مقدار Hit@10 به ۶۴,۴٪ و میانگین رتبه‌ی ۸۷ برسد.

۲-۴- نقاط قوت و ضعف روش‌های موجود

در این بخش به بررسی نقاط قوت و نقاط ضعف روش‌های موجود می‌پردازیم. در بخش ۱-۴- به تفصیل در مورد مقیاس‌پذیری و معیارهای کیفیت روش‌ها صحبت شد، در این بخش میزان همبستگی و ارتباط موجودیت‌ها و رابطه‌ها را بررسی می‌کنیم که چه مقدار این همبستگی و ارتباط‌ها در مدل‌های آموزش دیده شده درک شده است.

روش‌هایی که معرفی شداند در بهترین حالت تا ۶۴,۴٪ دقت در یافتن پاسخ در ۱۰ جواب اول (hit@10) را داشتند. نمونه‌هایی از این سوال و ۱۰ جواب اول مدل TransE برای آن را در جدول ۴ مشاهده می‌کنیم.

جدول ۴ نمونه‌ی سوال از مدل TransE و ده پاسخ اول [40]

Lil Wayne born in?! (Lil_Wayne, born_in, ?)	New Orleans, Atlanta, Austin, St. Louis, Toronto, New York City, Wellington, Dallas, Puerto Rico
WALL-E has the genre?! (WALL-E, has_genre, ?)	Animations, Computer Animation, Comedy film, Adventure film, Science Fiction, Fantasy, Stop motion, Satire, Drama

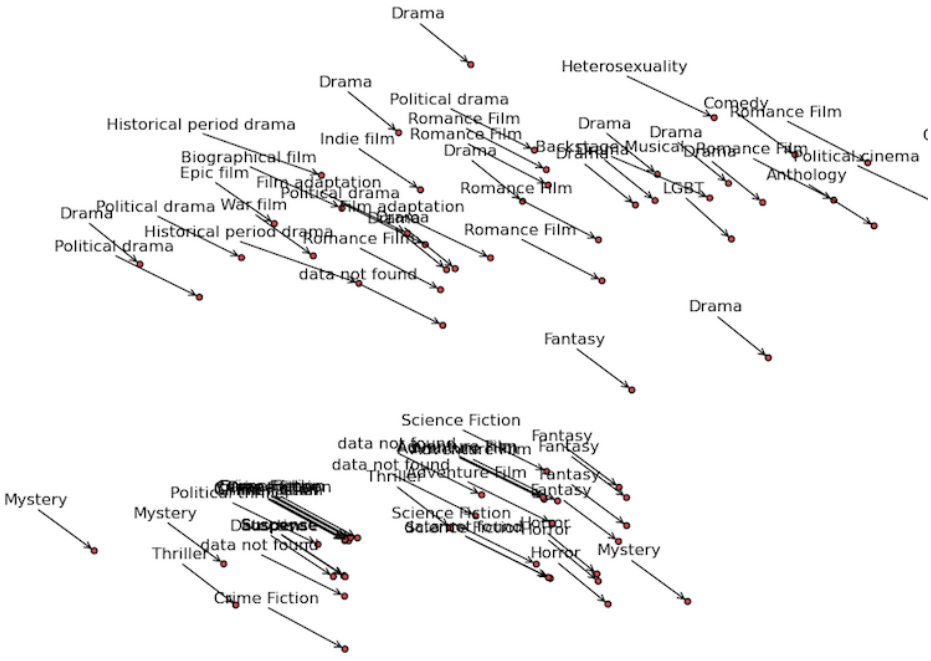
همان‌طور که مشاهده می‌شود جواب‌هایی که داده شده است همبستگی خوبی دارند، برای مثال در نمونه‌ی اول که از مدل پرسیده شده است Lil Wayne که یک خواننده‌ی آمریکایی است متولد کجاست، تمام ۱۰ جواب اول همان‌طور که انتظار می‌رود ایالت‌های مختلف آمریکا هستند و پاسخ اول یعنی New Orleans جواب درست می‌باشد و جواب‌ها منطقی هستن و مدل TransE این رابطه که پاسخ سوال «متولد کجا است؟» باید یک شهر باشد را خوب فهمیده است. همینطور در مثال دوم وقتی سوال این‌که «ژانر یک فیلم چیست؟» از مدل پرسیده شده است، جواب‌ها همبستگی خوبی دارند و جواب دور از ذهن داخل آن‌ها نیست.

برای مشاهده‌ی بهتر این موضوع، پس از آموزش دیدن مدل TransE ، بردارهای ۵۰بعدی ۵هزار موجودیت را به وسیله‌ی ابزار tnse [42] ,[41] در شکل ۲۱ و در دو بعد نمایش دادیم.

شکل ۲۱ بازنمایی بردار ۵۰ هزار موجودیت که به روش TransE آموزش دیده شده است

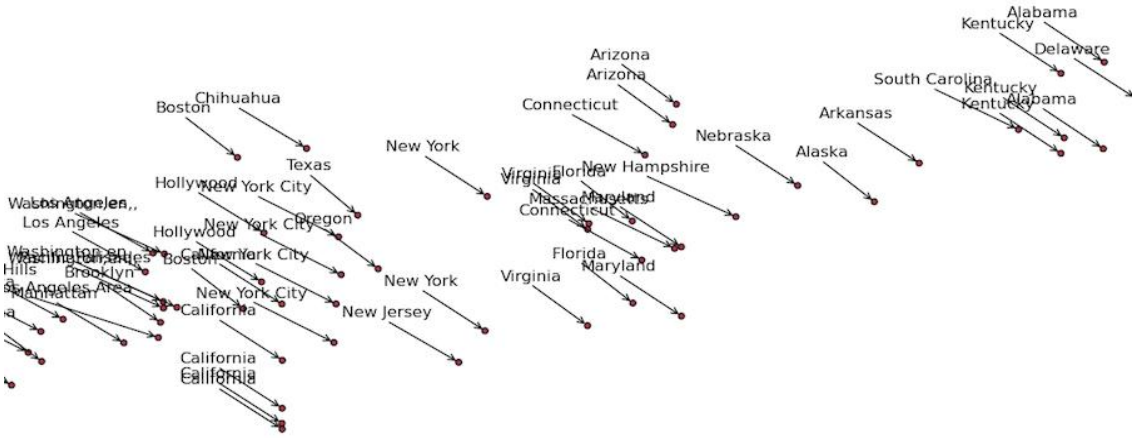
در این شکل مشاهده می‌شود که موجودیت‌ها به صورت جزیره‌های کوچک دور هم جمع شده‌اند و به نظر می‌رسد که تشکیل این جزیره‌ها تصادفی نیست، زیرا در صورت تصادفی بودن انتظار داشتیم موجودیت‌ها در صفحه پخش شوند.

برای مطمئن شدن از این فرض چند جزیره را در شکل ۲۲ و شکل ۲۳ و شکل ۲۴ بزرگ‌نمایی کرده‌ایم.



شکل ۲۲ بازنمایی قسمتی از موجودیت‌ها که در مورد ژانر مجموعه‌های تلوزیونی هستند

در شکل ۲۲ مشاهده می‌شود که عمده‌ی موجودیت‌هایی که در این جزیره قرار گرفته‌اند ژانر فیلم‌های سینمایی هستند، مانند درام[1]، علمی تخیلی[2]، طنز و ...



شکل ۲۳ بازنمایی قسمتی از موجودیت‌ها که در مورد ایالت‌های آمریکا هستند

در شکل ۲۳ مشاهده می‌شود که ایالت‌های آمریکا یک جزیره را تشکیل داده‌اند، مانند آریزونا، کالیفورنیا، تگزاس و ...

[1] Drama

[2] Science fiction

شکل ۲۴ بازنمایی قسمتی از موجودیت‌ها که در مورد نام سریال‌های تلوزیونی آمریکا هستند

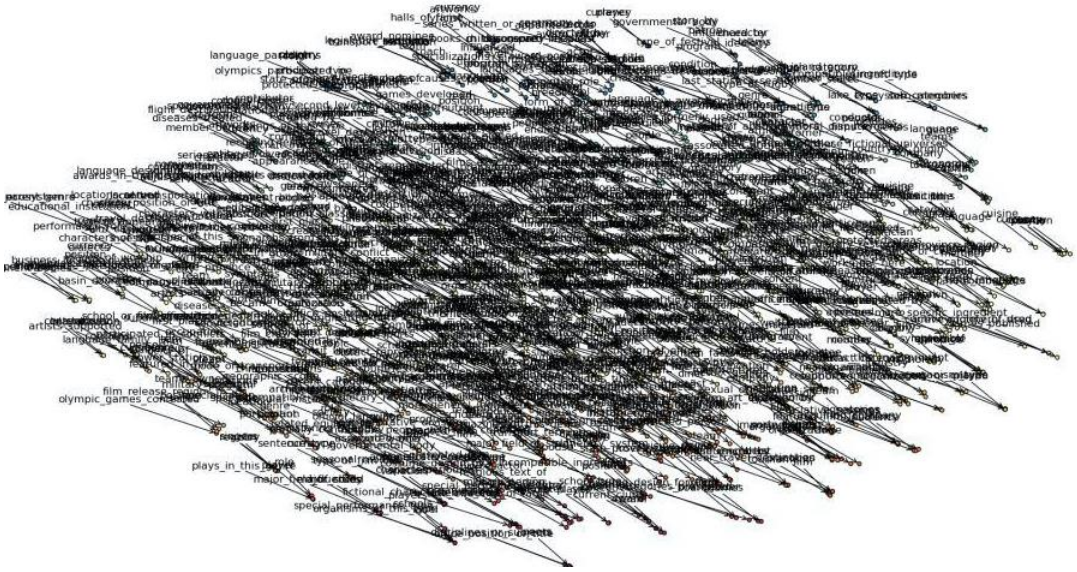
در شکل ۲۴ مشاهده می‌شود که نام سریال‌های تلوزیونی یک جزیره را تشکیل داده‌اند.

در مثال‌هایی که گفته شد، مشخص است که رابطه‌ی بین موجودیت‌ها به خوبی در این روش‌ها درک شده است و یک همبستگی بین بردارهای آموزش دیده شده وجود دارد.

می‌دانیم که رابطه‌ها هم مانند موجودیت‌ها می‌توانند ارتباط‌ها و همبستگی‌های خود را داشته باشند. برای مثال رابطه‌ی «پدر بودن» به رابطه‌ی «فرزندی» نزدیک‌تر است نسبت به رابطه‌ی «نویسنده‌ی کتاب» بودن و انتظار داریم بردارها و ماتریس‌های آموزش دیده شده برای رابطه‌ها این همبستگی‌ها را درک کرده باشد و در فاصله‌ی کمی از یکدیگر قرار گرفته باشند و همان دسته‌بندی و جزیره شدنی که برای موجودیت‌ها اتفاق افتاده بود را این‌جا نیز شاهد باشیم.

حال همین مساله را در بردارهای آموزش دیده‌شده بررسی می‌کنیم. برای بررسی این امر، در شکل ۲۵ بردارهای همه رابطه‌های استفاده شده در Freebase15k را در دو بعد به نمایش در آورده‌ایم. همان‌طور که در شکل مشخص است، برای رابطه‌ها حالت جزیره‌ای شدن که در بازنمایی موجودیت‌ها رخ داده بود،

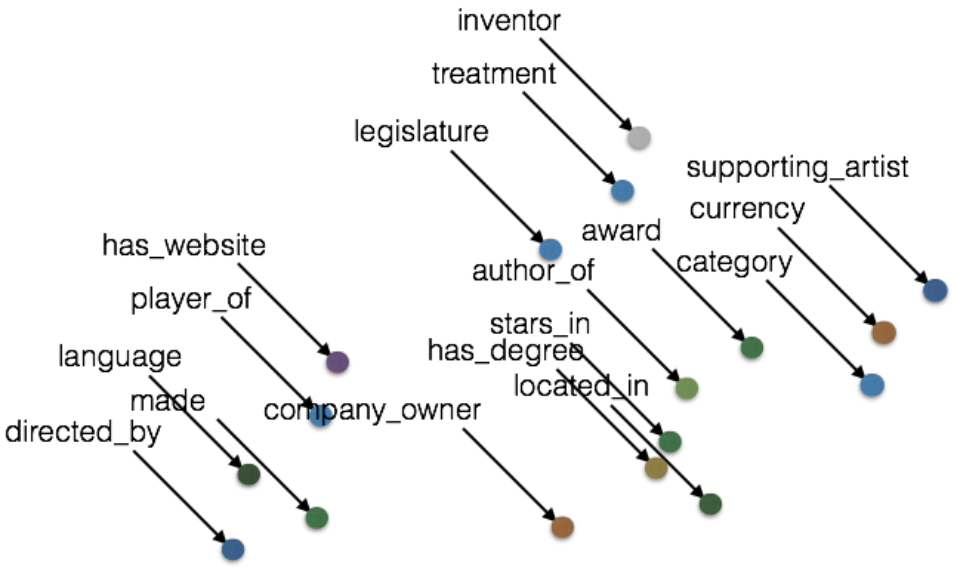
مشاهده نمی‌شود و بردارها روی صفحه پخش[1] است. به نظر می‌رسد که در بردارهای آموزش دیده برای رابطه‌ها این همبستگی و ارتباط وجود ندارد.



شکل ۲۵ بازنمایی بردار روابط پایگاه دانش freebase15k که توسط روش TransE آموزش دیده شده است

برای بررسی بیشتر این موضوع چند قسمت از بازنمایی این بردارها را در شکل ۲۶ بزرگ‌نمایی کرده‌ایم. همان‌طور که پیش‌بینی کرده بودیم رابطه‌هایی که در کنار هم قرار گرفته‌اند هیچ ارتباط و همبستگی‌ای با یکدیگر ندارند و کاملا در صفحه پخش شده‌اند و به نظر می‌رسد که این مدل‌ها نتوانسته‌اند در این قسمت خوب عمل کنند و رابطه‌ها را درک کنند. در فصل بعد روی این مساله تمرکز و سعی می‌کنیم این مساله را در آموزش مدل‌ها دخیل کنیم و همبستگی بین رابطه‌ها را نیز در آموزش مدل‌ها بگنجانیم و انتظار داریم نتیجه‌ی بهتری در معیارهای کیفیت بگیریم.

[1] Scatterd



شکل ۲۶ بزرگنمای قسمتی از بردارهای روابط موجود در پایگاه دانش freebase15k

۳-۴- بررسی عمل‌کرد روش‌ها در یک نگاه

در این قسمت روش‌های موجود را به طور کامل و در کنار هم بررسی خواهیم کرد و قدرت و ضعف هر یک را از نظر مقیاس‌پذیری و نتایج بدست آمده با یکدیگر مقایسه می‌کنیم.

در جدول ۵ تعداد پارامترهایی که هر یک از روش‌های بالا برای آموزش دیدن نیاز دارند، آورده شده است، همان‌طور که مشاهده می‌شود روش‌های رسکال و NTN تعداد پارامتر خیلی زیادی باید آموزش دهند و مشکل مقیاس‌پذیری بر دارند و روی پایگاه‌های دانش بزرگ خیلی کند خواهند بود. در مقابل روش‌های SE و TransE و TransH با ایده‌هایی که برای کاهش تعداد پارامترهای مساله پیاده‌سازی کرداند این تعداد را بسیار کاهش داده و مقیاس پذیری خوبی دارند، به صورتی که مدل TransE روی پایگاه دانش Freebase 1M در مدت تقریبا یک روز با hit@10 برابر با ۳۴٪ به جواب رسیده است در صورتی که روش‌های رسکال و NTN روی این پایگاه دانش به جواب نمی‌رسند، که این موضوع بر عدم مقیاس‌پذیری روش‌های رسکال و NTN تاکید می‌کند.

جدول ۵ تعداد پارامترهای هر روش

Method	#Params	On FB15K
رسکال	$O(n_e d + n_r d^2)$	88M (d=250)
MLP (NTN)	$O(n_e d + n_r d^3)$	165M (d=50)
SE	$O(n_e d + 2n_r d^2)$	8M (d=50)
TransE	$O(n_e d + n_r d)$	0.8M (d=50)

در جدول ۶ میزان Hit@10 و میانگین رتبه‌ی هر یک از روش‌های معرفی شده نمایش داده شده است. در این جدول مشاهده می‌شود که روش NTN در hit@10 نتیجه‌ی مطلوبی نگرفته است، در معرفی این روش اشاره کردیم که این روش به سرعت به بیش‌برازش میل می‌کند و همین امر باعث شده است که پاسخ‌های درستی در مورد مساله نداشته باشد.

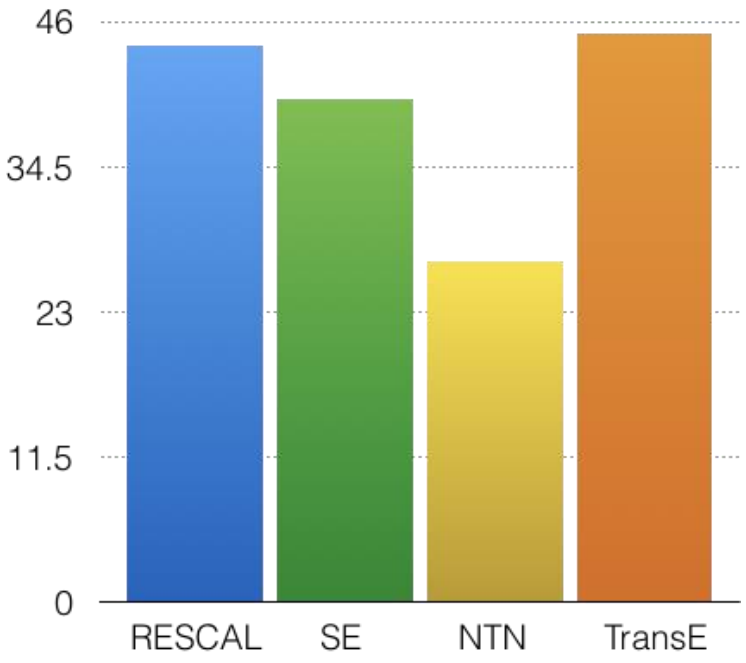
نکته‌ی مهم دیگری که در این جدول دیده می‌شود کارایی خوب روش‌های SE و TransE هست که علاوه بر کاهش پارامتری که داشته‌اند، هم از نظر hit@10 و هم از نظر میانگین رتبه نتایج خوبی داشته‌اند.

در روش رسکال نیز مقدار hit@10 خوب بوده است اما رتبه‌ی میانگین مقدار بسیار بالایی نسبت به روش‌های دیگر داشته که نشان از این دارد که علی‌رغم این‌که بسیاری از روابط را درست فهمیده است و نتایج خوبی از آن‌ها گرفته است، اما بسیاری از روابط را نیز اصلا درک نکرده و پاسخ‌هایی که به

سوالات در مورد این روابط داده است جواب‌های تقریبا تصادفی‌ای داشته است و رتبه‌های دور از انتظاری گرفته‌اند که باعث شده میانگین رتبه‌ی این روش چنین افزایشی داشته باشد.

جدول ۶ مقدار رتبه‌ی میانگین و hit@10 در روش‌های مورد بررسی

	Hit@10	**Mean**
رسکال	42.1%	683
SE	39.8%	162
NTN	27%	164
TransE	45.1%	125



شکل ۲۷ درصد hit@10 در روش‌های مورد بررسی

۴-۴- استفاده از قوانین انجمنی برای بهبود نتایج

در این فصل به طور کامل قوانین انجمنی و چگونگی استخراج آن‌ها را شرح دادیم و پنج قانون هدف نیز مشخص کردیم که به کمک آن‌ها تصمیم داریم مدل‌های موجود مبتنی بر ویژگی پنهان را بهبود دهیم. در ادامه به چگونگی انجام این بهبود می‌پردازیم.

در برخی از روش‌های معرفی شده در بخش ۵-۲- ابتدا همه‌ی این قانون‌ها را روی حقیقت‌های مجموعه‌ی آموزش اعمال و حقایق جدید بدست آمده را دوباره به پایگاه دانش اضافه کردیم و سپس از پایگاه دانش جدید که بزرگ‌تر، کامل‌تر و دقیق‌تر شده است استفاده نمودیم.

اما در روش‌های TransE و TransH این قوانین را مستقیما وارد فرایند آموزش نمودیم، به این صورت که پایگاه دانش را گسترش نداده و با همان حدود ۵۰۰ هزار حقیقت اولیه عملیات آموزش را شروع کردیم اما در حین آموزش از این قوانین استفاده کردیم تا نتایج دقیق‌تری بگیریم که در ادامه به این موضوع خواهیم پرداخت که این قوانین چگونه استفاده شده‌اند.

عدم اعمال این قوانین روی پایگاه دانش باعث شد که هم مساله‌ی پیش‌پردازش برای اعمال قوانین روی پایگاه دانش را نداشته باشیم و هم مساله را با پایگاه دانش کوچک‌تری شروع کنیم که در دو مرحله باعث کاهش زمان محاسبات می‌شود.

در ادامه توضیحات استفاده از این قوانین را روی روش TransE می‌دهیم. در روش TransE هدف کاهش فاصله‌ی بین «بردار موجودیت اول به‌علاوه‌ی بردار رابطه» و «بردار موجودیت دوم» بود که برای حقیقت $r(e_1,e_2)$ این رابطه را به صورت زیر می‌نویسیم:

$$|| e_1 + r - e_2 ||$$

۴-۴-۱- قانون R-subsumption

برای قانون R-subsumption که به صورت $r(x, y) => r'(x, y)$ است در عملیات آموزش هرجا عملیات کمینه‌سازی روی حقیقت $r(x, y)$ انجام گرفت آن را روی روابط هم‌ارز آن که از این قانون بدست می‌آیند نیز انجام می‌دهیم، یعنی روابط $r'(x, y)$. پس در کمینه‌سازی‌ها علاوه بر کمینه کردن رابطه‌ی $\| x + r - y \|$ رابطه‌ی $\| x + r' - y \|$ را نیز کمینه می‌کنیم.

۴-۴-۲- قانون R-equivalence

در قانون R-equivalence که همانند رابطه‌ی قبل است با این تفاوت که قانون برگشت‌پذیر نیز هست، یعنی $r(x, y) <=> r'(x, y)$ می‌توان بصورت بالا عمل کرد و هر قسمت (سمت چپ و راست قانون) از این قانون در حقیقت‌ها دیده شد قسمت دیگر را نیز در معادله‌ی کمینه سازی قرار دهیم. اگر معیار اطمینان این قانون بالا باشد، عملا اتفاقی که می‌افتد این است که رابطه‌های r و r' به یک شکل آموزش دیده می‌شوند و به یک صورت عمل خواهند کرد، که با توجه به هم‌معنا بودن رابطه‌های مثل located_in و contained_by این اتفاق، اتفاق منطقی‌ای خواهد بود و از آن امید بهبود در مدل‌ها را داریم.

۴-۴-۳- قانون تعدی (2-hope translation)

قانون تعدی یا 2-hope translation که به صورت

$$r_1(e_1, e_2) , r_2(e_2, e_3) => r'(e_1, e_3)$$

تعریف می‌شود، نسبت به قانون‌های قبلی پیچیدگی محاسباتی بیشتری خواهد داشت، زیرا پس از پیدا کردن دو حقیقت در سمت بدنه‌ی قانون قادر به اعمال قانون خواهیم بود. پس در هر مرحله از آموزش که به یکی از حقیقت‌های سمت چپ قانون رسیدیم، عملیات جستجو برای حقیقت دیگر را شروع می‌کنیم و در صورت پیدا شدن حقیقت دوم این قانون را اعمال می‌کنیم. (عملیات جستجو برای حقیقت دوم به صورت موازی انجام می‌پذیرد و باعث کاهش سرعت آموزش نخواهد شد.)

پس از یافتن هر دو حقیقتِ سمتِ بدنه‌یِ قانون یعنی $r_1(e_1, e_2)$ و $r_2(e_2, e_3)$ برای اعمال قانون در آموزش باید عملیات کمینه‌سازی را برای حقیقت $r'(e_1, e_3)$ نیز انجام دهیم یعنی عبارت $\| e_1 + r' - e_3 \|$ را نیز کمینه کنیم.

از طرفی روابط را در روش TransE به صورت یک بردار انتقال در نظر گرفتیم پس انتظار داریم که مجموع انتقالی که بردارهای r_1 و r_2 ایجاد می‌کنند برابر باشد با انتقال بردار r' یعنی:

$$r_1 + r_2 = r'$$

پس از عکس این رابطه نیز می‌توانی استفاده کنیم و اگر جایی به قانون r' را مشاهده کردیم در کنار کمینه کردن مقدار

$$\| e_1 + r' - e_3 \|$$

مقدار مجموع دو بردار دیگر را نیز کمینه کنیم:

$$\| e_1 + r_1 + r_2 - e_3 \|$$

۴-۴-۴- **قانون** Triangle alignment

در این قانون نیز مانند قانون قبل می‌توان عمل کرد و با پیدا کردن هر یک از حقیقت‌های موجود در بدنه‌ی قانون، دومین حقیقت را جستجو کرد و در صورت پیدا کردن آن، علاوه بر کمینه سازی که روی خود حقیقت‌های اصلی انجام می‌دهیم کمینه سازی را روی حقیقت بدست‌آمده از قانون نیز اعمال کنیم.

$$r_1(e_1, e_3) , r_2(e_2, e_3) => r'(e_1, e_2)$$

$$|| e_1 + r_1 - e_3 || , || e_2 + r_2 - e_3 || => \quad || e_1 + r' - e_2 ||$$

۴-۴-۵- **قانون** Specefic R-subsumption

در این قانون همچون دو قانون قبل باید دو حقیقت که در سمت بدنه‌ی قانون آمده‌اند را یافته و پس از آن نتیجه‌ی قانون را در عملیات آموزش تاثیر دهیم. این قانون را به این صورت تاثیر می‌دهیم که اگر دو قانون $r_1(x, y)$ و $r_2(x, V)$ در حقایق وجود داشت، علاوه بر کمینه کردن فاصله‌ی بین این دو حقیقت عبارت نتیجه‌ی این قانون را نیز در عملیات کمینه‌سازی شرکت می‌دهیم به صورت

$$|| x + r' - y ||$$

فصل ۵: ارزیابی

۵-۱- مقدمه

در فصل قبل قوانین انجمنی و نحوه‌ی استخراج، و چگونگی استفاده از آن‌ها را توضیح دادیم. در این فصل این قوانین انجمنی را وارد عملیات آموزش کرده و نتایج بدست‌آمده را بررسی و مقایسه می‌کنیم. در بخش ۵-۲- به تشریح آزمایش‌ها و نتایج بدست‌آمده می‌پردازیم و آزمایش‌ها را به تفکیک قوانین هدف که در بخش ۳-۳- مطرح شد توضیح می‌دهیم. بخش ۵-۴- را به بررسی نتایج و مقایسه آن‌ها و جمع بندی این فصل اختصاص می‌دهیم و در فصل بعد نتیجه‌گیری خواهیم داشت و کارهایی که در آینده برای بهبود بیشتر نتایج می‌توان انجام داد را معرفی می‌کنیم.

۵-۲- آزمایش‌ها

در این تحقیق هدف فقط بهبود نتایج روش‌های مبتنی بر ویژگی‌های پنهان نبوده و به دنبال ارائه‌ی چارچوبی هستیم که به کمک آن بتوانیم قدرت و نقاط قوت و ضعف روش‌ها را نیز مشخص کنیم. برای ارائه‌ی این چارچوب از بخش‌بندی‌ای که در بخش ۳-۳- روی قوانین انجمنی داشتیم استفاده می‌کنیم. و بررسی می‌کنیم که هر یک از این دسته قوانین چه مقدار توسط مدل درک شده‌اند و خوب جواب می‌دهند.

برای انجام آزمایش‌ها باید پایگاه دانش و قوانین انجمنی که می‌خواهیم عملیات آموزش را روی آن‌ها انجام دهیم را مشخص کنیم. پایگاه دانش استفاده شده همان پایگاه دانش Freebase15k است که در بخش ۳-۲- معرفی شد (حدود ۶۰۰ هزار حقیقت، ۱۵ هزار موجودیت و ۱۳۰۰ رابطه) است. در ادامه قوانین انجمنی را روی این پایگاه دانش اجرا می‌کنیم و قوانین بدست آمده را دسته‌بندی می‌کنیم.

همان‌طور که در بخش ۴-۳- اشاره شد، پس از اجرای قوانین انجمنی روی پایگاه دانش Freebase15k ۴۱۱۹۶ قانون استخراج شد اما همه‌ی این قانون‌ها از نظر معیار کیفیت و اطمینان، شرایط استفاده در عملیات آموزش را ندارند، نیاز است تا قوانین باکیفیت را مشخص کنیم و فقط از آن‌ها استفاده کنیم. برای مشخص کردن قوانین با کیفیت با آزمون و خطا به معیار اطمینان PCA برابر ۰٫۸ و پوشش سر ۲۰۰ رسیدیم و فقط قوانینی که معیار اطمینان بالای ۰٫۸ و پوشش سر بیش از ۲۰۰ داشتند را انتخاب و در عملیات آموزش شرکت دادیم. پس از اعمال این محدودیت‌ها تعداد قوانین کاندید بدست‌آمده برابر با ۴۱۹۳ قانون شد. این قوانین را بر اساس تقسیم‌بندی‌های بخش ۳-۳- دسته‌بندی کردیم که این دسته‌بندی در جدول ۷ مشخص شده است.

جدول ۷ تعداد قوانین استخراج شده روی freebase15k به تفکیک قوانین هدف

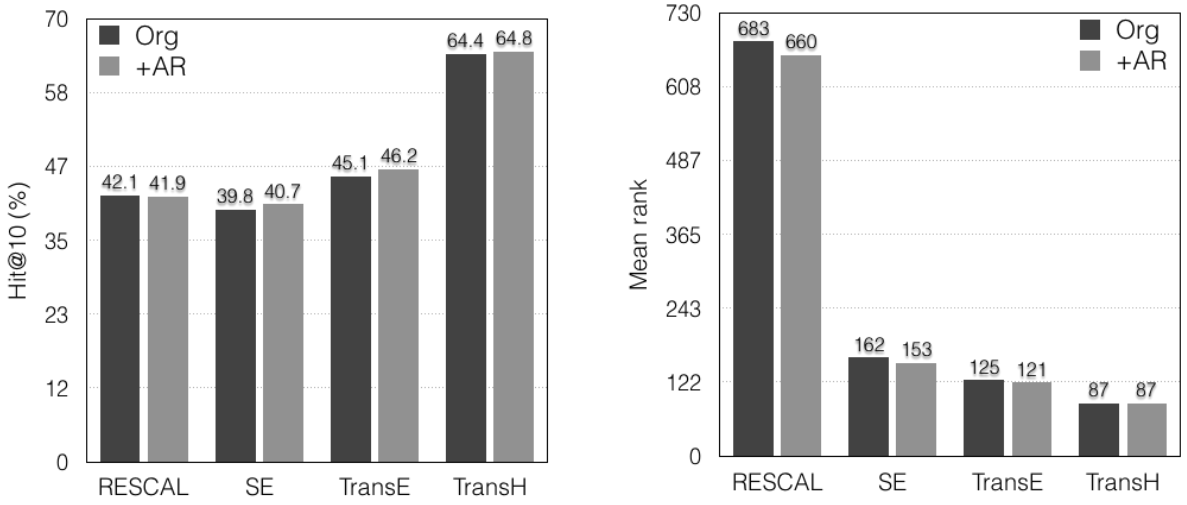
Rules		
name	**Rule**	**#rule**
R-Subsumption	r(x, y) => r'(x, y)	۱۱۲۷
R-Equivalence	r(x, y) <=> r'(x, y)	۷۸۲
2-hope	r1(x, y), r2(y, z) => r'(x, z)	۸۱۰
Triangle alignment	r(x, z), r(y, z) => r'(x, y)	۶۹۵
SR-Subsumption	r_1(x, y), r_2(x, v) => r'(x, y)	۷۷۹

All rules		۴۱۹۳

همان‌طور که قبل‌تر هم اشاره شد تصمیم داریم که علاوه بر بهبود نتایج یک چارچوب برای سنجش کیفیت روش‌ها نیز ارائه دهیم، برای این کار طبق آنچه در بخش ۴-۲- گفته شد این قوانین را به صورت دسته‌ای به روش‌های آموزش اضافه می‌کنیم و نتایج را مشاهده می‌کنیم و در آخر نیز همه‌ی قوانین را باهم به روش‌ها اضافه می‌کنیم و میزان بهبود انجام شده توسط این قانون‌ها را در روش‌های مختلف بررسی می‌کنیم.

در نمودار شکل ۲۸-چپ معیار hit@10 و در نمودار شکل ۲۸-راست

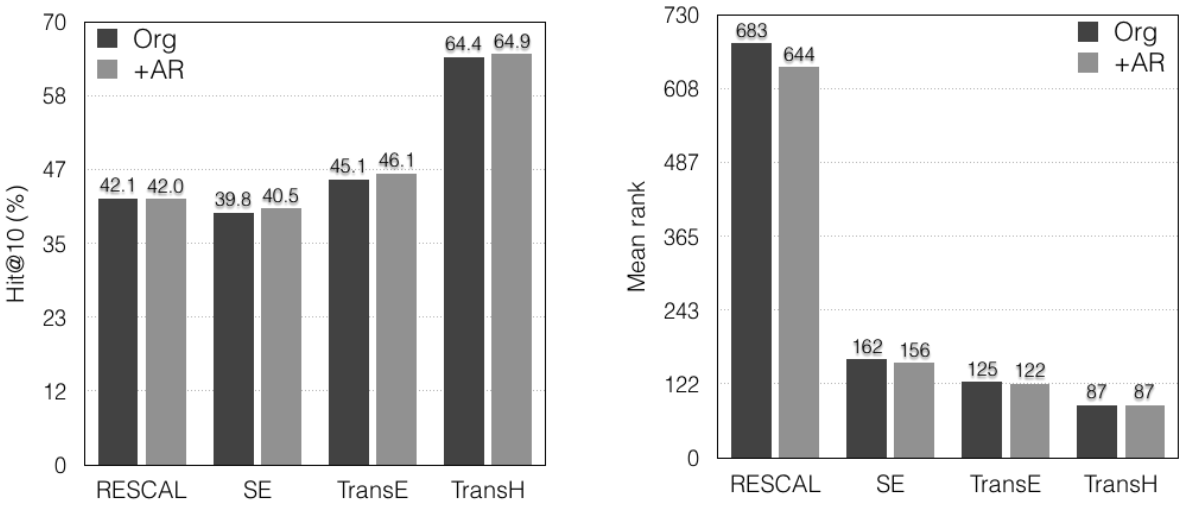
معیار میانگین رتبه برای روش‌های معرفی شده در ۵-۲-، قبل و بعد از اضافه کردن قوانین استخراج شده در دسته‌ی R-Subsumption را مشاهده می‌کنیم.



شکل ۲۸-چپ تاثیر اعمال قانون R-Subsumption بر hit@10 -راست تاثیر اعمال قانون R-Subsumption بر رتبه‌ی میانگین

در نمودارها مشاهده می‌شود که اضافه کردن قوانین R-subsumption در هیچ یک از معیارها تاثیر چندان چشم‌گیری نداشته و بهبودهای جزئی روی نتایج روش‌ها داشته است. این مساله نشانگر این است

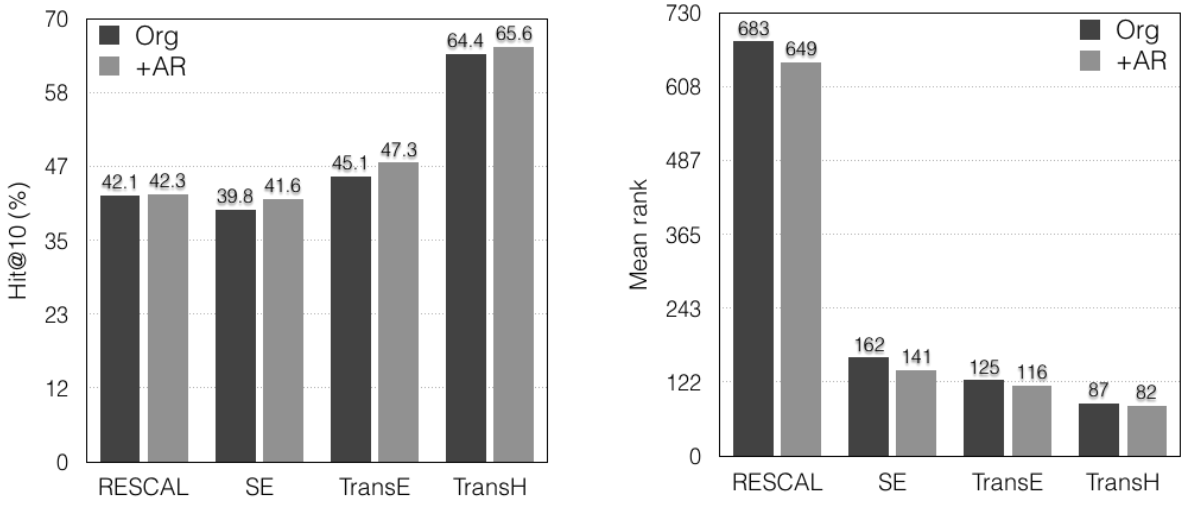
که هر ۴روش معرفی شده در فهمیدن قوانین R-subsumption مشکلی نداشته‌اند و در زمان آموزش این روابط توسط مدل شناسایی و درک شده است.



شکل ۲۹-راست تاثیر اعمال قانون SR-Subsumption بر رتبه‌ی میانگین - چپ تاثیر اعمال قانون SR-Subsumption بر hit@10

در نمودارهای شکل ۲۹-راست و شکل ۲۹-چپ نتایج معیارهای کیفیت روش‌های آموزش را قبل و بعد از اضافه کردن قوانین SR-subsumption مشاهده می‌کنیم. این قانون‌ها نیز مانند قانون‌های R-subsumption تاثیر چندانی در معیارهای hit@10 و میانگین رتبه نداشته است.

می‌توان نتیجه گرفت که مدل‌های مبتنی بر ویژگی‌های پنهان در کل قوانینی که به طور مستقیم از روی یکدیگر نتیجه گرفته می‌شوند را به خوبی درک می‌کنن و مدل آموزش دیده این روابط رو به خوبی تشخصی می‌دهد و در پاسخ به سوالات به اشتباه نمی‌افتد.

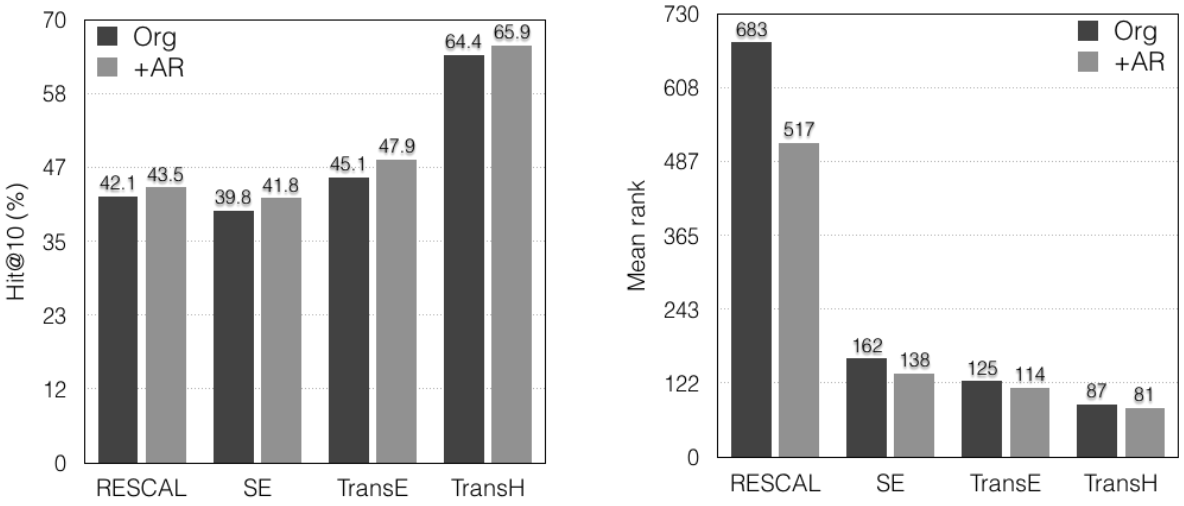


شکل ۳۰-چپ تاثیر اعمال قانون R-equvalence بر hit@10 -راست تاثیر اعمال قانون R-equvalence بر رتبه‌ی میانگین

در نمودارهای شکل ۳۰-چپ و شکل ۳۰-راست مشاهده می‌شود اضافه کردن قوانین R-equvalence تاثیر چندانی روی نتایج رسکال نداشته است و بهبود جزئی داشته است، اما در روش‌های دیگر تاثیر نسبتا خوبی داشته است و باعث بهبود نتایج شده است.

همان‌طور که در بخش ۲-۳-۳- عنوان کردیم این قوانین به کشف روابطی می‌پردازند که به دو صورت بیان شده‌اند ولی معنای یکسانی دارند، مانند روابط located_in و contained_by که هر دو معنی قرار داشتن یک مکان در مکان دیگر را دارد اما به دو صورت بیان شده است. اینجا مشاهده می‌شود که تعداد زیاد این چنین رابطه‌ها در پایگاه دانش و عدم شناسایی آن‌ها توسط مدل‌های آموزش تاثیر منفی‌ای در نتایج داشته است که با اضافه کردن قوانین R-equvalence این مشکل رفع شده و بهبود نسبتا خوبی حاصل شده است.

عدم بهبود در روش رسکال نیز به این دلیل است که این روش همان‌طور که در توضیح آن در بخش ۲-۵-۲- توضیح دادیم همه‌ی جفت ویژگی‌های پنهان از دو موجودیت را با یکدیگر مقایسه و بررسی می‌کند و می‌تواند به مقدار خوبی این روابط یکسان را ببیند و در ماتریس‌های رابطه قرار دهد.



شکل ۳۱-چپ تاثیر اعمال قانون 2-hope translation بر hit@10 -راست تاثیر اعمال قانون 2-hope translation بر رتبه‌ی میانگین

در نمودارهای شکل ۳۱-راست و شکل ۳۱-چپ با اضافه کردن قوانین 2-hope translation نیز در هر دو معیار بهبود محسوسی دیده می‌شود. مشاهده می‌شود که در روش رسکال نیز که با اضافه کردن قانون‌های قبلی تغییر چندانی نکرده بود، با اضافه کردن این قانون بهبود بسیار زیادی داشته‌ایم و مقدار hit@10 به ۴۳,۵ و مقدار میانگین رتبه با ۱۶۶ رتبه کاهش به ۵۱۷ رسیده است که بهبود چشم‌گیری است. هم‌چنین در دیگر مدل‌ها بهبودهای زیادی را داشته‌ایم، در روش‌های SE و TransE و TransH هرکدام به ترتیب ۲ و ۲,۸ و ۱,۵ درصد رشد را داشته ایم. هم‌چنین در این آزمایش میانگین روش TransH از ۸۷ به ۸۱ رسیده است که با توجه به پایین بودن میانگین اولیه و سخت بودن تغییر در آن، بهبود بزرگی محسوب می‌شود.

می‌توانیم نتیجه بگیریم که هیچ یک از این مدل‌ها قادر به شناسایی قوانین غیر مستقیم پیچیده نبوده است. در آزمایش‌های قبلی دیدیم که قوانینی غیر مستقیم که با استنتاج از روی یک رابطه ساخته شده بودند (در قسمت بدنه‌ی قانون فقط یک حقیقت وجود داشت) را مدل‌ها بهتر درک کرده بودن و اضافه کردن قوانین مربوط به این آزمایش‌ها تاثیر چندانی در نتیجه نداشت، اما اضافه کردن قوانین غیرمستقیم پیچیده‌تری مانند قانون 2-hope translation (که در سمت بدنه‌ی قانون از دو حقیقت استفاده شده

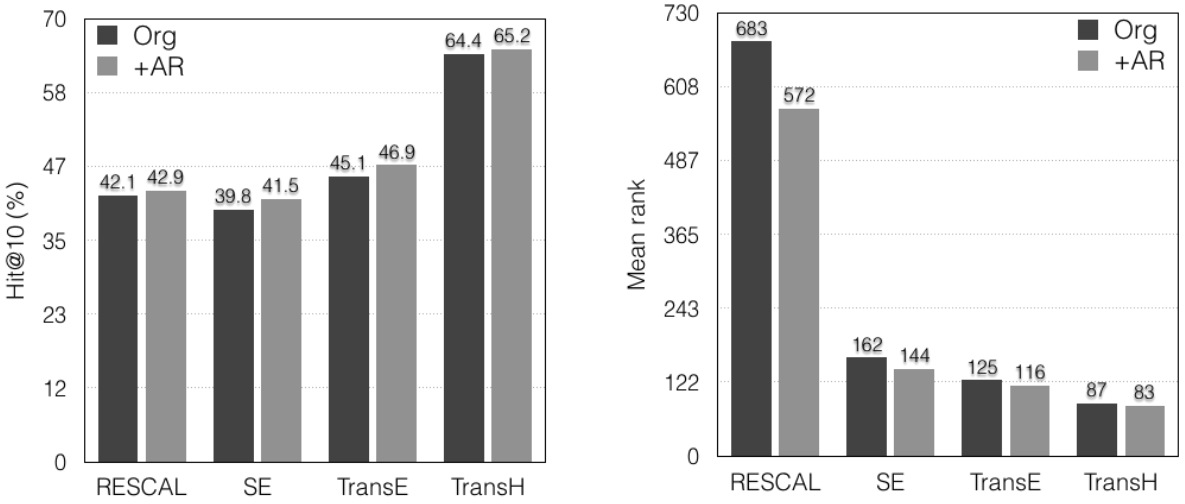
است) بهبودهای زیادی را در همه‌ی مدل‌ها اعمال کرده است. پس به کمک این قانون‌ها توانستیم چند قدم جلوتر از آنچه مدل‌ها قادر به دیدن آن بودند را به آن‌ها نشان دهیم و قوانین پیچیده‌ی موجود در پایگاه‌های دانش را نیز در امر آموزش دخیل کنیم.

بهبود در معیار hit@10 نشان می‌دهد که اضافه کردن این قوانین باعث شده که در جواب‌های دقیق که به سوالات داده شده است بهبود داشته باشیم و تعداد جواب‌هایی صحیح که زیر ۱۰ کاندید اول بودند را افزایش دهیم و از طرفی بهبود در معیار میانگین رتبه نشان می‌دهد که در رابطه‌هایی که مدل برای آن‌ها درست آموزش ندیده است و جواب‌های پرتی برای آن‌ها در نظر گرفته است، اوضاع پاسخ‌ها بهتر شود.

به طور خاص یکی از دلایل بالا بودن معیار میانگین رتبه این است که برای برخی روابط حقیقت‌های کمی در پایگاه دانش موجود است و مدل‌ها از روی این تعداد کم رابطه قادر به شناسایی الگو برای پاسگویی به سوالات در مورد این حقیقت‌ها و روابط نیستند. همین امر باعث می‌شود که پاسخ‌هایی که به سوالات در مورد این حقیقت‌ها می‌دهند به صورت تصادفی باشد و رتبه‌های پرتی بگیرد. این رتبه‌های پرت باعث افزایش زیادی در معیار میانگین رتبه می‌شود، در روش رسکال این مساله را به خوبی مشاهده می‌کنیم که با وجود این‌که قریب به ۴۲٪ جواب‌ها زیر رتبه‌ی ۱۰ قرار می‌گیرند اما میانگین رتبه‌ای که برای این روش اعلام شده است ۶۸۳ است.

این مشکل عدم وجود تعداد نمونه‌ی کافی برای آموزش مدل را تا حدی می‌توان با اضافه کردن نمونه‌های غیر مستقیم موجود در پایگاه دانش حل کرد، به این معنی که برای خیلی از روابط نمونه‌هایی وجود دارد که از روی بقیه‌ی حقایق درون پایگاه دانش می‌توان به آن‌ها پی برد و از آن‌ها استفاده کرد. در اینجا هم ما با اضافه کردن قانون‌های 2-hope translation چنین کاری صورت داده‌ایم و حقیقت‌هایی که از به وسیله‌ی قانون تعدی قابل شناسایی بوده‌اند را به پایگاه دانش اضافه کرده‌ایم و در عملیات

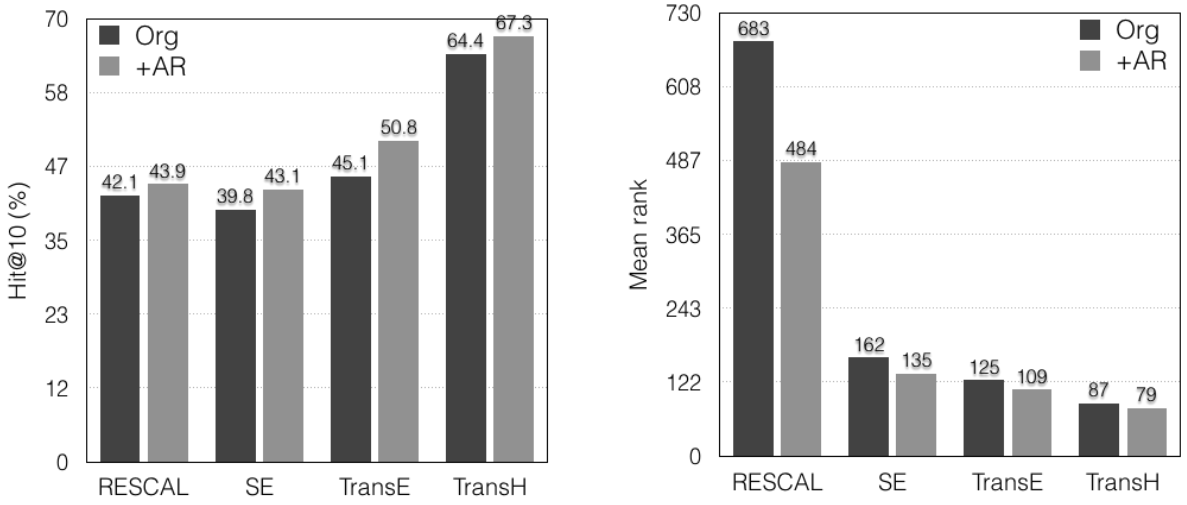
آموزش تاثیر داده‌ایم و نهایتا نتایج خوبی نیز در پاسخ به سوالاتی که از مدل پرسیده می‌شود بدست آمده است.



شکل ۳۲-چپ تاثیر اعمال قانون Triangle alignment بر hit@10 -راست تاثیر اعمال قانون Triangle alignment بر رتبه‌ی میانگین

در نمودارهای شکل ۳۲-راست و شکل ۳۲-چپ نیز تاثیر استفاده از قوانین Triangle alignment را مشاهده می‌کنیم که بهبودهای این نتایج نیز به مانند آزمایش قبل قابل قبول بوده است و همان دلیل پیچیدگی قانون که مدل قادر به درک آن نبوده است در این مورد هم صدق می‌کند و باعث بهبود نتایج شده است.

در ادامه نتایج اعمال کل قانون‌هایی که در قسمت ۳-۳- معرفی کردیم را روی مدل‌های معرفی شده را بررسی می‌کنیم. نتایج بدست آمده در نمودارهای شکل ۳۳-چپ و شکل ۳۳-راست نشان داده شده است.



شکل ۳۳-چپ تاثیر اعمال همه‌ی قانون‌های هدف بر hit@10 - راست تاثیر اعمال همه‌ی قانون‌های هدف بر رتبه‌ی میانگین

مشاهده می‌شود که اعمال همه‌ی قانون‌ها در کنار هم نیز باعث بهبود در نتایج همه‌ی روش‌های معرفی شده، شده است و علاوه بر چارچوبی که برای سنجش نقاط قوت و ضعف روش‌ها معرفی کردیم توانستیم در همه‌ی روش‌ها بهبودهای قابل قبولی نیز اعمال کنیم.

۳-۵- بررسی زمانی اجرای الگوریتم‌ها

زمانی که برای یادگیری مدل صرف می‌شود را برای همه‌ی مدل‌ها به دو قسمت تقسیم می‌کنیم: زمانی که برای استخراج قوانین صرف می‌شود و زمانی که عملیات یادگیری مدل انجام می‌شود.

زمان استخراج قوانین در همه‌ی روش‌ها یکسان بوده و یک بار قبل از عملات آموزش روی پایگاه دانش صورت می‌گیرد و با توجه به موازی بودن الگوریتم استخراج قوانین که در بخش ۲-۳- توضیح داده شد فرایند زمانبری نیست. برای مثال این فرایند برای ا ستخراج قوانین روی پایگاه دانش freebase15k در حد چند دقیقه است.

زمانی که صرف یادگیری مدل‌ها می‌شود بسته به مدل و روش یادگیری متفاوت است. این زمان را نیز در دو د سته برر سی می‌کنیم: روش‌هایی که قوانین انجمنی را به طور غیر م ستقیم روی پایگاه دانش

اعمـال کردیم و روش‌هـایی کـه در آن قوانین انجمنی را واد فراینـد آموزش کردیم (TransE و TransH).

در روش‌هایی که قوانین انجمنی را وارد پروسه‌ی آموزش نکردیم و صرفا آن‌ها روی پایگاه دانش اعمال کردیم و پایگاه دانش را کامل‌تر کردیم وپس از آن فرایند آموزش را روی پایگاه دانش کامل‌تر انجام دادیم زمان اجرای الگوریتم‌ها تغییر خطی‌ای نسبت به میزان بزرگ شدن پایگاه دانش داشته است، مثلا اگر پایگاه دانش پس از اعمال قوانین انجمنی روی آن ۱,۲ برابر شود، زمان اجرا و آموزش هم ۱,۲ برابر خواهد شـــد و تاثیر نمایی روی زمان اجرا نخواهد داشـــت و با توجه به این‌که ما فقط قوانین با کیفیت که معیارهای اطمینان کیفیت آن‌ها را تضمین می‌کند استفاده می‌کنیم و تعداد آن‌ها نسبت به اندازه پایگاه دانش خیلی زیاد نیست این افزایش زمان مقدار چندان چشم‌گیری نیست.

اما در روش‌هایی که قوانین انجمنی را وارد فرایند آموزش می‌کنیم مثل روش TransE که در بخش ۴-۴- نحوه‌ی این فرایند توضـــیح داده شـــد، زمان آموزش بصـــورت خطی تغییر نمی‌کند زیرا ورودی بزرگ‌تری به الگوریتم نداده‌ایم و تغییراتی درون خود الگوریتم ایجاد کرده‌ایم. ابتدا عملیات آموزش بصورت غیر موازی انجام می‌شد و به این صورت بود که الگوریتم به هر حقیقت میرسید جدول قوانین انجمنی انتخاب شده را بررسی می‌کرد و حقیقت‌هایی که در یک یا چند قانون انجمنی حضور داشتند باعث توقف الگوریتم می‌شـــدند تا قوانین روی پایگاه دانش اجرا شـــود و نمونه‌های مثبت را یافته و در فرایند آموزش شـــرکت دهند. این امر باعث می‌شـــد که زمان اجرای الگوریتم تا چند برابر افزایش یابد، مثلا برای اجرای الگوریتم TransE که بدون قوانین انجمنی روی پایگاه دانش freebase15k حدود ۳,۵ساعت طول می‌کشید روی الگوریتم جدید که از قوانین انجمنی استفاده می‌کرد قریب به ۱۱ساعت زمان می‌برد.

برای حل مشـــکل زمان این الگوریتم‌ها از یک فرایند پیش‌پردازش برای یافتن قوانین انجمنی مرتبط با حقیقت‌ها استفاده کردیم. این پیش‌پردازش به این صورت هست که ریسمان‌هایی به صورت موازی شـــروع به یافتن حقیقت‌هایی که قانون انجمنی معادل برای آن‌ها هســـت می‌کنند و قوانین انجمنی مربوط به این حقیقت‌ها را از پایگاه دانش اســـتخراج می‌کنند و وقتی اجرای الگوریتم به این حقیقت‌ها رســـید دیگر نیاز به اســـتخراج قوانین از پایگاه دانش نداشـــته و قوانین را در دســـت دارد. با اجرای این الگوریتم تقریبا موازی روی ســـیســـتمی با ۴۰هســـته پردازش‌گر زمان اجرا از ۱۱ســـاعت به ۵ســـاعت و ۱۷دقیقه رسید که زمان بسیار خوبی برای چنین تسکی هست.

همان‌طور که توضیح داده شد مشاهده می‌شود که این الگوریتم‌ها با اضافه کردن قوانین انجمنی از نظر مرتبه‌ی زمانی وضعیت نامطلوبی پیدا نکرده و با سربار زمانی کمی به جواب خواهند رسید.

۴-۵- جمع‌بندی

در جدول ۸ و جدول ۹ جمع‌بندی‌ای داریم روی نتایج اعمال قوانین روی روش‌های معرفی شده که در بخش قبل به تفصیل توضیح داده شد و این نتایج را به تفکیک معیار hit@10 و میانگین رتبه در کنار هم قرار داده‌ایم.

جدول ۸ نتایج اعمال قوانین انجمنی هدف بر روی معیار hit@10 روش‌های مورد بررسی

Hit@10 on Freebase 15K (%)							
Method	orginal	R-Subsumption	R-Equivalence	2-hope	Triangle alignment	SR-Subsumption	all rules (~improve)
رسکال	42.1	41.9	42.3	43.5	42.9	42.0	**43.9 (1.8%)**
NTN	27	-	-	-	-	-	**-**
SE	39.8	40.7	41.6	41.8	41.5	40.5	**43.1 (3.3%)**
TransE	45.1	46.2	47.3	47.9	46.9	46.1	**50.8 (5.7%)**
TransH	64.4	64.8	65.6	65.9	65.2	64.9	**67.3 (2.9%)**

جدول ۹ نتایج اعمال قوانین انجمنی هدف بر روی معیار رتبه‌ی میانگین روش‌های مورد بررسی

Mean rank on Freebase 15K							
Method	orginal	R-Subsumption	R-Equivalence	2-hope	Triangle alignment	SR-Subsumption	all rules (~improve)

رسکال	683	660	649	517	572	644	**484 (30%)**
NTN	164	-	-	-	-	-	**-**
SE	162	153	141	138	144	156	**135 (17%)**
TransE	125	121	116	114	116	122	**109 (13%)**
TransH	87	87	82	81	83	87	**79 (10%)**

نکاتی که در این جداول قابل توجه هستند:

- اعمال همه‌ی قوانین روی هر چهار روش مورد آزمایش باعث بهبود قابل قبول نتایج گشته است.
- قوانین پیچیده‌تر مانند قانون 2-hope و قانون Triangle alignment که به طور مستقیم از روی داده‌های موجود در پایگاه دانش قابل برداشت نیستند توسط مدل‌ها به خوبی درک نشده‌اند و از نقاط ضعف روش‌ها به شمار می‌روند که در جدول مشاهده می‌کنیم اضافه کردن این قوانین بهبود خوبی در نتایج داشته است.
- قانون R-equivalence مانند دو قانون بند قبل قانون پیچیده‌ای نیست، اما به حل یکی از مشکلات اساسی پایگاه‌های دانش پرداخته است که وجود روابط مختلف با تعاریف یکسان است. این خاصیت این قانون که روابط یکسان را تشخیص و در عملیات آموزش تاثیر می‌دهد نیز باعث تاثیر خوبی در نتایج همه‌ی روش‌ها بجز روش رسکال شده است که در بخش ۱-۱-۴- دلیل عدم بهبود برای روش رسکال را توضیح دادیم.

- قوانین ساده‌تر مانند R-subsumption و SR-subsumption تقریبا توسط همه‌ی مدل‌ها به خوبی درک شده است و اضافه کردن این قانون‌ها بهبود چندانی را در نتایج حاصل نکرده است.
- به طور خاص در روش رسکال بهبود در معیار hit@10 نسبت به دیگر روش‌ها کمتر بوده است (حدود ۱٫۸٪) اما بهبود در معیار میانگین رتبه مقدار زیادی بوده است و این معیار را از ۶۸۳ به ۴۸۴ (حدود ۳۰٪) کاهش داده است. این مساله نشان می‌دهد که روش رسکال برای روابطی که به میزان کافی نمونه برای آموزش دیدن داشته است خوب عمل کرده و جواب حدود ۴۰٪ از سوال‌ها را در رتبه‌ی زیر ۱۰ پاسخ داده است، اما برای مواردی که به میزان کافی نمونه برای آموزش نداشته است خیلی بد عمل کرده و باعث شده است که میانگین رتبه به مقدار زیادی بالا برود. اضافه کردن قوانین انجمنی به این روش نیز تاثیر چندانی در hit@10 نداشته است و برای روابطی که نمونه‌ی کافی از آن‌ها موجود بوده است مفید واقع نشده است، اما کاهش بسیار زیاد میانگین رتبه‌ی پاسخ‌ها نشان می‌دهد که کمک بسیاری به روابطی که نمونه‌ی کافی نداشته‌اند شده است و جواب‌های بیشتری به سمت منطقی شدن پیش رفته است.
- متاسفانه پیاده‌سازی روش NTN کامل نشده و نتایج اعمال این قوانین روی این روش را برای مقایسه در اختیار نداریم.

در این تحقیق ابتدا روش‌های موجود در زمینه‌ی پیش‌بینی پیوند که از ویژگی‌های پنهان استفاده می‌کردند معرفی شد و سپس با معرفی و استفاده از قوانین انجمنی سعی شد تا در این روش‌ها بهبودهایی ایجاد شود. با توجه به نتایجی که مشاهده کردیم و نکاتی که گفته شد در کل دست‌آوردهای این تحقیق را می‌توان به صورت زیر خلاصه کرد:

- استفاده از قوانین انجمنی در بهبود روش‌های مبتنی ویژگی‌های پنهان

- ارایه‌ی چارچوبی برای مقایسه و بررسی عملکرد و نقاط ضعف و قوت روش‌های موجود مبتنی بر ویژگی‌های پنهان که قابل گسترش به روش‌های دیگر در زمینه‌ی پیش‌بینی پیوند نیز هست.

فصل ۶: نتیجه‌گیری

۱-۶- نتیجه‌گیری

در این پژوهش قصد داشتیم که روی نتایج پیش‌بینی پیوند در مدل‌های مبتنی بر ویژگی‌های پنهان به کمک قوانین انجمنی بهبودی داشته باشیم، از این رو به توضیح و تبیین این روش‌ها پرداخته و نقاط قوت و ضعف آن‌ها را بررسی کردیم و اقدام به رفع برخی ضعف‌های موجود در این روش‌ها نمودیم. از طرفی نیز این روش‌ها برای کار با پایگاه‌های دانش بسیار بزرگ طراحی شده‌اند و نباید محا سبات این روش‌ها پیچیده و زمانبر با شند، که در بخش ۳-۴- تو ضیح داده شد که پیچیدگی هر یک از روش‌ها به چه مقدار هست و هر کدام تا چه حد می‌توانند پاسخ‌گوی نیازهای پایگاه دانش باشند و در روش پی شنهادی این پژوهش نیز سربار زیادی روی محا سبات هیچ یک از روش‌ها ا ضافه ن شد و همه‌ی روش‌ها در زمان معقولی به پاسخ می‌رسند.

همان‌طور که در بخش ۱-۴- گفته شد، روش‌های مبتنی بر ویژگی‌های پنهان به خوبی روابط بین موجودیت‌ها را تشخیص داده و از آن برای پیشبینی پیوند بین موجودیت‌های مختلف استفاده می‌کنند. در بخش ۲-۴- نشان داده شد که این روابط بین پیوندها به خوبی تشخیص داده نمی‌شود و بین بردار پیوندهای آموزش دیده شده هیچ‌گونه ارتباطی موجود نیست. در این پژوهش به کمک قوانین انجمنی و پنج قانون هدفی که انتخاب کردیم توانستیم که این ارتباط بین پیوندها را تا حدی برای مدل م شخص کنیم و به مدل کمک کنیم از روی داده‌های موجود بتواند برای داده‌های بی شتری استنتاج کند.

پس از اعمال این تغییر روی مدل‌های معرفی شده در بخش ۵-۲- و انجام آزمایش‌ها در بخش ۲-۵- م شاهده شد که در همه‌ی مدل‌ها بهبودهایی حا صل شده ا ست که ن شان می‌دهد ارتباط بین پیوندها نیز ارتباط معناداری است و مشخص کردن آن به دستیابی به نتایج بهتر کمک می‌کند.

هم‌چنین به کمک دسته‌بندی‌ای که در بخش ۳-۳- روی قوانین انجمنی استخراج شده داشتیم و اعمال بخش بخش این قوانین روی روش‌های موجود توانستیم چارچوبی ارائه دهیم که توسط آن بتوان مدل‌های فعلی و مدل‌هایی که در آینده معرفی می‌شوند را بررسی کرده و نقاط ضعف و قوت این روش‌ها را مشخص کنیم.

هم‌چنین استفاده از قوانین انجمنی این امکان را به ما داده است که حقیقت‌های نهفته در پایگاه‌های دانش که به صورت صریح وجود ندارند را تشخیص داده و به پایگاه دانش اضافه کنیم، که این کار باعث کامل‌تر شدن پایگاه دانش و در نتیجه بهبود در نتایج شده است.

۲-۶- کارهای آینده

با روش‌هایی که معرفی کردیم توانستیم علاوه بر چارچوبی که برای سنجش کیفیت روش‌های موجود و یافتن نقاط و قوت و ضعف این روش‌ها معرفی کردیم، بهبودهایی در همه‌ی روش‌های معرفی شده داشته باشیم. اما با وجود بهبودهایی که داشتیم بهترین نتیجه‌ای که بهترین روش به ما داده است مقدار hit@10 برابر با ۶۷٫۳٪ است که برای استفاده‌های واقعی از این روش‌ها مقدار خوب و قابل قبولی نیست. این عدد به این معناست که اگر ما سوالی از این مدل بپرسیم به احتمال ۳۳٪ جواب اشتباه می‌دهد و اگر جواب درست بدهد نیز باید این جواب درست را از بین ۱۰ جواب کاندید انتخاب کنیم. این‌که این ۱۰ جواب کاندید با یکدیگر در ارتباط هستند این نوید را می‌دهد که از این روش‌ها در سیستم‌های توصیه‌گر که لیستی از موارد را به کاربر توصیه می‌کند به خوبی قابل استفاده هستند، اما اگر بخواهیم از این مدل‌ها برای پاسخ به سوالات استفاده کنیم بجای hit@10 باید مقدار hit@1 را در نظر بگیریم، زیرا فقط یک جواب صحیح مدنظر است و دیگر جواب‌ها فاقد اعتبارند و جواب‌های درست به شدت کاهش پیدا می‌کند.

با این که این روش‌ها در حال حاضر در آخرین تکنولوژی‌های شرکت‌های بزرگی مانند گوگل و IBM در حال استفاده هستند، اما این روش‌ها هنوز باید بسیار دقیق‌تر شوند تا در کاربردهایی مثل پاسخگویی به سوالات نیز قابل استفاده باشند.

بهبودهایی که می‌توان روی این مساله داد می‌توانند از طریق راهکارهای زیر باشند:

- استفاده از روشی که در این تحقیق ارائه شده و یافتن نقاط ضعف روش‌ها و انتخاب راه‌حل برای حل این ضعف‌ها
- در این تحقیق فقط قوانین انجمنی با پیچیدگی ۱ و ۲ (قوانینی که در قسمت بدنه‌ی خود دو حقیقت را بررسی می‌کردند) در نظر گرفته شدند، مطمئنا درک قوانین با پیچیدگی‌های بیشتر نیز برای مدل‌های موجود سخت بوده و ممکن است نقاط ضعف این روش‌ها باشند، پس اضافه کردن قوانین با پیچیدگی بیشتر از ۲ می‌تواند بهبودهایی را حاصل کند.
- راهکار دیگری برای پایین آوردن معیار میانگین رتبه، فیلتر کردن نتایج روش‌ها است. می‌دانیم پاسخ‌هایی که روش‌های موجود به سوالات می‌دهند به صورت لیستی از جواب‌های مرتب شده است که ممکن است جواب درست سوال مورد ۵۰ام باشد. با بررسی جواب‌ها مشخص می‌شود که خیلی از پاسخ‌هایی که در ۴۹ پاسخ غلط وجود داشته کلا بی‌ارتباط با سوال است. مثلا سوال شده که «نویسنده‌ی کتاب x چه شخصی است؟» و انتظار داریم جواب‌های دریافتی نام اشخاص باشد، اما پاسخ‌هایی از جنس‌های دیگر مانند نام کشورها، وضعیت‌های آب و هوا و ... در پاسخ‌های غلط پیش از پاسخ درست وجود دارد. برای حل این مشکل و فیلتر کردن پاسخ‌های بی ربط می‌توان از روش‌های کشف جامعه[1] استفاده کرد و موجودیت‌ها را دسته‌بندی کرد، و به مدل‌ها آموزش داد که جواب هر سوال باید از جنس چه مجموعه موجودیتی باشد و پس از

[1] community detection

دریافت پاسخ‌ها، پاسخ‌هایی که در این مجموعه موجودیت نیستند را حذف کرد و سریع‌تر به جواب صحیح رسید.

فصل ۷: مراجع

[1] F. M. Suchanek, G. Kasneci, and G. Weikum, "YAGO: A Core of Semantic Knowledge Unifying WordNet and Wikipedia."

[2] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase," in *Proceedings of the 2008 ACM SIGMOD international conference on Management of data - SIGMOD '08*, 2008, p. 1247.

[3] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," Springer, Berlin, Heidelberg, 2007, pp. 722–735.

[4] A. Singhal, "Introducing the Knowledge Graph: things, not strings," 2012. [Online]. Available: https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html. [Accessed: 08-Aug-2017].

[5] G. A. Miller and G. A., "WordNet: a lexical database for English," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, Nov. 1995.

[6] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, "A Review of Relational Machine Learning for Knowledge Graph," *Proc. IEEE*, no. 28, pp. 1–23, 2015.

[7] P. Orbanz and D. M. Roy, "Bayesian Models of Graphs, Arrays and Other Exchangeable Random Structures."

[8] W. Cukierski, B. Hamner, and B. Yang, "Graph-based Features for Supervised Link Prediction."

[9] M. Richardson and P. Domingos, "Markov logic networks," *Mach. Learn.*, vol. 62, no. 1–2, pp. 107–136, Feb. 2006.

[10] A. Garcia-Duran, A. Bordes, N. Usunier, and Y. Grandvalet, "Combining Two And Three-Way Embeddings Models for Link Prediction in Knowledge Bases," *Arxiv*, vol. 1, p. 26, 2015.

[11] G. Angeli and C. D. Manning, "Philosophers are Mortal: Inferring the Truth of Unseen Facts," pp. 133–142.

[12] R. West, E. Gabrilovich, K. Murphy, S. Sun, R. Gupta, and D. Lin, "Knowledge

base completion via search-based question answering," in *Proceedings of the 23rd international conference on World wide web - WWW '14*, 2014, pp. 515–526.

[13] L. Bottou, "Large-Scale Machine Learning with Stochastic Gradient Descent," in *Proceedings of COMPSTAT'2010*, Heidelberg: Physica-Verlag HD, 2010, pp. 177–186.

[14] A. Bordes, N. Usunier, J. Weston, and O. Yakhnenko, "Translating Embeddings for Modeling Multi-Relational Data," *Adv. NIPS*, vol. 26, pp. 2787–2795, 2013.

[15] M. Nickel, V. Tresp VOLKERTRESP, and H.-P. Kriegel KRIEGEL, "A Three-Way Model for Collective Learning on Multi-Relational Data."

[16] M. Nickel, "Tensor Factorization for Relational Learning," 2013.

[17] M. Nickel, V. Tresp, and H.-P. Kriegel, "Factorizing YAGO," in *Proceedings of the 21st international conference on World Wide Web - WWW '12*, 2012, p. 271.

[18] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," *Computer (Long. Beach. Calif).*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[19] T. G. Kolda and B. W. Bader, "Tensor Decompositions and Applications," *SIAM Rev.*, vol. 51, no. 3, pp. 455–500, Aug. 2009.

[20] M. Nickel and V. Tresp, "Tensor Factorization for Multi-Relational Learning."

[21] S. Kok and P. Domingos, "Statistical predicate invention," in *Proceedings of the 24th international conference on Machine learning - ICML '07*, 2007, pp. 433–440.

[22] Z. Xu, V. Tresp, K. Yu, and H.-P. Kriegel, "Infinite Hidden Relational Models," Jun. 2012.

[23] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada, and N. Ueda, "Learning Systems of Concepts with an Infinite Relational Model."

[24] I. Sutskever, R. Salakhutdinov, and J. B. Tenenbaum, "Modelling Relational Data using Bayesian Clustered Tensor Factorization."

[25] D. Krompaß, M. Nickel, and V. Tresp, "Large-Scale Factorization of Type-Constrained Multi-Relational Data."

[26] T. G. Kolda, B. W. Bader, and J. P. Kenny, "Higher-Order Web Link Analysis Using Multilinear Algebra," 2005.

[27] G. S. Halford, W. H. Wilson, and S. Phillips, "Processing capacity defined by relational complexity: implications for comparative, developmental, and cognitive psychology.," *Behav. Brain Sci.*, vol. 21, no. 6, pp. 803-31-64, Dec. 1998.

[28] P. Smolensky, “Tensor Product Variable Binding and the Representation of Symbolic Structures in Connectionist Systems.”

[29] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space,” Jan. 2013.

[30] X. Dong *et al.*, “Knowledge vault: a web-scale approach to probabilistic knowledge fusion,” in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD ’14*, 2014, pp. 601–610.

[31] R. Socher, D. Chen, C. D. Manning, and A. Y. Ng, “Reasoning With Neural Tensor Networks for Knowledge Base Completion,” *Proc. Adv. Neural Inf. Process. Syst. 26 (NIPS 2013)*, pp. 1–10, 2013.

[32] B. Yang, W. Yih, X. He, J. Gao, and L. Deng, “Embedding Entities and Relations for Learning and Inference in Knowledge Bases,” Dec. 2014.

[33] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, “A review of relational machine learning for knowledge graphs,” *Proc. IEEE*, vol. 104, no. 1, pp. 11–33, 2016.

[34] P. D. Hoff, A. E. Raftery, and M. S. Handcock, “Latent Space Approaches to Social Network Analysis.”

[35] A. Bordes and J. Weston, “Learning Structured Embeddings of Knowledge Bases,” *Artif. Intell.*, no. Bengio, pp. 301–306, 2009.

[36] Z. Wang, J. Zhang, J. Feng, and Z. Chen, “Knowledge Graph Embedding by Translating on Hyperplanes - TransH,” *AAAI Conf. Artif. Intell.*, pp. 1112–1119, 2014.

[37] L. Galárraga and C. Teflioudi, “Amie: association rule mining under incomplete evidence in ontological knowledge bases,” *Proc. 22nd ...*, pp. 413–422, 2013.

[38] L. Galárraga *et al.*, “Fast Rule Mining in Ontological Knowledge Bases with AMIE+.”

[39] L. A. Galárraga, N. Preda, and F. M. Suchanek, “Mining rules to align knowledge bases,” in *Proceedings of the 2013 workshop on Automated knowledge base construction - AKBC ’13*, 2013, pp. 43–48.

[40] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, “Learning Entity and Relation Embeddings for Knowledge Graph Completion - TransR,” *Proc. Twenty-Ninth AAAI Conf. Artif. Intell. Learn.*, pp. 2181–2187, 2015.

[41] L. J. P. Van Der Maaten and G. E. Hinton, “Visualizing high-dimensional data using t-sne,” *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

[42] L. Van Der Maaten, “Accelerating t-SNE using Tree-Based Algorithms,” *J. Mach. Learn. Res.*, vol. 15, pp. 1–21, 2014.

فصل ۸: واژه‌نامه انگلیسی به فارسی

قانون انجمنی	Association Rule
نمونه‌برداری دسته‌ای	Batch Sampling
دوخطی	Bilinear
لایه‌ی مخفی دوخطی	Bilinear Hidden Layer
فرض جهان بسته	Close World Assumption
فاصله‌ی پنهان	Distance Models
بروزرسانی‌های بسته‌ی کارآمد	Efficient Close-Form Update
فاصه‌ی اقلیدسی	Euclidean Distance
شبکه‌های عصبی پیش‌خور	Feedforward Neural Networks
سیستم رسمی منطق	Formal System Of Logic
وابستگی جهانی	Global Dependency
کمینه‌سازی برمبنای گرادیان	Gradient-Based Minimization
پوشش سر	Head Coverage

Heterogeneous	ناهمگن
Hidden Layer	لایه‌ی مخفی
Hit@10	حدس در ۱۰
Homogeneous	همگن
Knowledge Base	پایگاه دانش
Knowledge Representation	بازنمایی پایگاه دانش
Large-Scale Clustering	خوشه‌بندی با مقیاس بالا
Latent Factor	ویژگی پنهان
Linear	خطی
Local Optimom	بهینه‌ی محلی
Machine Learning	یادگیری ماشین
Mean Rank	رتبه‌ی میانگین
Multi-Relational	چند رابطه‌ای
Nearst Neighbors	نزدیک‌ترین همسایه
Negative Example	نمونه‌ی منفی
Neural Tensor Networks	شبکه‌های عصبی تنسور
Open World Assumption	فرض جهان باز
Overfit	بیش‌برازش
Partial Completeness Assumption	فرض نیمه‌کامل
Positive Example	نمونه‌ی مثبت
Relational	رابطه‌ای

Relational Domain	فضای رابطه‌ای
Relational Latent Factor	آموزش رابطه‌های
Relational Learning	یادگیری رابطه‌ای
Resulotion	بازنمایی
Semantic	معنایی
Semantically Similar	قرابت معنایی
Sequence	توالی
Shared Representations	بازنمایی مشترک
Social Network	شبکه‌ی اجتماعی
Standard Confidence	اطمینان استاندارد
Stochastic Gradient Descent	گرادیان نزولی تصادفی
Support Rule	قانون پشتیبان
Tensor Factorization	عامل‌بندی تنسور
Uni-Relational	تک رابطه‌ای
Unknown	ناشناخته

فصل ۹: واژه‌نامه فارسی به انگلیسی

آموزش رابطه‌ای	Relational Latent Factor
اطمینان استاندارد	Standard Confidence
بازنمایی	Resulotion
بازنمایی پایگاه دانش	Knowledge Representation
بازنمایی مشترک	Shared Representations
بروزرسانی‌های بسته‌ی کارآمد	Efficient Close-Form Update
بهینه‌ی محلی	Local Optimom
بیش‌برازش	Overfit
پایگاه دانش	Knowledge Base
پوشش سر	Head Coverage
تک رابطه‌ای	Uni-Relational
توالی	Sequence
چند رابطه‌ای	Multi-Relational
حدس در ۱۰	Hit@10
خطی	Linear
خوشه‌بندی با مقیاس بالا	Large-Scale Clustering
دوخطی	Bilinear

رابطه‌ای	Relational
رتبه‌ی میانگین	Mean Rank
سیستم رسمی منطق	Formal System Of Logic
شبکه‌های عصبی پیش‌خور	Feedforward Neural Networks
شبکه‌های عصبی تنسور	Neural Tensor Networks
شبکه‌ی اجتماعی	Social Network
عامل‌بندی تنسور	Tensor Factorization
فاصله‌ی پنهان	Distance Models
فاصه‌ی اقلیدسی	Euclidean Distance
فرض جهان باز	Open World Assumption
فرض جهان بسته	Close World Assumption
فرض نیمه‌کامل	Partial Completeness Assumption
فضای رابطه‌ای	Relational Domain
قانون انجمنی	Association Rule
قانون پشتیبان	Support Rule
قرابت معنایی	Semantically Similar
کمینه‌سازی برمبنای گرادیان	Gradient-Based Minimization
گرادیان نزولی تصادفی	Stochastic Gradient Descent
لایه‌ی مخفی	Hidden Layer
لایه‌ی مخفی دوخطی	Bilinear Hidden Layer
معنایی	Semantic

Unknown	ناشناخته
Heterogeneous	ناهمگن
Nearst Neighbors	نزدیک‌ترین همسایه
Batch Sampling	نمونه‌برداری دسته‌ای
Positive Example	نمونه‌ی مثبت
Negative Example	نمونه‌ی منفی
Homogeneous	همگن
Global Dependency	وابستگی جهانی
Latent Factor	ویژگی پنهان
Relational Learning	یادگیری رابطه‌ای
Machine Learning	یادگیری ماشین

Abstract:

It has become a hard problem nowadays to use machine learning in order to process the relationships among enormous volumes of data which is being generated by information retrieval, biological computations, natural language processing etc. Various methods have been introduced for handling such data, analysing large knowledge bases and extracting relationships among them. One class of these methods is latent-feature-based methods that map the problem into a multidimensional vector space and then try to solve it with a few parameters. The power of these methods is in their simplicity in learning phase, their need to a few parameters, and their scalability for working on extra-large knowledge bases. These models can discover the relationships between the entities and after a learning phase, find a good classification for the entities. But this is not the case about links between entities and such links could not be recognized as well.

In this research, we intend to use the association rules to discover the relationships between the links and involve these rules in the learning model. Using the solution provided, we were able to otimize RESCAL, NTN, Structured Embedding, TransE, and TransH methods and our optimizations were verified using two quality criteria. We also presented a framework for evaluating the existing methods and discovering their strengthes and weaknesses. In addition to improving current methods, this solution will lead to the extraction of latent facts from knowledge bases so learning of models can be performed using richer knowledge bases.

Keywords: Machine learning, link prediction, latent feature

University of Tehran

College of Engineering

Faculty of Electrical and Computer Engineering

Improvement in Link-prediction Model based on Latent Factor using Association Rules

A thesis submitted to the Graduate Studies Office
In partial fulfillment of the requirements for
The degree of M.Sc in
Software Engineering

By:
Masoud Hashemian

Supervisor:
Dr. Nasser Yazdani