دانشگاه تهران

پردیس دانشکده های فنی

دانشکده مهندسی کامپیوتر

دسته‌بندی بسته در شبکه‌های نرم‌افزار محور

پایان‌نامه جهت دریافت درجه کارشناسی ارشد

در رشته مهندسی کامپیوتر گرایش معماری

نگارش:

محمدرضا پیروزی

استاد راهنما:

دکتر ناصر یزدانی

بهمن ماه ۱۳۹۵


بسم الله الرحمن الرحيم

دانشگاه تهران

پردیس دانشکده های فنی

دانشکده مهندسی کامپیوتر

دسته‌بندی بسته در شبکه‌های نرم‌افزار محور

پایان‌نامه جهت دریافت درجه کارشناسی ارشد

در رشته مهندسی کامپیوتر گرایش معماری

نگارش:

محمدرضا پیروزی

استاد راهنما:

دکتر ناصر یزدانی

بهمن ماه ۱۳۹۵

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**				
۵	**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.

کلیه حقوق مادی و معنوی این اثر متعلق به دانشکده فنی دانشگاه تهران می باشد.

نام و نام خانوادگی دانشجو :

امضای دانشجو :

تقدیم به پدر و مادرم و همسر عزیزم

چکیده

در شـبکه‌های نرم‌افزار محور، یک جریان بر اسـاس تعداد دلخواهی از فیلدها در هر لایه از سـرآیند بسته، قابل تعریف است. به عنوان مثال در نسخه ۱,۳,۱ پروتکل OpenFlow از ۱۵ فیلد برای تعریف یک جریان استفاده می شود. این امر سبب شده تا اولاً عرض مدخل در جدول جریان افزایش یابد، ثانیاً کنترل ریزتر بر روی ترافیک شبکه باعث افزایش تعداد مدخل‌ها در جدول جریان شده است. در نتیجه طراحی یک راه‌حل دسـته‌بندی بسـته در محیط‌هایی که به کارآیی بالا نیاز دارند، به یک چالش بزرگ تبدیل گشته است. برای حل این مسئله در سوئیچ‌های نرم‌افزاری سعی می شود تا حد امکان اندازه فضای نمونه برای مطابقت دادن یک چندتایی با قوانین، کوچک و کوچک‌تر شود. در الگوریتم MC-SBC یک ساختار مبتنی بر درخت تصمیم دو سطحی برای جدول‌های جستجو مطرح شده، و در آن بیت‌های موثر جهت افراز مجموعه قوانین و ایجاد درخت، با استفاده از یک روش آماری مشخص می‌گردند. در این پایان‌نامه از الگوریتم ژنتیک برای پیدا کردن مجموعه بیت‌های موثر جهت افراز مجموعه قوانین استفاده شده است. نتایج گزارش‌شده نشان می‌دهد مجموعه قوانین، با استفاده از این الگوریتم به صورت بهتری افراز می‌شوند، به طوری که هم نسبت تکرار قوانین در آن کاهش می‌یابد، و هم بیشینه تعداد قوانین در برگ‌های درخت کمتر می‌شود. بهبود کیفیت افراز مجموعه قوانین باعث می‌شـود هنگامی که اندازه مجموعه قوانین بزرگ اسـت، سـرعت عمل دسـته‌بندی بسـته‌ها در نمونه پیاده‌سازی شده بر روی پردازنده گرافیکی تا حدود ۱۰ درصد بهبود یابد.

واژه‌های کلیدی: شبکه‌های نرم‌افزار محور، دسته‌بندی بسته، سوئیچ OpenFlow، پردازنده گرافیکی، الگوریتم ژنتیک

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 شـــبکه‌ها انجام گرفته اســـت. داده‌های شـــبکه‌های اجتماعی یکی از ابزار محبوب برای مدل کردن رابطه و رفتار افراد و جامعه یا گروهی که در آن عضو هستن بشمار می‌رود. این داده‌ها معمولا به صورت گرافی نمایش داده می‌شود که در آن گره‌ها افراد و لبه‌ها 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 ماشـــین، 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 اســـت. 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 که می‌توان آن را به صورت یک گراف داده نمایش داده به این صورت که گره‌ها موجودیت‌ها و یال‌های بین گره‌ها نشـــانگر روابط بین آن‌ها باشـــند، که این یال‌ها می‌توانند از یک نوع باشـــند یعنی گراف نشانگر یه 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 شـبکه‌های همگون و ناهمگون اسـت به این معنی که از اطلاعات موجود در گراف دانش اسـتفاده کرده و وجود یا عدم وجود یک یال را پیش‌بینی کرد، یا به عبارت دیگر مسـاله‌ی پیش‌بینی پیوند این ا ست 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 ماشـین مورد اسـتفاده قرار می‌گیرد و کاربرد گ سترده‌ای در زندگی انسان پیدا کرده ا ست. 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 تک‌رابطه‌ای بوده ا ست. به این معنا که روابط بین موجودیت‌ها از یک نوع اسـت و این روابط به صـورت دوتایی‌های مرتب مورد اسـتفاده قرار می‌گیرند. برای مثال اگر در یک شـبکه اجتماعی رابطه را دوسـتی 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 دانشـی که اخیرا معرفی شـده‌اند عمدتا داده‌ها را به صـورت داده‌های چند رابطه‌ای ذخیره می‌کنند و اطلاعات بیشـتری از یک رابطه دوتایی بلی یا خیر به ما می‌دهند. منظور از داده‌های چند رابطه‌ای گراف جهت‌داری اسـت متشـکل از موجودیت‌ها و روابط بین آن‌ها که بصـورت «مبدا h، رابطه r، مقصـد t» نمایش داده می‌شـود، به این معنی که یک رابطه r بین موجودیت‌های h و t وجود دارد. برای مثال سـه‌تایی «تهران، واقع در، ایران» این اطلاع را به ما می‌دهد که اسـتان تهران داخل کشور ایران قرار دارد. در این نوع پایگاه دانش هم انواع مختلف موجودیت وجود دارد و هم انواع مختلف رابطه بین موجودیت ها. پایگاه های دانشـی مانند Graph Google Knowledge، FreeBase و WordNet وجود دارند که شامل تعداد زیادی نمونه چندرابطه‌ای می‌باشند و تعداد زیادی موجودیت و روابط بین آن‌ها را می‌توان در آن‌ها یافت و از آن برای یادگیری مدل استفاده کرد.
[۵].

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 ویژگی های پنهان، روش های ویژگی های گراف و ...، تمرکز ما در این مقاله بر روی روش های 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. 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 تاثیر گذار 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 ثابتی برای حل مساله استفاده می‌کنند و هر کدام با نوآوری‌هایی که داشته‌اند بهبودهایی در نتایج بدست‌آمده حاصل کرده‌اند. این روش‌ها در بخش + به طور 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 روش‌ها معروف مبتنی بر ویژگی‌های پنهان که به حل مساله‌ی پیش‌بینی پیوند پرداخته‌اند را انتخاب می‌کنیم و در بخش + 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 کردن قوانین انجمنی در بخش + 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 پایگاه‌دانش در فرایند آموزش استفاده شده است. مثلا اگر در پایگاه‌دانش 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 نگرفته اند. در این پژوهش قصد داریم که اینگونه قوانین را به کمک روش‌هایی که در بخش + توضیح داده می شوند از پایگاه دانش استخراج کنیم و به کمک معیارهای کیفیت که در بخش + معرفی می شوند 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 + توضیح داده شده است.

۳-۱- کاربردهای پژوهش

مسائل پیش‌بینی پیوند در زمینه‌های زیادی قابل بکارگیری 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 یه عنوان 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 ξکالای ψ را خواهد خرید یا خیر یا اینکه کالای ξ 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 پایگاه‌دانش بدست می‌آید و کیفیت لازم را دارد را نیز به پایگاه دانش اضافه کنیم و آن را کامل‌تر کنیم.

۴-۱- ساختار پایان‌نامه

ادامه این پایان‌نامه بدین صورت سازمان‌دهی شده است: در فصل.

فصل ۲: زمینه‌های تحقیق

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

۲-۱- شبکه‌های همگون و ناهمگون

در سال‌های اخیر شبکه‌های اجتماعی پیشرفت زیادی داشته است و در زمینه‌های مختلفی شبکه سازی شده است. عمدتا در این شبکه‌ها روابط خاصی مد نظر و قابل استخراج است، مثل روابط دوستی، همکاری و ... که اگر گراف این گونه شبکه‌ها را رسم کنیم در این گراف‌ها 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 [X]. در شکل + یک نمونه شبکه‌ی همگون که در آن اشخاص و رابطه‌ی دوستی بین آن‌ها به تصویرکشیده شده است را مشاهده می‌کنیم.

عکس شبکه‌ی معمولی

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 پایگاه‌دانش[1] نیز می‌نامیم. در شکل + 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.

عکس شبکه‌ی ناهمگون

۲-۲- 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 به عنوان ورودی مساله دریاف می‌کنیم و وجود یا عدم وجود یک یال بین دو موجودیت را پیش‌بینی می‌کنیم. در شکل + یک گراف از یک شبکه‌ی ناهمگون را مشاهده می‌کنیم. مساله‌ی پیش‌بینی لینک تلاش می‌کند که بررسی کند که رابطه‌ی + بین گره‌های + و + قرار دارد یا نه. این پیش‌بینی عمدتا از روی دیگر روابط مرتبط بین موجودیت‌ها انجام می‌شود و با بررسی شباهت‌ها و معیارهایی که در ادامه‌ی این پژوهش توضیح خواهیم داد تصمیم بگیرد که پیوند رابطه‌ی + بین موجودیت‌های + و + برقرار هست یا خیر.

شکل گراف با سوال

راه‌کارهای مختلفی برای حل این مساله مطرح شده است که به طور کلی می‌توان این راه‌کارها را به سه دسته‌ی ۱- روش‌های مبتنی بر ویژگی‌های گراف ۲- روش‌های مدل تصادفی مارکوف ۳- روش‌های مبتنی بر ویژگی‌های پنهان تقسیم کرد که در این پژوهش تمرکز ما روی دسته‌ی سوم یعنی

[1] Knowledgbase

روش‌های مبتنی بر ویژگی‌های پنهان خواهد بود و در بخش + این روش‌ها را به تفصیل توضیح خواهیم داد.

۳-۲- فرض جهان باز[۱] و فرض جهان بسته[۲]

فرض‌های جهان باز و جهان بسته در سیستم رسمی منطق[۳] برای بازنمایی دانش[۴] مورد استفاده قرار می‌گیرد. در فرض جهان بسته در نظر می‌گیریم که اگر داده‌ای در پایگاه دانش نبود، آن داده را غلط فرض می‌کنیم [x] . برای مثال اگر در پایگاه دانش حقیقت «x, r, y» که به معنی این است که x با y رابطه‌ی r را دارد وجود نداشته باشد، می‌توانیم در نظر بگیریم که این حقیقت اشتباه است و مطمئنیم که x با y رابطه‌ی r ندارد.

اما در فرض جهان باز اینگونه نیست و اگر حقیقتی در پایگاه دانش وجود نداشته باشد نمی‌توانیم با اطمینان بگوییم که آن حقیقت اشتباه است، و ممکن است صحیح باشد [x] . وجود این فرض از کامل نبودن پایگاه‌های دانش ناشی می‌شود که وقتی نمی‌توانیم همه‌ی اطلاعات موجود در مورد موضوع 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[۵] در نظر می‌گیریم.

دسته‌ی دیگر فرض جهان بسته‌ی محلی است، که این فرض میانه‌ای بین دو فرض قبل است...

---

1 Open world assumption (OWA)

2 Closed world assumption (CWA)

3 Formal system of logic

4 Knowledge representation

5 unknown

۴-۲- گرادیان نزولی تصادفی[1] (SGD)

[1] Stochastic gradient descent

فصل ۳: پیشینه پژوهش

در این بخش به معرفی روش‌های موجود که به حل مساله‌ی پیش‌بینی پیوند در شبکه‌های ناهمگون می‌پردازند، خواهیم پرداخت. روش‌هایی با راه‌کارهای مختلفی در حوزه‌های مختلف تلاش به حل این مساله کرده‌اند.این روش‌ها می‌توانند به سه دسته‌ی مختلف تقسیم می‌شوند: ۱- روش‌های مبتنی بر ویژگی‌های گراف ۲- روش‌های مدل تصادفی مارکوف ۳- روش‌های مبتنی بر ویژگی‌های پنهان. 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 مارکف 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.

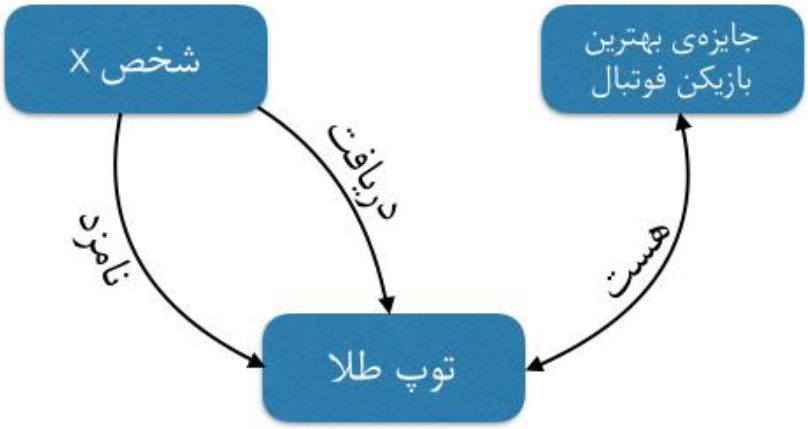
همان‌طور که در بخش XX گفتیم تمرکز ما در این تحقیق روی روش‌های مبتنی بر ویژگی‌های پنهان است، در ادامه این روش‌ها را به صورت کامل توضیح داده و تعدادی از آن‌ها را به طور مختصر معرفی کرده و نحوه‌ی کار آن‌ها را توضیح می‌دهیم و نتایج بدست آمده از آن‌ها را توضیح می‌دهیم.

۱-۳- روش‌های مبتنی بر ویژگی‌های پنهان

روش‌های مبتنی بر ویژگی‌های پنهان از جدید ترین راه‌کارهایی اســـت که در حوزه‌ی پیش‌بینی پیوند روی پایگاه های دانش موجود معرفی شـــده اســـت.همانطور که در بخش قبل گفتیم این روش از ویژگی‌هایی که در نگاه اول از پایگاه‌دانش بردا شت نمی شود ا ستفاده می‌کند که ویژگی‌های پنهان نام دارند، در همه‌ی روش‌های مبتنی بر ویژگی پنهان این ویژگی‌ها را به صـــورت برداری تعریف می‌کنیم که هر مولفه از این بردار نشان دهنده‌ی یک ویژگی می‌باشد.

برای مثال در رابطه‌ی دوســـتی فاکتورهایی تاثیر گذار هســـتن و اگر در پایگاه دانش همه‌ی رابطه‌های دو ستی موجود را بررسی کنیم به یک بردار از فاکتورها می‌رسیم که بردار رابطه‌ی دو ستی 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 دو ستی تاثیر دارند و اهمیت هر کدام چقدر ا ست و این میزان اهمیت را چگونه در ت شخیص 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 –شخصx 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.



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

در ادامه ابتدا پایگاه‌دانش 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.

۲-۳- 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. 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. 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 پایگاه‌دانش 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 پایگاه‌دانش به نام Freebase15k استفاه می‌کنیم که در آن ۱۴۹۵۱ موجودیت، ۱۳۴۵ رابطه‌ی مختلف و ۵۹۲۲۱۳ حقیقت وجود دارد.

	Entities (n_e)	Rel. (n_r)	Train. Ex.	Valid. Ex.	Test Ex.
FB13	75,043	13	316,232	5,908	23,733
FB15k	14,951	1,345	483,142	50,000	59,071
FB1M	1×10^6	23,382	17.5×10^6	50,000	177,404

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 آمورش 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 – تقسیم و استفاده شده است، به این صورت که برای قسمت آموزش 483,142 حقیقت استفاده کرده‌ایم، برای قسمت ارزیابی 50k حقیقت استفاده کرده‌ایم و برای قسمت آزمون 59k حقیقت استفاده کرده‌ایم.

۳-۳- 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 بهدی را شروع می‌کنیم.

در ادامه این سه مرحله را توضیح می‌دهیم.

۱-۳-۳- 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.

۲-۳-۳- 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 می‌کنم.

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

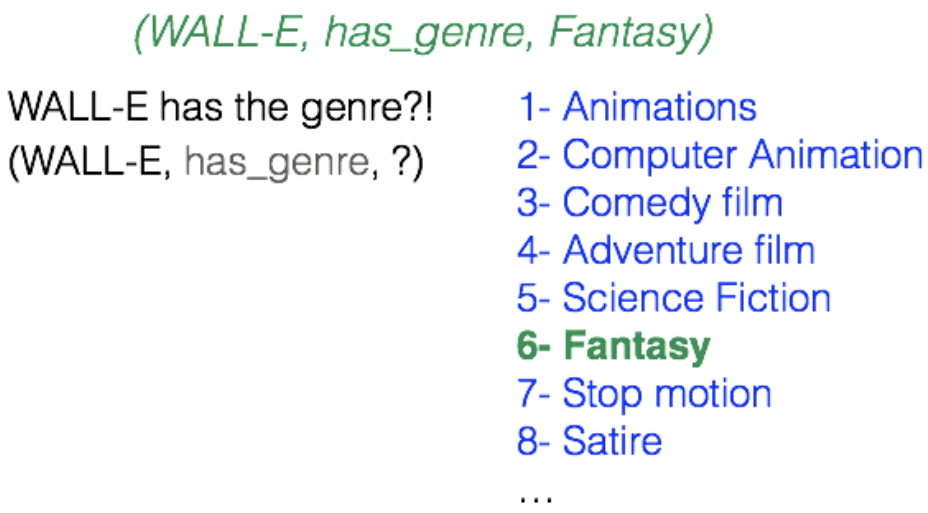
(WALL-E, has_genre, Fantasy)

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 زیر می‌باشد:



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 بررسی کنیم که مدل با توجه به این معیارها بهبود داشته یا خیر. در این تحقیق از دو معیار زیر استفاده شده است:

- 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.

۳-۳-۳- مرحله‌ی آزمون:

پس از انجام کامل مراحل آموزش و ارزیابی و متوقف شدند الگوریتم از داده‌ای آزمون که در بخش – توضیح دادیم و معیارهای کیفیت که در بخش – 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 به 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.

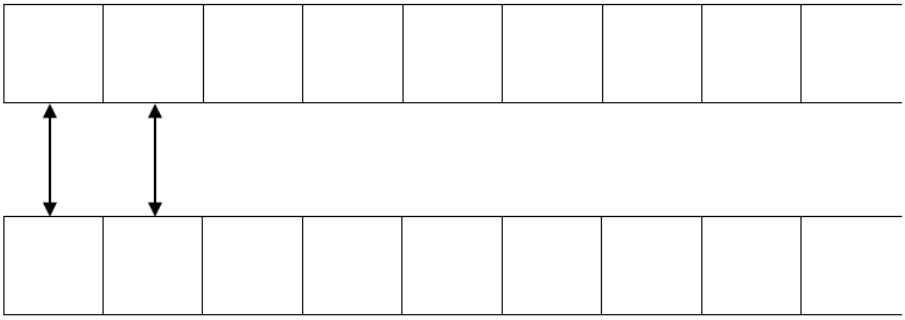
۳-۴- بررسی روش‌های موجود

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

۳-۴-۱- روش‌های خطی

همانطور که در بخش – گفتیم در روش‌های مبتنی بر ویژگی‌های پنهان موجودیت‌ها و رابطه‌ها به صورت بردارهایی در یک فضای n بعدی تبدیل می‌شوند که به کمک معیارهای مختلف شباهت،

رابطه‌ی بین دو موجودیت را بدســـت می‌آوریم. در روش‌های خطی در زمان آموزش و بررســـی میزان شـباهت بردارها را به صـورت خطی با یکدیگر مقایسـه می‌کنیم، به این صـورت که هر اندیس از بردار موجودیت اول را با اندیس متناظر آن در بردار رابطه یا موجودیت دیگر بررسی می‌کنیم، شکل +.

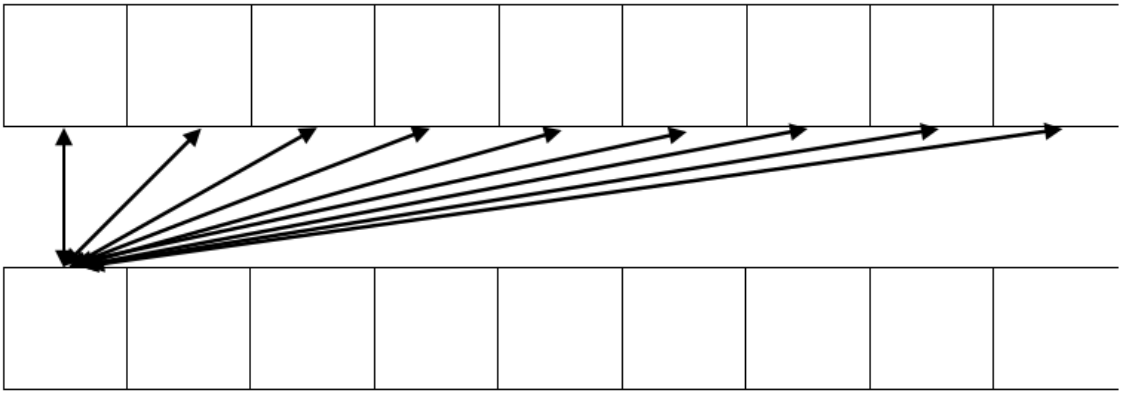


با توجه به نتایج ضعیفی که روش‌های این دسته در آموزش مدل و پیش‌بینی پیوند بدست آورده اند به این روش‌ها نمی‌پردازیم و به همین معرفی کفایت می‌کنیم.

۲-۴-۳- روش‌های دوخطی – رسکال

روش RESCAL [] یک روش ویژگی‌های پنهان رابطه‌ای[1] 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 +. به همین دلیل این روش را روش دوخطی نیز می‌نامیم.

[1] Relational latent factor



در این روش امتیاز هر سه‌تایی را از رابطه‌ی + 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*H_e است که هر اندیس w_{abk} نشانگر این است که ویژگی پنهان a و b در رابطه‌ی k چقد با هم کنش می‌کنند.

$$f_{ijk}^{\text{RESCAL}} := \mathbf{e}_i^\top \mathbf{W}_k \mathbf{e}_j = \sum_{a=1}^{H_e} \sum_{b=1}^{H_e} w_{abk} e_{ia} e_{jb}$$

همانطور که در رابطه‌ی + 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 را 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 + 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 + 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 صورت مشترک[3] 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[4] در داده‌ها را تشخیص دهیم[1].

ارتباط معنایی بردارها[5]: خاصیت بازنمایی مشترک در این روش کمک می‌کند که میزان شباهت موجودیت‌ها در فضای رابطه‌ای[6] 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

[1] Relational Learning

[2] Shared representations

[3] jointly

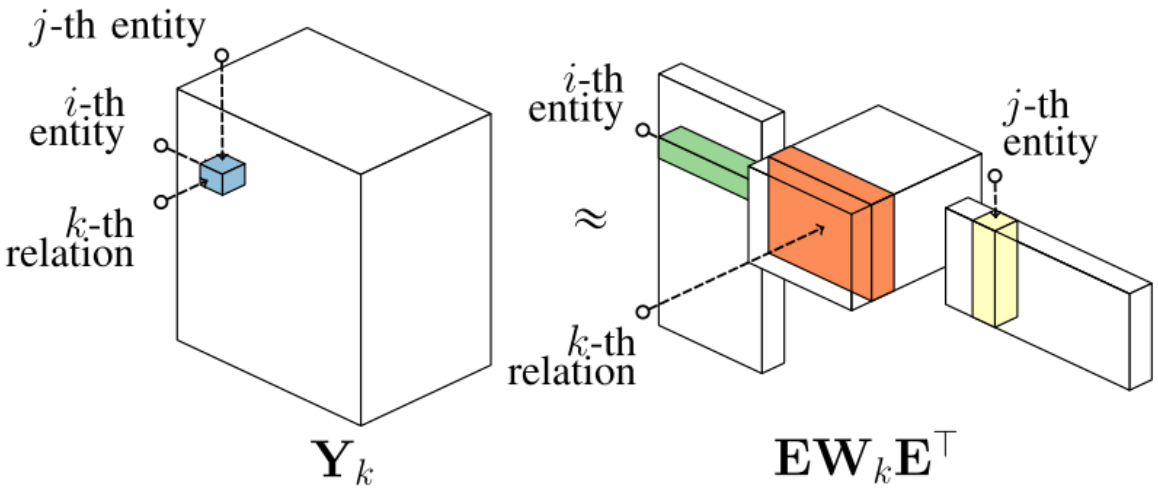
[4] global dependency

[5] Semantic embeddings

[6] relational domain

رابطه‌ی مشترک بازنمایی یکسانی خواهند داشت. این خصوصیت می‌تواند در بازنمایی[1] و خوشه‌بندی‌های با مقیاس بالای[2] موجودیت‌ها روی داده‌های رابطه‌ای مورد استفاده قرار گیرد.[63,64,1]

ارتباط با عامل‌بندی تنسور[3]: این روش شباهت زیادی به روش‌های استفاده شده در سیستم‌های توصیه‌گر[66] و عامل‌بندی تنسور سنتی دارد]۶۷[. ضرب ماتریس که در معادله‌ی + معرفی کردیم می‌تواند به صورت $F_k = EW_kE^T$ نوشته شود که در آن $F \in R^{Ne*Ne}$ ماتریسی است که همه‌ی امتیازات مربوط به رابطه‌ی k را در خود جا داده است و سطر i ام از ماتریس $E \in R^{Ne*He}$ بازنمایی موجودیت e_i است که برداری از ویژگی‌های پنهان این موجودیت است. در شکل + این تبدیل ماتریس به تنسور نمایش داده شده است.[1]



[1] resulotion

[2] large-scale hierarchical clustering

[3] Tensor factorization

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][68] . این روش می‌توانیم به صورت یک روش بر مبنای امتیاز حل کرد که می‌تواند از 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.[63,64] به صورت تحلیلی می‌توان نشان داد که در 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.[64]

پیش‌بینی مجزا: در رابطه‌ی + 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.[1]

نتایج یادگیری رابطه‌ای: رسکال در زمینه‌های مختلف مدل‌های یادگیری مدرن[5] مورد استفاده قرار گرفته است. مثلا در [63] نشان داده شده است که رسکال موفق شده در پیش‌بینی رابطه نتایج نزدیک

[1] gradient-based minimization

[2] stochastic gradient descent

[3] sequence

[4] efficient close-form update

[5] state-of-the-art

یا بهتر روی چندین مجموعه داده‌ی معیار نسبت به روش‌های [70] Markov Logic Networks و [71, 72] the Infinite (Hidden) Relational model و Bayesian Clustered Tensor Factorization [73] بدست آورد. همچنین رسکال برای پیش‌بینی پیوند روی کل پایگاه‌دانش مانند YAGOو DBpedia مورد استفاده قرار گرفته است.[74 ,64] فارق از پیش‌بینی پیوند، رسکال در روش‌های یادگیری رابطه‌ای تصادفی (SRL)[1] مانند بازنمایی موجودیت‌ها[2] و خوشه‌بندی بر مبنای پیوند[3] عملکرد موفقی داشته است. برای مثال این روش عملکرد موفقی در دسته‌بندی نویسندگان، ناشران و سالن‌های انتشار روی مجموعه داده‌های ناشران داشته است[63,65]. علاوه بر این، ارتباط معنایی موجودیت‌ها که در این روش محاسبه شد در ایجاد طبقه‌بندی[4] به کمک خوشه‌بندی سلسله مراتبی[5] روی داده‌های دسته‌بندی نشده[6] به کار برده شده است [75].

[1] stocastic relational learning

[2] entity resulotion

[3] link-based clustring

[4] taxonomies

[5] hierarchical clusterings

[6] uncategorized data via

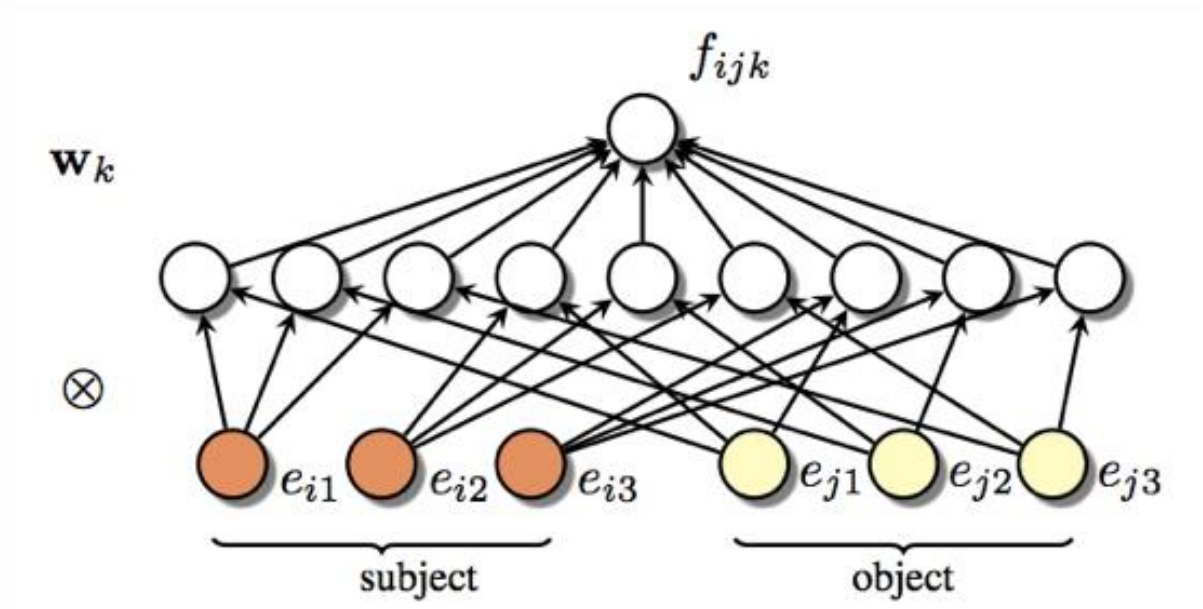
۳-۴-۳- روش‌های ادراک چندلایه‌ای[1]

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 + و + بازنویسی کنیم.

$$f_{ijk}^{\mathrm{RESCAL}} := \mathbf{w}_k^{\top} \boldsymbol{\phi}_{ij}^{\mathrm{RESCAL}}$$
$$\boldsymbol{\phi}_{ij}^{\mathrm{RESCAL}} := \mathbf{e}_j \otimes \mathbf{e}_i,$$

در روابط + ضرب داخلی e_i و e_j که در محاسبه‌ی مجموع امتیازها شرکت می‌کردند را از مجموع بیرون کشیده و به صورت ضرب خاجی نوشتیم. بنابراین رسکال بازنمایی جفت موجودیت i و j را به صورت ضرب تنسور ویژگی‌های پنهان این دو موجودیت بدست آورد رابطه‌ی + و وجود حقیقت x_{ijk} را از روی بازنمایی جفت موجودیت‌ها (φ_{ij}) و ماتریس وزن‌دار رابطه (w_k) بدست آورد. این تقسیم‌بندی رابطه‌ی رسکال در شکل + به تصویر کشیده شده است. برای توضیح بیشتر در مورد ایجاد بازنمایی پنهان به وسیله‌ی ضرب تنسور به [88, 89, 90] مراجعه شود.

[1] Multi-layer perceptrons



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.

مدل E-MLP[2]:

[1] feedforward neural networks

[2] Entity-MLP

در ابتدا مدل Entity-MLP را معرفی می‌کنیم. تابع امتیاز این مدل به صورت معادلات + و + و + محاسبه می‌گردد.

$$
\begin{aligned}
f_{ijk}^{\text{E-MLP}} &:= \mathbf{w}_k^\top \mathbf{g}(\mathbf{h}_{ijk}^a) \\
\mathbf{h}_{ijk}^a &:= \mathbf{A}_k^\top \boldsymbol{\phi}_{ij}^{\text{E-MLP}} \\
\boldsymbol{\phi}_{ij}^{\text{E-MLP}} &:= [\mathbf{e}_i; \mathbf{e}_j]
\end{aligned}
$$

در روابط فوق g(u) = [g(u1),g(u2),. ..] 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 ha یک لایه‌ی مخفی[1] اضافه شده است که ماتریس وزن‌دار دیگر روی بازنمایی موجودیت‌ها اعمال می‌کند. در اصل ما در رابطه‌ی + دو موجودیت ei و ej را ترکیب کردیم و هیچ کنشی بین این دو موجودیت محاسبه و تاثیر داده نشده است، بنابراین نیاز به یک ماتریس وزن‌دار برای محاسبه‌ی تاثیر این دو موجودیت نیاز بود و ha وارد معادله شد.

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 Ak مورد بررسی قرار می‌گیرد. این راهکار به طور قابل ملاحضه‌ای تعداد پارامتری که باید آموزش دیده شوند را کاهش می‌دهد.

روش ER-MLP[2]:

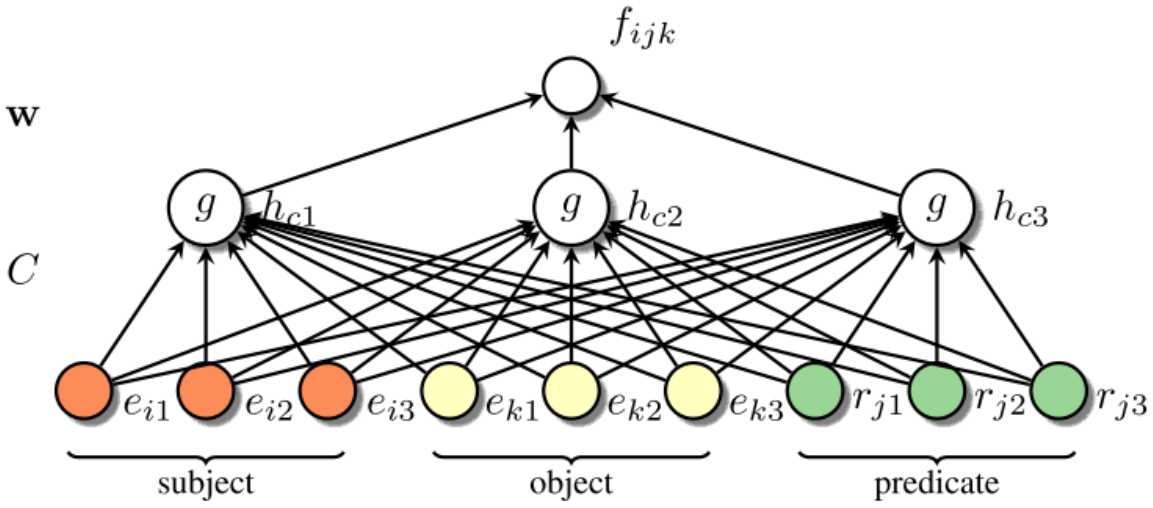
[1] hidden layer

[2] Entity-Relation-MLP

یکی از اشکالات روش E-MLP این است که باید برای هر رابطه‌ی ممکن، یک بردار wk و یک ماتری Ak 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 + و + و + خواهند بود.

$$
\begin{aligned}
f_{ijk}^{\text{ER-MLP}} &:= \mathbf{w}^\top \mathbf{g}(\mathbf{h}_{ijk}^{c}) \\
\mathbf{h}_{ijk}^{c} &:= \mathbf{C}^\top \boldsymbol{\phi}_{ijk}^{\text{ER-MLP}} \\
\boldsymbol{\phi}_{ijk}^{\text{ER-MLP}} &:= [\mathbf{e}_i; \mathbf{e}_j; \mathbf{r}_k].
\end{aligned}
$$

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 + نمایش داده شده است.



در [91] 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 [28] این ویژگی نشان داده شده است، برای مثال به کمک MLP مساله‌ی نزدیک‌ترین همسایه[2] برای بازنمایی پنهان چندین رابطه‌ی انتخاب شده از پایگاه‌دانش 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

۴-۴-۳- شبکه‌های عصبی تنسور[1]

در [92] 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 + و + و + است.

$$
\begin{aligned}
f_{ijk}^{\mathrm{NTN}} &:= \mathbf{w}_k^\top \mathbf{g}([\mathbf{h}_{ijk}^a; \mathbf{h}_{ijk}^b]) \\
\mathbf{h}_{ijk}^a &:= \mathbf{A}_k^\top [\mathbf{e}_i; \mathbf{e}_j] \\
\mathbf{h}_{ijk}^b &:= \left[\mathbf{e}_i^\top \mathbf{B}_k^1 \mathbf{e}_j, \ldots, \mathbf{e}_i^\top \mathbf{B}_k^{H_b} \mathbf{e}_j\right]
\end{aligned}
$$

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 +و+و+ مشخص است که این روش مشکلات مقیاس پذیری که در هر دو روش E-MLP و RESCAL وجود داشت را دارد. همچنین در [95] و [28] نشان داده شده است که این روش به بیش‌برازش[3] میل می‌کند (حداقل روی مجموعه داده‌هایی که در این مقالات استفاده شده است)[1].

[1] Neural tensor networks

[2] bilinear hidden layer

[3] overfit

۵-۴-۳- روش‌های فاصله‌ی پنهان[1]

کلاس دیگری از مدل‌ها، مدل‌های فاصله‌ی پنهان هستند (همچنین در تحلیل شبکه‌های اجتماعی به مدل‌های فضای پنهان معروفند) که احتمال وجود رابطه بین بین موجودیت‌ها را از فاصله‌ی بین بازنمایی پنهان آن‌ها در فضا می‌سنجد: موجودیت‌ها با یکدیگر رابطه دارند اگر بازنمایی پنهان آن‌ها با یک معیار فاصله نزدیک به هم باشد [۱].

در [96] مدلی برای داده‌های تک-رابطه‌ای[2] برای اولین بار راهکاری در این زمینه معرفی شده است. این راهکار احتمال وجود پیوند در شبکه‌های اجتماعی را با تابع امتیاز f(ei,ej)= -d(ei, ej) محاسبه می‌کند، که در آن d(.,.) یک تابع اندازه‌گیری فاصله‌ی دلخواه مانند فاصله‌ی اقلیدسی[3] است.

مدل **Structured embedding**

در [93] مدلی به نام structured embedding (SE) معرفی شده که در آن ایده‌ای که در بند قبل توضیح داده شد برای داده‌های چند-رابطه‌ای[4] گسترش داده شده است. در این روش تابع امتیاز برای حقیقت‌های xijk به صورت + مدل شده است.

$$f_{ijk}^{\mathrm{SE}} := -\|\mathbf{A}_k^s \mathbf{e}_i - \mathbf{A}_k^o \mathbf{e}_j\|_1 = -\|\mathbf{h}_{ijk}^a\|_1$$

[1] Latent distance models

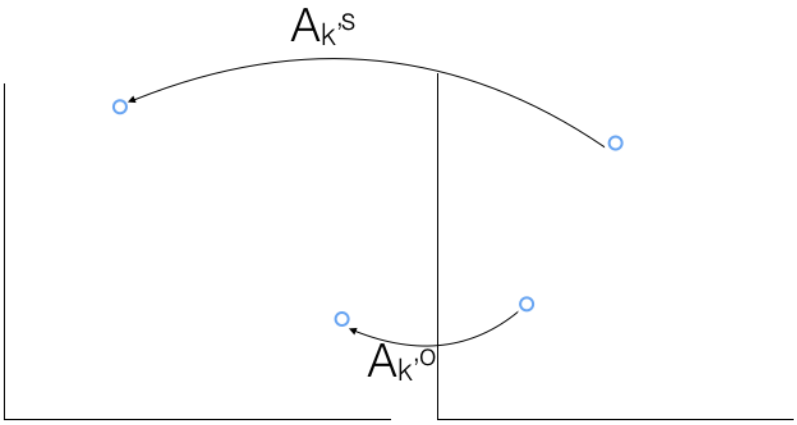
[2] uni-relational data

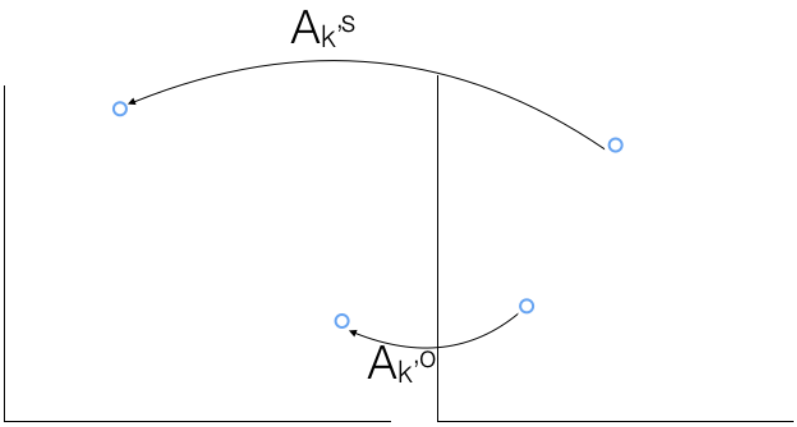
[3] Euclidean distance

[4] multi-relational

در رابطه‌ی + Ak از ماتریس‌های مربوط به موجودیت‌های اول و دوم [Aks, Ako] تشکیل شده است. ماتریس‌های Aks و Ako 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 + 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 Ako وAks این دو موجودیت به یکدیگر نزدیکتر شده اند. همینطور در شکل + مشاهده می‌شود که در صورتی که رابطه‌ی k’ بین دو موجودیت i و j برقرار نباشد، پس از نگاشت این دو موجودیت توسط ماتریس‌های نگاشت Ak’s و Ak’o دو موجودیت در فاصله‌ی بیشتری از یکدیگر قرار می‌گیرند.





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.

۶-۴-۳- مدل TransE[1]

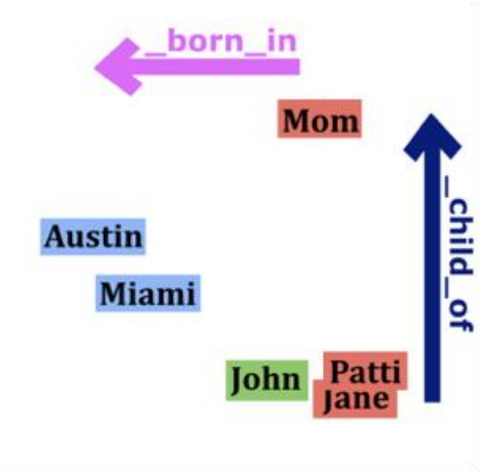
در [94] مدلی برای حل مشکل تعداد پارامتر زیاد در روش SE مطرح شده است که بجای اینکه از ماتریس‌های Aks و Ako برای تاثیر دادن رابطه‌ها استفاده شود، رابطه را یک بردار همانند بردار موجودیت‌ها در نظر گرفته شده و به عنوان یک آفست در کنار موجودیت اول در محاسبات از آن استفاده می‌کند. [1] به طور خاص امتیاز حقیقت xijk از رابطه‌ی + بدست می‌آید.

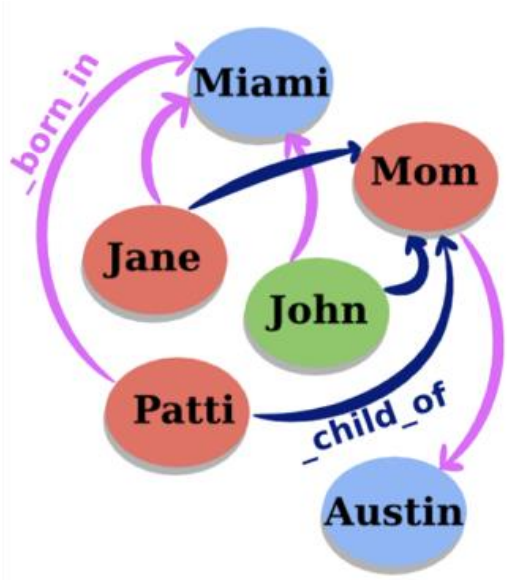
$$f_{ijk}^{\text{TransE}} := -d(\mathbf{e}_i + \mathbf{r}_k, \mathbf{e}_j).$$

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 + یک گراف را مشاهده می‌کنیم که در آن روابط فرزندی و مکان تولد بین ۶ موجودیت نمایش داده شده است. در شکل + 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.

[1] Translation Embedding

مثلا اگر john را توسط بردار Child_of منتقل کنیم به موجودیت Mom نزدیک می‌شود که حقیقت (john, Child_of, Mom) را تشکیل می‌دهد. اما اگر همین موجودیت را توسط بردار رابطه‌ی born_in منتقل کنیم، به موجودیت Miami نزدیک می‌شود که حقیقت (john, born_in, Miami) را تشکیل می‌دهد.





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 (e1+r) را با موجودیت دوم (e2) کمینه کند. همانطور که قبل‌تر گفتیم تابع امتیاز در این روش‌ها یک تابع فاصله است، پس در اینجا هدف کاهش فاصله بین (e1+r) و e2 است که بصورت + نوشته می‌شود.

$$d(e1, r, e2) = ||e1 + r – e2||$$

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 انجایی که

پایگاه‌های دانش مورد استفاده از فرض جهان‌باز پیروی می‌کنند وقتی حقیقت (e1, r, e2) در پایگاه دانش 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 +). در رابطه‌ی + 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.

$$S' = \{(\text{sub}',\text{rel},\text{obj})|\text{sub}' \in \mathcal{E}\} \cup \{(\text{sub},\text{rel},\text{obj}')|\text{obj}' \in \mathcal{E}\}$$

$$\sum_{pos} \sum_{neg \in S'} [\gamma + ||\mathbf{s} + \mathbf{r} - \mathbf{o}||_2^2 - ||\mathbf{s}' + \mathbf{r} - \mathbf{o}'||_2^2]_+$$

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 + 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

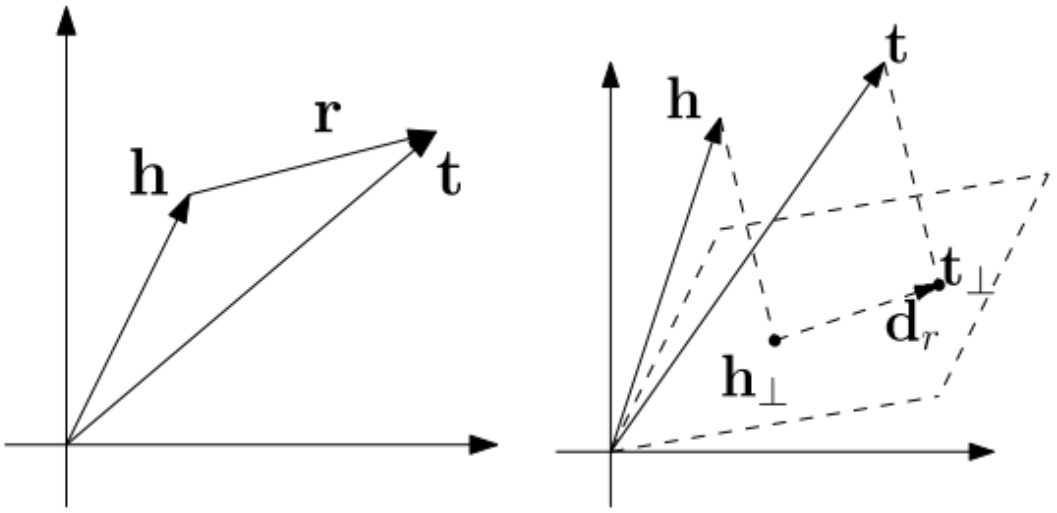
۷-۴-۳- مدل TransH[1]

در بخش قبل مشاهده کردیم که TransE روش مناسبی را ارائه داد با تعداد پارامتر کم و مقیاس‌پذیر که قادر به آموزش مدل است. در [] 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 این روش TransE نداریم.با هزینه و پیچیدگی کمی مانند روش TransE، را نیز در نظر گرفت.



(a) TransE (b) TransH

همانطور که در شکل – 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 – نشان داده شده است ابتده 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⊥ و th⊥به صورت معادله‌ی – بدست می‌آیند که در آن wr بردار انتقال به ابرصفحه‌ی مربوط به رابطه‌ی r است و dr معادل بردار r در ابرصفحه‌ی رابطه است.

$$\mathbf{h}_\perp = \mathbf{h} - \mathbf{w}_r^\top \mathbf{h}\mathbf{w}_r, \quad \mathbf{t}_\perp = \mathbf{t} - \mathbf{w}_r^\top \mathbf{t}\mathbf{w}_r.$$

پس تابع امتیاز این روش به صورت زیر خواهد بود:

$$f_r(\mathbf{h}, \mathbf{t}) = \|(\mathbf{h} - \mathbf{w}_r^\top \mathbf{h}\mathbf{w}_r) + \mathbf{d}_r - (\mathbf{t} - \mathbf{w}_r^\top \mathbf{t}\mathbf{w}_r)\|_2^2.$$

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 fr که در معادله‌ی ـ مطرح کردیم از تابع امتیاز fr که در ـ معرفی کردیم استفاده می‌کنیم و تابع کمینه‌سازی به صورت ـ خواهد شد.

$$\mathcal{L} = \sum_{(h,r,t)\in\Delta} \sum_{(h',r',t')\in\Delta'_{(h,r,t)}} [f_r(\mathbf{h}, \mathbf{t}) + \gamma - f_{r'}(\mathbf{h}', \mathbf{t}')]_+$$

فصل ۴: قوانین انجمنی

در این قسمت سعی بر این داریم که نقطه ضعفی که در بخش + از آن صحبت کردیم را به کمک قوانین انجمنی حل کنیم. در ابتدار ...

۴-۱- 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[۱]، پوشش سر[۲] و معیار اطمینان استاندارد[۳] است.

[1] support rule

[2] head coverage

[3] Standard confidence

۱-۱-۴- معیار پشتیبانی قانون:

این معیار برای میزان اهمیت یک قانون استفاده می‌شود و به این صورت تعریف می‌شود: تعداد قوانین یافت شـــده در پایگاه‌دانش که یک حقیقت را نتیجه می‌دهند. برای مثال در + قانون R معرفی شـــده است که مشخص می‌کند هر شخص x که در شهر y زندگی می‌کند، در آن شهر متولد شده است. به تعداد دفعاتی که این دو حقیقت در پایگاه‌دانش اتفاق بیافتد و قانون R را نقض نکند، پشـــتیبان این قانون گفته می‌شود.

$$R: livesIn(x, y) \Rightarrow wasBornIn(x, y)$$

این معیار به صورت رابطه‌ی + تعریف می‌شود.

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

در رابطه‌ی + B مجموعه‌ای از حقیقت‌ها اســـت که نتیجه میدهد حقیقت r(x,y) را، به تعداد باری که این اتفاق می‌افتد معیار پشتیبانی قانون گفته می‌شود. در جدول + یک پایگاه‌دانش متشکل از دو رابطه و ۵ حقیقت وجود دارد. قانون + را در نظر بگیرید، میزان پشتیبان قانون R در این جدول برابر ۱ است، بخاطر وجود حقیقت (Adam, LivesIn, Paris) و (Adam, wasBornIn, Paris) که از قانون R پیروی می‌کنند [AMIE].

livesIn	*wasBornIn*
(Adam, Paris)	(Adam, Paris)
(Adam, Rome)	(Carl, Rome)
(Bob, Zurich)	

۲-۱-۴- معیار پوشش سر:

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

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

$$size(r) := \#(x',y') : r(x',y')$$

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 + مقدار معیار پوشش سر بخاطر وجود (Carl, Rome) برابر با ۰,۵ است [AMIE].

۳-۱-۴- 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.

در این معیار میزان پشتیبان هر قانون را بر تعداد باری که قسمت بدنه‌ی این قانون در پایگاه‌دانش دیده شده 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.

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

رابطه‌ی بالا به زبان ساده برابر است با تعداد رخداد قانون بخش بر تعداد دفعاتی که می‌توانست رخ دهد. برای مثال در پایگاه‌دانش معرفی شده در جدول + 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 پایگاه‌دانش 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 جهان بسته 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 پایگاه‌دانش دلیلی بر غلط بودن آن حقیقت بوده است. اما همانطور که در بخش + اشاره کردیم فضای این مساله فرض جهان باز است و حقیقت‌هایی که در این پایگاه‌های دانش وجود ندارند لزوما غلط نیستند[AMIE]. برای درک بیشتر این موضوع به مثال زیر دقت کنید:

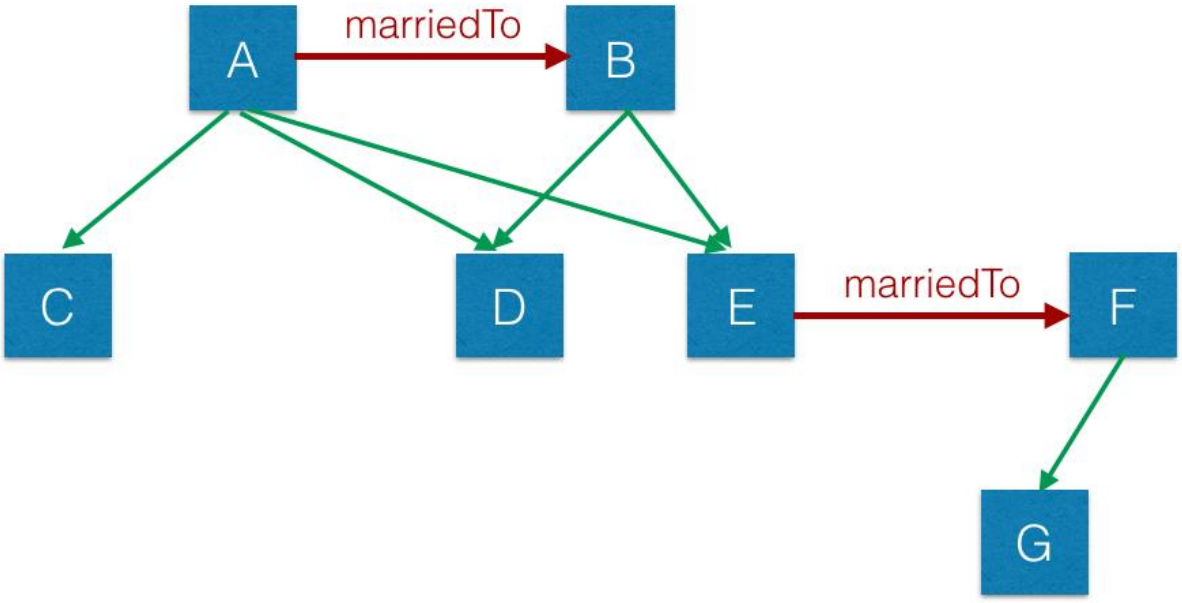
به گراف ارائه شده در شکل + دقت کنید. در این گراف رابطه‌های افقی رابطه‌ی ازدواج (marriedTo) و روابط عمودی رابطه‌ی داشتن فرزند (hasChild) را مشخص می‌کنند. در این گراف قصد داریم که قانون R با تعریف زیر را بررسی کنیم:

hasChild(y, x), marriedTo(y, z) => hasChild(z, x)

[1] positive example

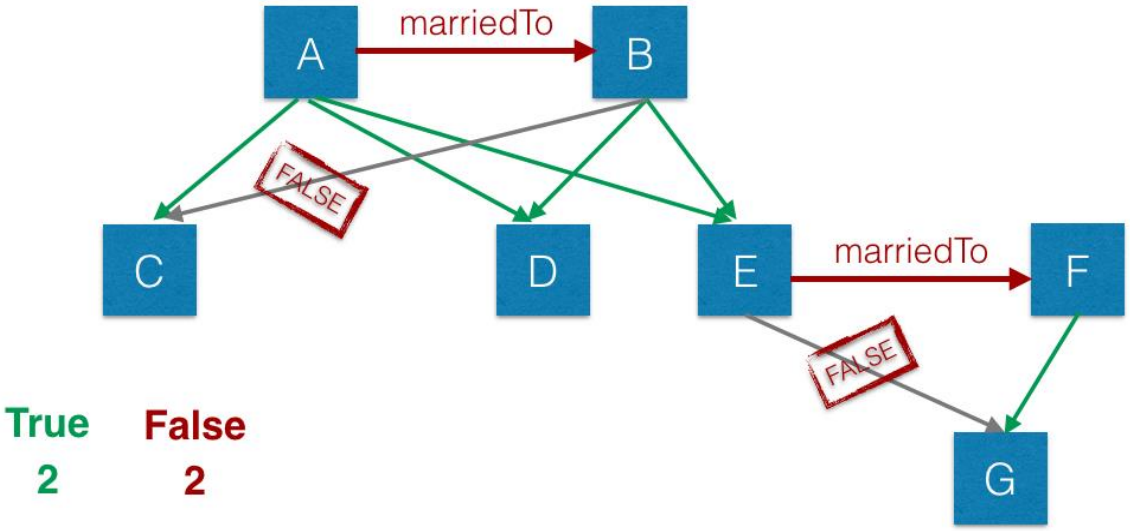
[2] negative example

این قانون به این معنا ست که اگه شخص y فرزندی به نام x دا شته با شد و همین شخص با شخص دیگری به نام z ازدواج کرده باشد، میتوان نتیجه گرفت که شخص z هم فرزندی به نام x دارد.

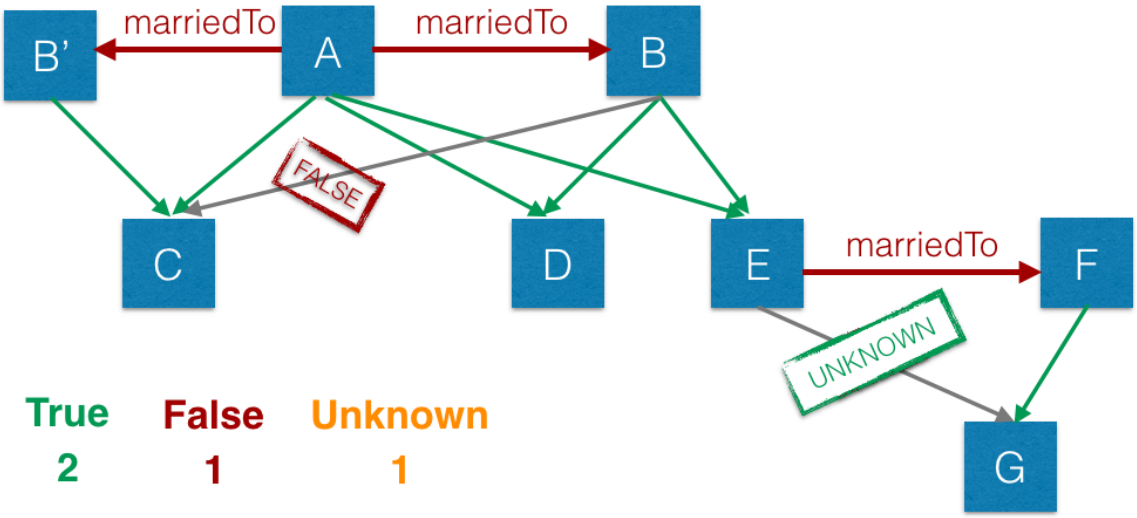


اگر بخواهیم معیار اطمینان استاندارد در این گراف را بررسی کنیم، مشاهده می‌شود که این قانون ۲بار در این گراف صدق کرده ا ست در حالی که می‌توان ست ۴ بار اتفاق بیافتد. در شکل + دفعاتی که این قانون باید اتفاق می‌افتاده اســت و اتفاق نیافتاده اســت نمایش داده شــده و به عنوان نمونه‌ی منفی در نظر گرفته شده ا ست. با این او صاف معیار اطمینان ا ستاندارد قانون R در گراف معرفی شده برابر ۰,۵ می‌شود.

$$\text{Standard Confidence} = \frac{\sup(B => r(x,y))}{\#(x,y):B} = 2 / 4 = 0.5$$



اما همانطور که توضیح دادیم عدم وجود حقایق در پایگاه‌های دانش فرض جهان باز دلیل بر غلط بودن این حقیقت‌ها نیست. برای مثال فرض کنید که در گراف معرفی شده یک گره‌ی B’ داشته باشیم که رابطه‌ی ازدواج داشته باشد با گره‌ی A و رابطه‌ی فرزندی داشته باشد 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 +).



پس معیار اطمینان استاندارد برای پایگاه‌های دانش جهان باز معیار مناسبی نیست و نیاز به تعریف معیار بهتری داریم. در [AMIE] معیار اطمینان با فرض نیمه‌کامل[1] (PCA-Confidence) را معرفی کرده است.

۴-۱-۴- معیار اطمینان با فرض نیمه‌کامل

در این معیار سعی شده نمونه‌هایی که در پایگاه‌دانش وجود ندارند و در معیار اطمینان استاندارد آن‌ها را نمونه‌ی غلط در نظر می‌گرفتیم را در اینجا بیشتر بررسی کنیم و با احتمال بهتری غلط بودن یا نبودن آن را مشخص کنیم.

در این معیار اگر حقیقت r(x,y) در پایگاه دانش موجود نبود، بررسی می‌شود که 'yی 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.

$$conf_{pca}(\overrightarrow{B} \Rightarrow r(x,y)) := \frac{supp(\overrightarrow{B} \Rightarrow r(x,y))}{\#(x,y) : \exists z_1, ..., z_m, y' : \overrightarrow{B} \wedge r(x,y')} \quad (1)$$

به مثال جدول + برگریدم، 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 اینکه livesIn (Bob, Zurich) داریم و در مورد محل تولد آن اطلاعی نداریم دلیل نمی‌شود 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 + 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

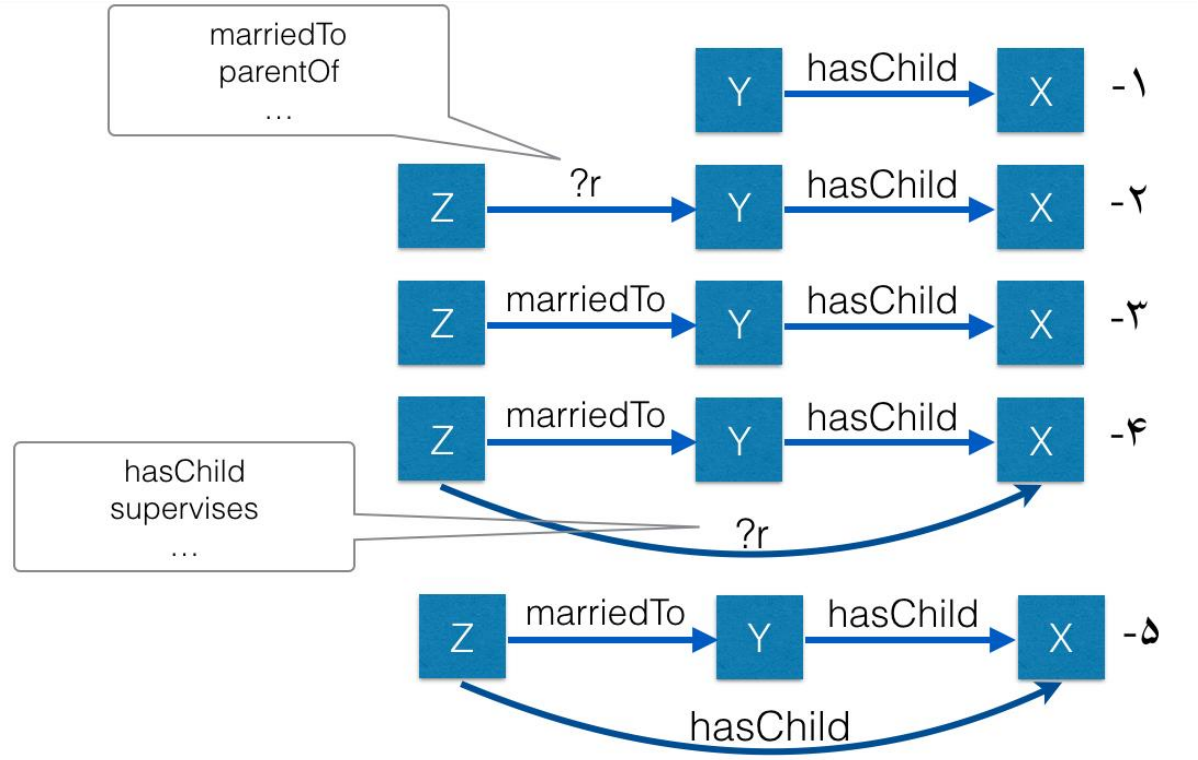
[1] Partial Completeness Assumption

شـــویم که رابطه‌ی فرزندی بین B و C برقرار نیســـت 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 ناشـــناخته در نظر می‌گیریم و از آن در فرومول PCA-confidence اســـتفاده نمی‌کنیم. پس PCA-Confidence در این مثال برابر با X است.

۲-۴- الگوریتم‌ها

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

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



به ازای هر حقیقت این روند یک بار تکرار می‌شود:

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

2 - 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.

3 - همه‌ی روابطی که در مرحله‌ی ۲ کاندید شده بودن 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.

4 - در این قسمت همه‌ی روابطی که 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.

5 - 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 به معیارها مقدار قابل قبولی داشته باشند قانون تولید شده را ذخیره می‌کنیم و در غیر این صورت از آن رد می‌شویم.

6 - 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 پایگاه‌دانش 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 پایگاه‌دانش 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 که 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 که یک نمونه از این قوانین به صورت زیر است. در این قانون داریم که اگر a یک موسسه‌ی آموزشی باشد که در مکان f قرار دارد و داشته باشیم که f در مکان b واقع شده است، می‌توانیم نتیجه

[1] Pruning

[2] rule body

بگیریم که موسسه‌ی آموزشی a نیز در مکان b قرار دارد. علی‌رقم 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.

۱-۲-۴- 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 آمورش مدل‌ها کمک بسیاری می‌تواند بکند زیرا ماهیت این قانون به این صورت است که روابط یکسان (روابطی که به دو صورت در پایگاه‌دانش 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.

۳-۴- 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.

۱-۳-۴- 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.

۲-۳-۴- قانون ۲-hope translation

این قانون، همان قانون تعدی است، به این صورت که اگر x و y رابطه‌ی r1 را باهم داشته باشند و همچنین y و z نیز رابطه‌ی r2 را داشته باشند، می‌توانیم نتیجه بگیریم که x و z نیز با یکدیگر در ارتباطند.

$$r1(x, y), r2(y, z) \Rightarrow r'(x, z)$$

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.

۴-۳-۴- 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 + قوانین معرفی شده به اختصار آمده‌اند.

name	Target rule		Body rule
R-subsumption	$r'(x, y)$	=>	$r(x, y)$
R-equivalence	$r'(x, y)$	<=>	$r(x, y)$
2-hope translation	$r'(x, z)$	=>	$r_1(x, y), r_2(y, z)$
Triangle alignment	$r'(x, y)$	=>	$r(x, z), r(y, z)$

Specific R-sub	r'(x, y)	=>	$r_1(x, y), r_2(x, v)$

۴-۴- جمع‌بندی

متن

فصل ۵: روش پیشنهادی

متن

۵-۱- چالش‌ها

متن

۵-۲- 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 RESCAL 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 پایگاه‌دانش Freebase15k انجام شده است که شامل حدود ۱۵هزار موجودیت، ۱۳۰۰ رابطه و در مجموع ۶۰۰هزار حقیقت است. تقسیم‌بندی داده‌های آموزش، ارزیابی و آزمون به شکلی که در + توضیح داده شد انجام شده است.

۵-۲-۱- روش رسکال:

تعداد پارامترها: در این روش برای هر موجودیت یک بردار dبعدی 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 آزمایشات تعداد ابعادی که بهترین نتیجه را بدست می‌دهد d=250 است. اگر ابعاد مساله را ۲۵۰ در نظر بگیر باید به تعداد ۸۸میلیون پارامتر آموزش داده شود (ned + nrd2 = 15k*250 + 1.3k*250*250 = 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.

۵-۲-۲- روش NTN:

تعداد پارامترها: در این روش نیز برای هر موجودیت یک بردار dبعدی در نظر گرفته شده و برای هر رابطه یک ماتریس Bk و دو بردار Ak و wk در نظر گرفته می‌شود که در مجموع برای هر رابطه d3 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 ($n_e d + n_r d^2$ 163,250k = 15k*50 + 1.3k*50*50*50).

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 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 نمونه‌ی آموزش ندیده است و نتایج خیلی پرتی بدست می‌دهد ولی برای ۴۲,۱٪ حقیقت که روابط بین آن‌ها به خوبی آموزش دیده شده است و پاسخ‌های درست در ۱۰ پاسخ اول بوده است.

۳-۲-۵- روش Structured Embeddi ng

تعداد پارامتر: در این روش برای هر موجودیت یک بردار dبعدی 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. ($n_e d + 2n_r d^2 = 15k*50 + 2*1.3k*50*50 = 7.250k$)

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.

۴-۲-۵- 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 dبعدی 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 نیزا داریم نیز dبعد 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 پایگاه‌دانش را با حدود ۸۰۰هزار پارامتر حل می‌کند. $(n_e d + n_r d = 15k*50 + 1.3k*50 = 815k)$.

کیفیت خروجی: این روش علی‌رقم کاهش چشم‌گیری که در تعداد پارامترها داشت و مقیاس‌پذیری بسیار بالایی که ایجاد کرده است، 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 پرت بسیار کمتر شده است و جواب سوال‌ها به جواب‌های منطقی نزدیک‌تر شده است.

۵-۲-۵- روش TransH:

تعداد پارامتر: همانطور که در بخش – 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 $O(n_ek + 2n_rk)$ هستند. تعداد پارامتر این روش روی پایگاه دانش freebase15k حدود 880 خواهد بود ($n_ed + 2n_rd = 15k*50 + 2*1.3k*50 = 880k$). 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 و چند به چند و چند به یک و بازگشتی توانسته است که دقت بهتری نسبت به روش‌های نام برده شده‌ی قبلی به دست آورد و بدون افزایش غیر منطقی تعداد پارامترها نسبت به روش TransE به مقدار Hit@10 برابر با ٪۶۴,۴ و میانگین رتبه‌ی ۸۷ برسد.

۳-۵- نقاط قوت و ضعف روش‌های موجود

در این بخش به بررسی نقاط قوت و نقاط ضعف روش‌های موجود می‌پردازیم. در بخش + 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 شدند 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 + مشاهده می‌کنیم.

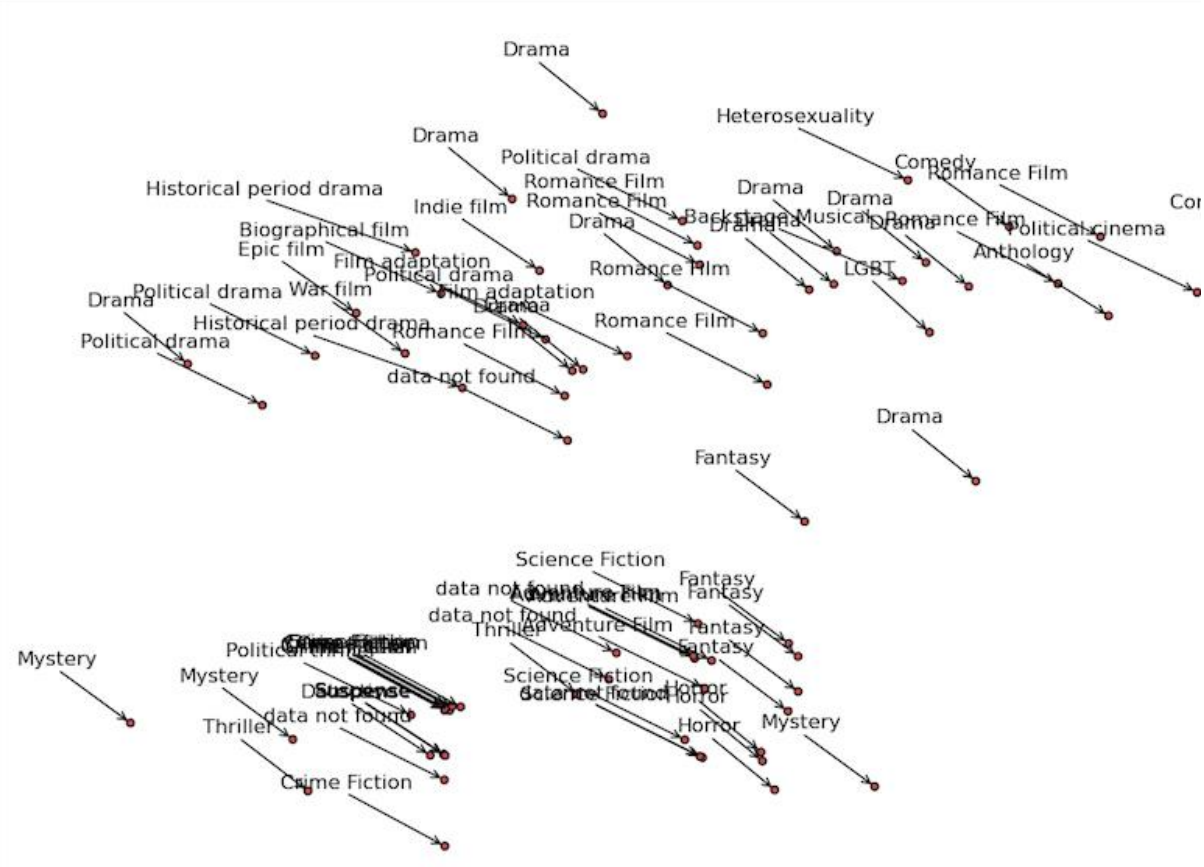
Lil Wayne born in?! (Lil_Wayne, born_in, ?)	New Orleans, Atlanta, Austin, St. Louis, Toronto, New York City, Wellington, Dallas, Puerto Rico
WALL-E has the genre?! (WALL-E, has_genre, ?)	Animations, Computer Animation, Comedy film, Adventure film, Science Fiction, Fantasy, Stop motion, Satire, Drama

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 پرت داخل آن‌ها نیست.

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

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

برای مطمئن شدن از این فرض چند جزیره را در شکل‌های + و + و + بزرگ‌نمایی کرده‌ایم.



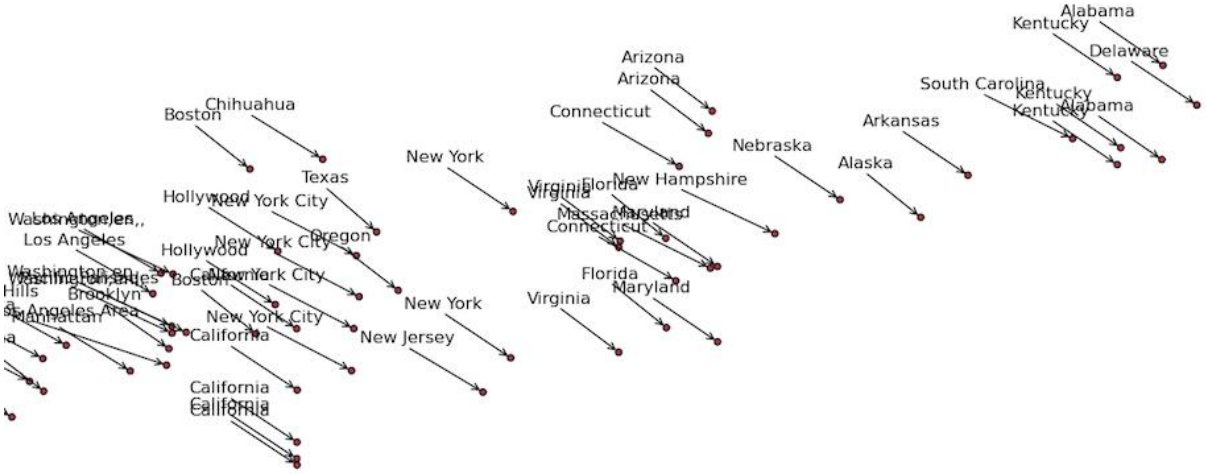
در شکل + مشاهده می‌شود که عمده‌ی موجودیت‌هایی که در این جزیره قرار گرفته‌اند ژانر فیلم‌های سینمایی هستند، مانند درام[1]، علمی تخیلی[2]، طنز[3] و ...

[1] Drama

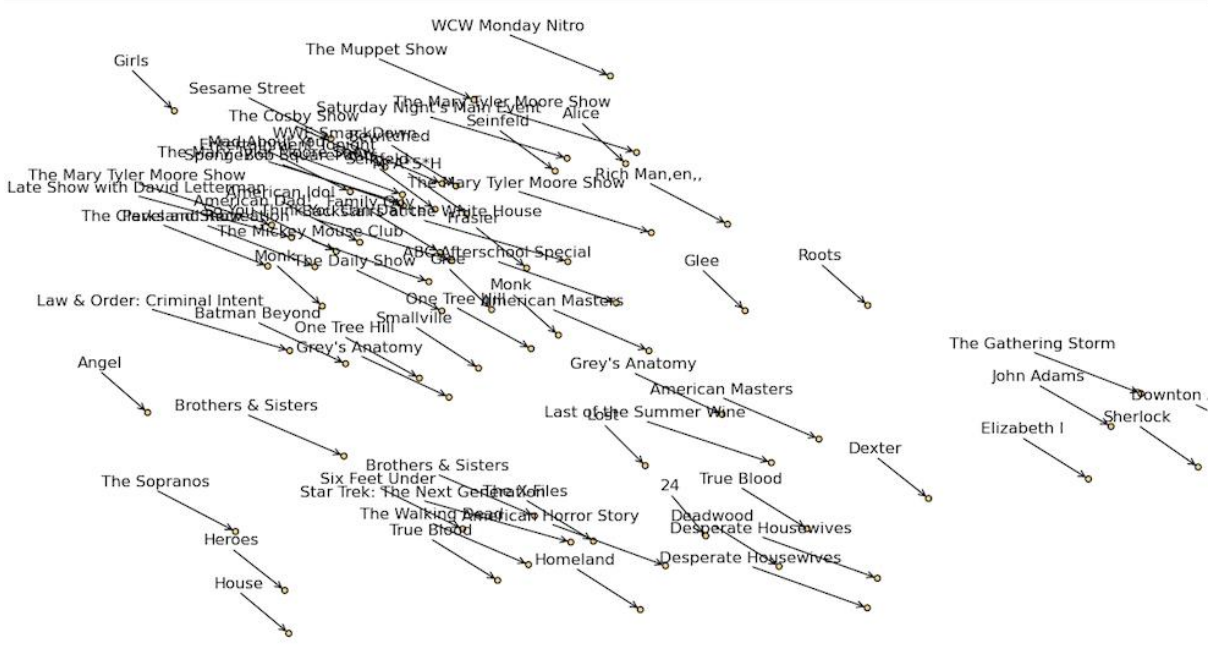
[2] Science fiction

[3] Comedy

Table 1



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



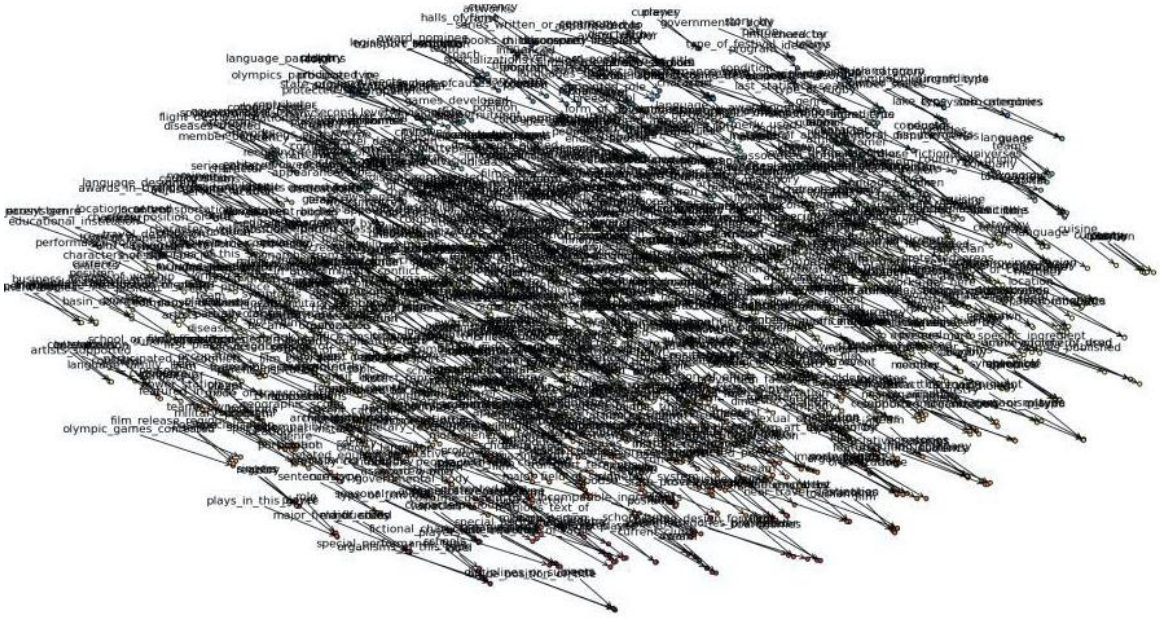
در شکل + مشاهده می‌شود که نام سریال‌های تلوزیونی یک جزیره را تشکیل داده‌اند.

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.

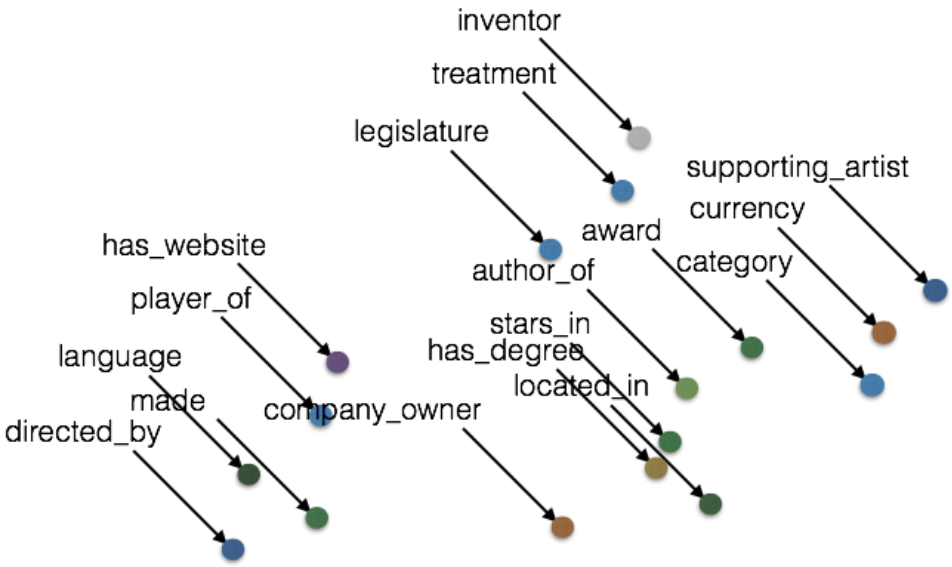
می‌دانیم که رابطه‌ها هم مانند موجودیت‌ها می‌توانند ارتباطات و همبستگی‌های خود را داشته باشند. برای مثال رابطه‌ی "پدر بودن" نزدیک‌تر است به رابطه‌ی "فرزندی" نسبت به رابطه‌ی "نویسنده‌ی کتاب" بودن و انتظار داریم بردارها و ماتریس‌های آموزش دیده شده برای رابطه‌ها این همبستگی‌ها را درک کرده باشد و در فاصله‌ی 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 + بردارهای همه رابطه‌های استفاده شده در Freebase15k را در دو بعد به نمایش در آورده‌ایم. همانطور که در شکل مشخص است، برای رابطه‌ها اتفاق جزیره‌ای شدن که در بازنمایی موجودیت‌ها اتفاق افتاده بود، به وجود نیامده است و بردارها روی صفحه بخش[1] است. به نظر می‌رسد که در بردارهای آموزش دیده برای رابطه‌ها این همبستگی و ارتباط وجود ندارد.

[1] Scatterd



برای بررسی بیشتر این موضوع چند قسمت از بازنمایی این بردارها را در شکل + و + 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 قوی عمل کنند و رابطه‌ها را درک کنند. در فصل بعد روی این مساله تمرکز می‌کنیم و سعی می‌کنیم به مساله را در آموزش مدل‌ها دخیل کنیم و همبستگی بین رابطه‌ها را نیز به مدل بفهمانیم و انتظار داریم نتیجه‌ی بهتری در معیارهای کیفیت بگیریم.



۴-۵- بررسی عمل‌کرد روش‌ها در یک نگاه

در این قسمت روش‌های موجود را ...

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

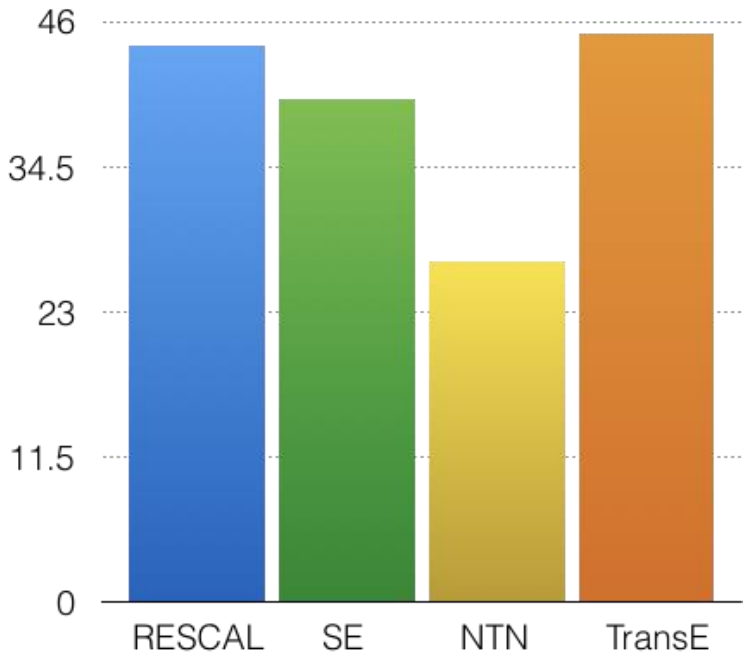
Method	#Params	On FB15K
RESCAL	$O(n_e d + n_r d^2)$	88M (d=250)
MLP (NTN)	$O(n_e d + n_r d^3)$	165M (d=50)
SE	$O(n_e d + 2n_r d^2)$	8M (d=50)
TransE	$O(n_e d + n_r d)$	0.8M (d=50)

در جدول + میزان Hit@10 و میانگین رتبه‌ی هر یک از روش‌های معرفی شده نمایش داده شده است که در این جدول مشاهده می‌شود که روش NTN در hit@10 نتیجه‌ی مطلوبی نگرفته است که 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 RESCAL هم hit@10 خوب بوده است اما رتبه‌ی میانگین مقدار بسیار بالایی نسبت به روش‌های دیگر داشته که نشان از این دارد که علی‌رقم اینکه بسیاری از روابط را درست فهمیده است و نتایج خوبی در 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 پرتی گرفته‌اند که باعث شده میانگین رتبه‌ی این روش چنین افزایشی داشته باشد.

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



۵-۵- 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 این بخش به چگونگی انجام این بهبود می‌پردازیم.

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

اما در روش‌های TransE و TransH این قوانین را مستقیما وارد فرایند آموزش کردیم. به این صورت که پایگاه‌دانش را گسترش ندادیم و با همان حدود ۵۰۰هزار 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 پایگاه‌دانش باعث شد که هم مساله‌ی پیش‌پرداز برای اعمال قوانین روی پایگاه‌دانش 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 + بردار رابطه و بردار موجودیت دوم بود که برای حقیقت r(e1,e2) این رابطه را به صورت زیر می‌نوشتیم.

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

۵-۵-۱- قانون R-subsumtion

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.

۵-۵-۲- قانون R-equi val ence

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.

۵-۵-۳- 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 r1 و r2 ایجاد می‌کنند برابر باشد با انتقال بردار r' یعنی:

$$r1 + r2 = r'$$

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:

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

۴-۵-۵- 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.

$$r_1(e_1, e_3) , r_2(e_2, e_3) \Rightarrow r'(e_1, e_2)$$

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

۵-۵-۵- 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 $r1(x, y)$ و $r2(x, V)$ 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 ۶: ارزیابی

۶-۱- 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 + به تشریح آزمایش‌ها و نتایج بدست‌آمده می‌پردازیم، در بخش + به بررسی نتایج و مقایسه آن‌ها اختصاص می‌دهیم و در بخش + به کارهایی که در آینده می‌توان برای بهبود بیشتر نتایج انجام داد را معرفی می‌کنیم.

۶-۲- 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 فصل + روی قوانین انجمنی داشتیم استفاده می‌کنیم. و بررسی می‌کنیم که هر یک از این دسته قوانین چه‌مقدار توسط مدل درک شده‌اند و خوب جواب می‌دهند.

برای انجام آزمایش‌ها باید پایگاه‌دانش و قوانین انجمنی که می‌خواهیم عملیات آموزش را روی آن‌ها انجام دهیم را مشخص کنیم. پایگاه‌دانش استفاده شده همان پایگاه‌دانش Freebase15k است که در

بخش + معرفی شد (حدود ۶۰۰هزار حقیقت، ۱۵هزار موجودیت و ۱۳۰۰ رابطه) است. در ادامه قوانین انجمنی را روی این پایگاه‌دانش اجرا می‌کنیم و قوانین بدست آمده را دسته‌بندی می‌کنیم.

همانطور که در بخش + اشاره شد، پس از اجرای قوانین انجمنی روی پایگاه‌دانش 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 + قانون شد. این قوانین را بر اساس تقسیم‌بندی‌های بخش + دسته‌بندی کردیم که این دسته‌بندی در جدول + مشخص شده است.

Rules		
name	**rule**	**#rule**
R-Subsumption	r(x, y) => r'(x, y)	1127
R-Equivalence	r(x, y) <=> r'(x, y)	782
2-hope	r1(x, y), r2(y, z) => r'(x, z)	810
Triangle alignment	r(x, z), r(y, z) => r'(x, y)	695
SR-Subsumption	r_1(x, y), r_2(x, v) => r'(x, y)	779
all rules		4193

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 + 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 + معیار hit@10 و در نمودار + معیار میانگین رتبه برای روش‌های معرفی شده در +، قبل و بعد از اضافه کردن قوانین استخراج شده در دسته‌ی R-Subsumption را مشاهده می‌کنیم.

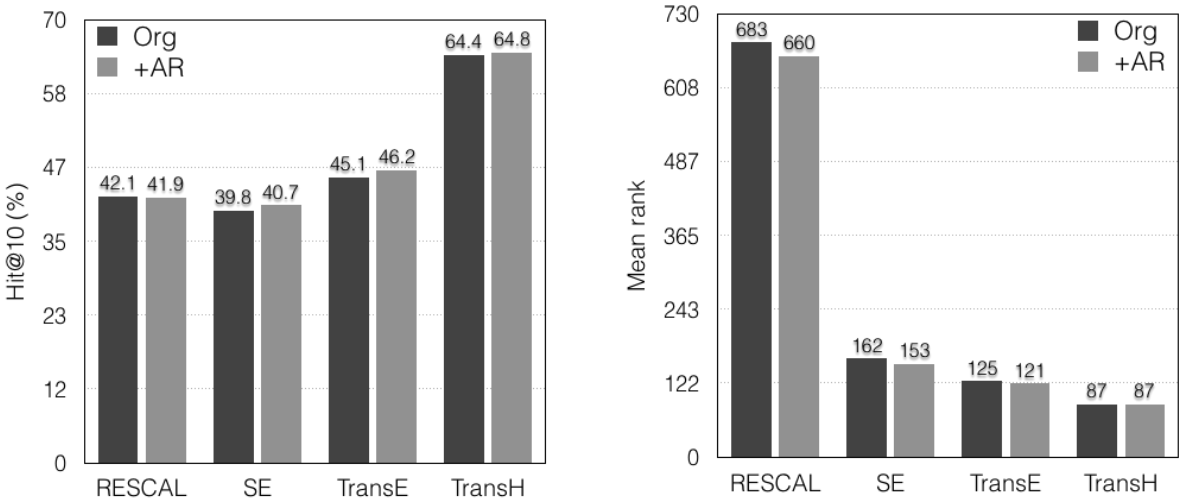


Figure 1 - R-sub

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.

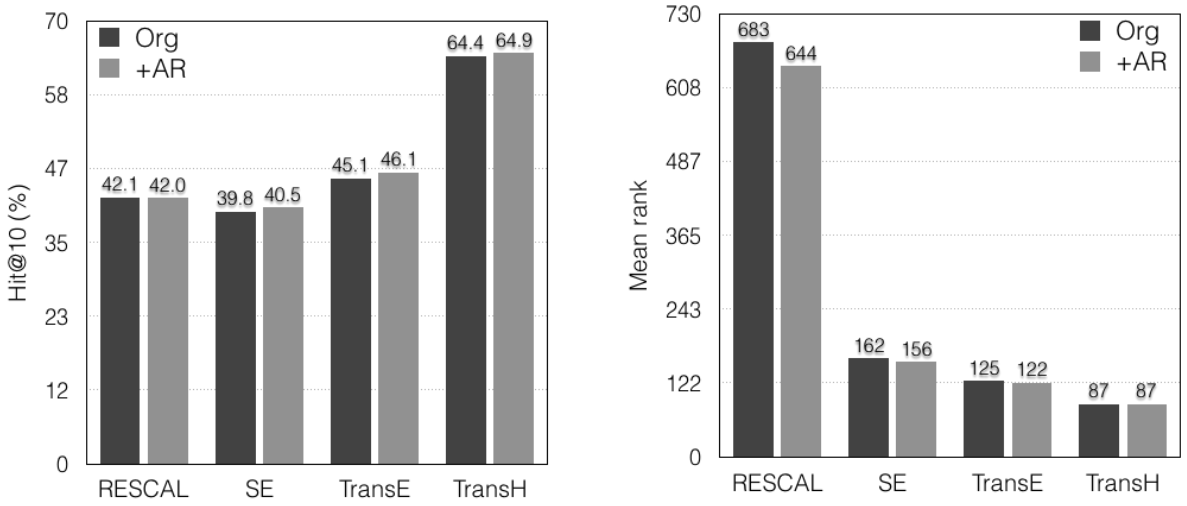
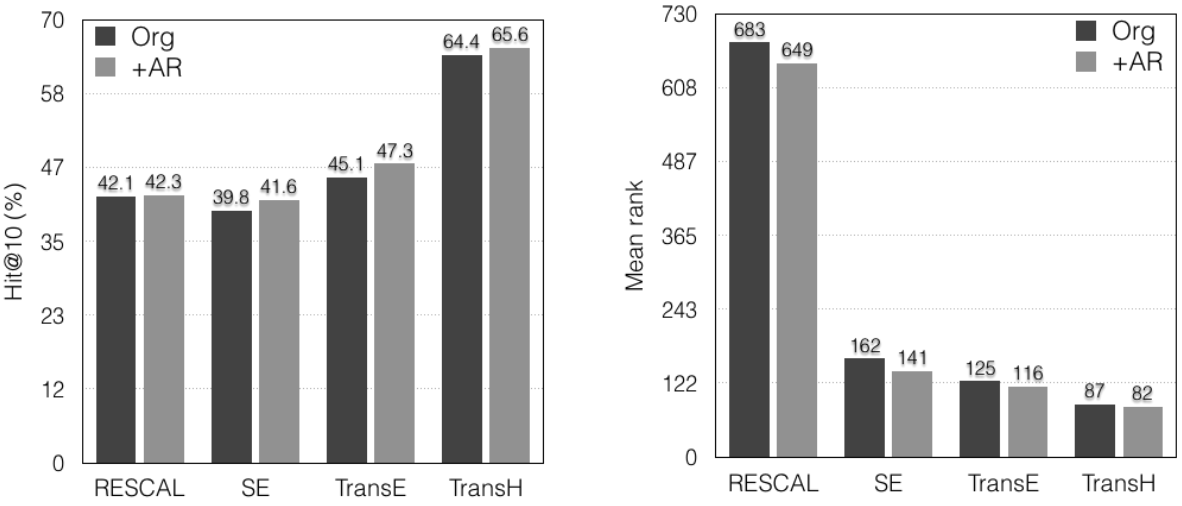


Figure 2- SR-sub

در نمودارهای + و + 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.

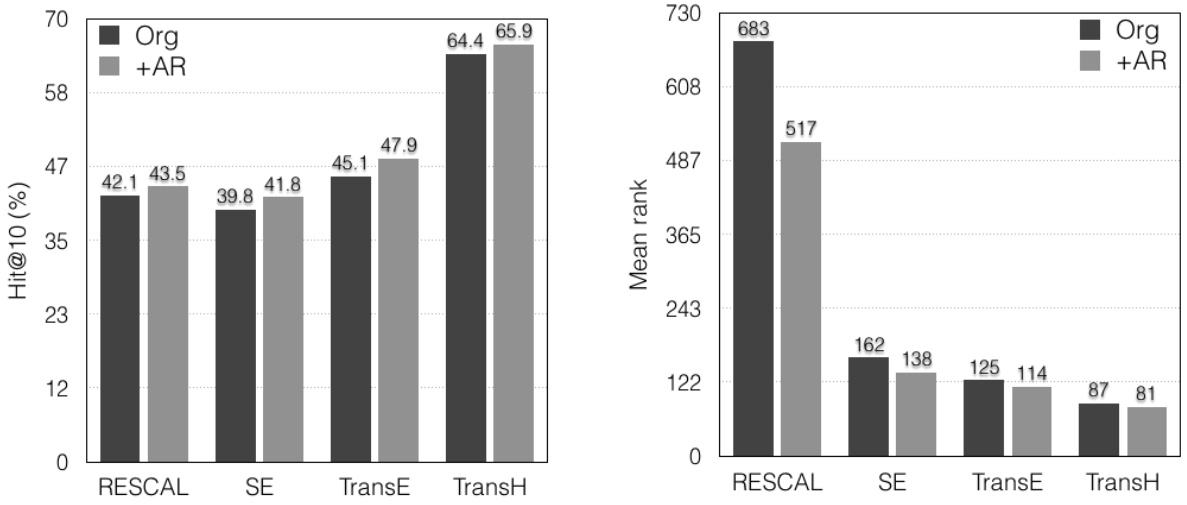


R-Equ-Figure 3

در نمودارهای + و + مشاهده می‌شود اضافه کردن قوانین R-equvalence تاثیر چندانی روی نتایج RESCAL 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 + 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 پایگاه‌دانش 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 RESCAL نیز به این دلیل است که این روش همانطور که در توضیح آن در بخش + 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.



در نمودارهای + و + با اضافه کردن قوانین 2-hope translation نیز در هر دو معیار بهبود محسوسی دیده می‌شود. مشاهده می‌شود که در روش RESCAL 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.

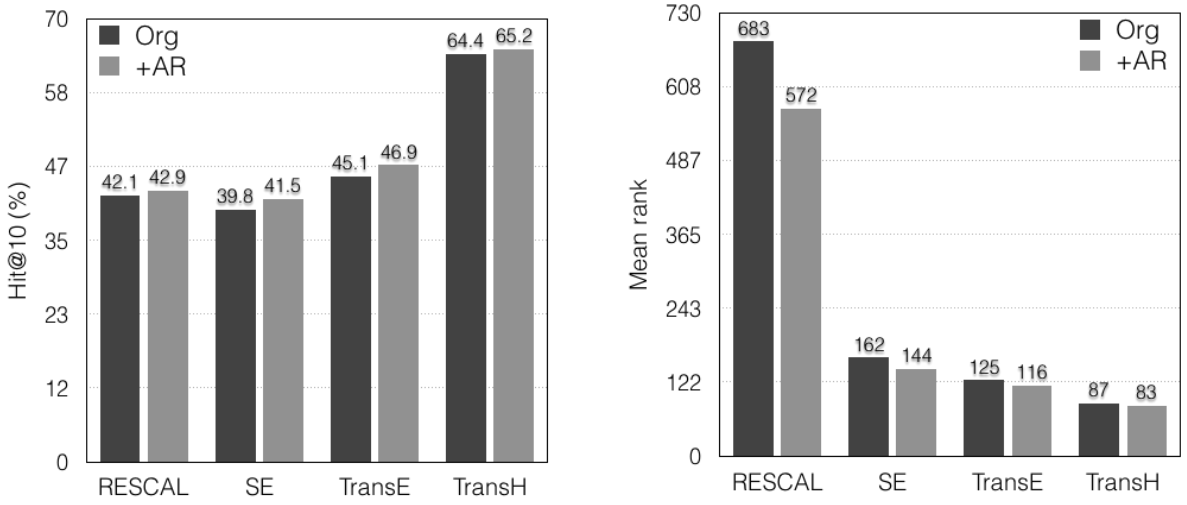
می‌توانیم نتیجه بگیریم که هیچ یک از این مدل‌ها قادر به شناسایی قوانین غیر مستقیم پیچیده نبوده است. در آزمایش‌های قبلی دیدیم که قوانینی غیر مستقیم که با استنتاج از روی یک رابطه ساخته شده بودند (در قسمت بدنه‌ی قانون فقط یک حقیقت وجود داشت) را مدل‌ها بهتر درک کرده بودن و اضافه کردن قوانین مربوط به این آزمایش‌ها 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.

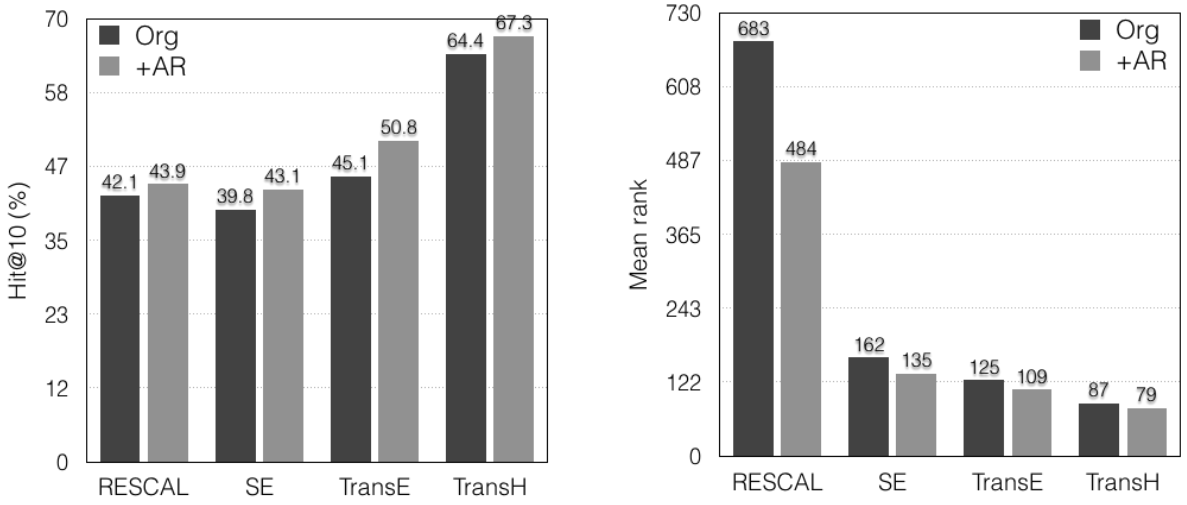
به طور خاص یکی از دلایل بالا بودن معیار میانگین رتبه این است که برای برخی روابط حقیقت‌های کمی در پایگاه‌دانش موجود است و مدل‌ها از روی این تعداد کم رابطه قادر به شناسایی الگو برای پاسگویی به سوالات در مورد این حقیقت‌ها و روابط نیستند. همین امر باعث می‌شود که پاسخ‌هایی که به سوالات در مورد این حقیقت‌ها می‌دهند به صورت تصادفی باشد و رتبه‌های پرتی بگیرد. این رتبه‌های پرت باعث افزایش زیادی در معیار میانگین رتبه می‌شود، در روش RESCAL 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 پایگاه‌دانش 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 پایگاه‌دانش 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.



در نمودارهای + و + 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 + معرفی کردیم را روی مدل‌های معرفی شده را بررسی می‌کنیم. نتایج بدست آمده در نمودارهای + و + نشان داده شده است.



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.

۳-۶- جمع‌بندی

در جدول + و + 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 را در کنار هم قرار داده‌ایم.

Hit@10 on Freebase 15K (%)							
Method	orginal	R-Subsumption	R-Equivalence	2-hope	Triangle alignment	SR-Subsumption	all rules (~improve)
RESCAL	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%)**

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

RESCAL	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 پایگاه‌دانش 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 RESCAL شده است که در بخش + دلیل عدم بهبود برای روش RESCAL 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 RESCAL 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 RESCAL 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 ۷: نتیجه‌گیری

۷-۱- نتیجه‌گیری

متن

۷-۲- 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@۱ 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 ۸: مراجع

[1] N. Feamster, J. Rexford, and E. Zegura, "The Road to SDN : An Intellectual History of Programmable Networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 2, pp. 87–98, 2014.

[2] D. Kreutz, E. Verıssimo, and S. Azodolmolky, "Software-Defined Networking: A Comprehensive Survey," *Proc. IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[3] R. Masoudi and A. Ghaffari, "Software defined networks: A survey," *J. Netw. Comput. Appl.*, vol. 67, no. C, pp. 1–25, 2016.

[4] "Software-Defined Networking : The New Norm for Networks," 2012. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/white-papers/wp-sdn-newnorm.pdf.

[5] H. Hata, "A Study of Requirements for SDN Switch Platform," in *Intelligent Signal Processing and Communications Systems (ISPACS), 2013 International Symposium on*, 2013, pp. 79–84.

[6] Y. R. Qu, S. Zhou, and V. K. Prasanna, "A Decomposition-based Approach for Scalable Many-field Packet Classification on Multi-core Processors *," *Int. J. Parallel Program.*, vol. 43, no. 6, pp. 965–987, 2015.

[7] Y. R. Qu and V. K. Prasanna, "Power-efficient range-match-based packet classification on FPGA," in *Field Programmable Logic and Applications (FPL)*, 2015.

[8] S. Banerjee and K. Kannan, "Tag-In-Tag : Efficient Flow Table Management in SDN Switches," in *Network and Service Management (CNSM), 2014 10th International Conference on*, 2014, pp. 109–117.

[9] H. Farhadi and A. Nakao, "Rethinking Flow Classification in SDN," in *Cloud Engineering (IC2E), 2014 IEEE International Conference on*, 2014, pp. 598–603.

[10] K. G. Pérez, X. Yang, and S. Sezer, "A Configurable Packet Classification Architecture for Software-Defined Networking," in *System-on-Chip Conference (SOCC)*, 2014, pp. 353–358.

[11] P. T. Congdon, P. Mohapatra, M. Farrens, and V. Akella, "Simultaneously reducing latency and power consumption in openflow switches," *IEEE/ACM Trans. Netw.*, vol. 22, no. 3, pp. 1007–1020, 2014.

[12] Y. R. Qu, H. H. Zhang, S. Zhou, and V. K. Prasanna, "Optimizing many-field packet classification on FPGA, multi-core general purpose processor, and GPU," in *Architectures for Networking and Communications Systems (ANCS)*, 2015, no. 3, pp. 87–98.

[13] M. Varvello, R. Laufer, F. Zhang, and T. V Lakshman, "Multi-Layer Packet Classification with Graphics Processing Units Categories and Subject Descriptors," in *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, 2014, pp. 109–120.

[14] C. Hsieh and N. Weng, "Many-Field Packet Classification for Software-Defined Networking Switches," in *Proceedings of the 2016 Symposium on Architectures for Networking and Communications Systems*, 2016, pp. 13–24.

[15] H. Lim, N. Lee, G. Jin, J. Lee, Y. Choi, and C. Yim, "Boundary cutting for packet classification," *IEEE/ACM Trans. Netw.*, vol. 22, no. 2, pp. 443–456, 2014.

[16] S. K. Oumya and C. H. S. E. M, “Selective Boundary Cutting For Packet Classification,” *Int. J. Sci. Eng. Technol. Res.*, vol. 4, no. 34, pp. 6786–6790, 2015.

[17] P. Gupta and N. Mckeown, “Packet Classification using Hierarchical Intelligent Cuttings,” in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, 2003, pp. 213–224.

[18] S. Singh, F. Baboescu, G. Varghese, and J. Wang, “Packet Classification Using Multidimensional Cutting,” in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, 2003, pp. 213–224.

[19] D. E. Taylor and J. S. Turner, “ClassBench: A packet classification benchmark,” *IEEE/ACM Trans. Netw.*, vol. 15, no. 3, pp. 499–511, 2007.

[20] W. Xia, Y. Wen, S. Member, C. H. Foh, and S. Member, “A Survey on Software-Defined Networking,” *IEEE Commun. Surv. Tutorials*, vol. 17, no. 1, pp. 27–51, 2014.

[21] “Traditional vs Software Defined Networking.” [Online]. Available: www.ipknowledge.net/wp-content/uploads/2014/12/SDN.pdf.

[22] Y. Gong, W. Huang, W. Wang, and Y. Lei, “A survey on software defined networking and its applications,” *Front. Comput. Sci.*, vol. 9, no. 6, pp. 827–845, 2015.

[23] V. Shamugam, I. Murray, L. J. A, and A. S. Sidhu, “Software Defined Networking challenges and future direction : A case study of implementing SDN features on OpenStack private cloud,” in *IOP Conference Series: Materials Science and Engineering*, 2016, vol. 121, no. 1, pp. 1–8.

[24] M. Karakus and A. Durresi, “A survey: Control plane scalability issues and approaches in Software-Defined Networking (SDN),” *Comput. Networks*, vol. 112, pp. 279–293, 2017.

[25] W. Li, W. Meng, and F. L. Kwok, “A survey on OpenFlow-based Software Defined Networks: Security challenges and countermeasures,” *J. Netw. Comput. Appl.*, vol. 68, no. C, pp. 126–139, 2016.

[26] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling Innovation in Campus Networks,” *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, p. 69, 2008.

[27] A. Doria, J. H. Salim, W. Wang, and L. Dong, “Forwarding and Control Element Separation (ForCES) Protocol Specification,” *Internet Engineering Task Force*, 2010. [Online]. Available: https://tools.ietf.org/html/rfc5810.

[28] B. Pfaff and B. Davie, “The Open vSwitch Database Management Protocol,” *Internet Engineering Task Force*, 2013. [Online]. Available: https://tools.ietf.org/html/rfc7047.

[29] H. Song, “Protocol-oblivious forwarding: unleash the power of SDN through a future-proof forwarding plane,” in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*, 2013, pp. 127–132.

[30] M. Smith, M. Dvorkin, and P. Garg, “OpFlex Control Protocol,” *Internet Engineering Task Force*, 2014. [Online]. Available: https://tools.ietf.org/html/draft-smith-opflex-00.

[31] G. Bianchi, M. Bonola, A. Capone, and C. Cascone, “OpenState: Programming Platform-independent Stateful OpenFlow Applications Inside the Switch,” *Sigcomm Ccr*, vol. 44, no. 2, pp. 44–51, 2014.

[32] A. Lara, A. Kolasani, and B. Ramamurthy, “Network Innovation using OpenFlow : A Survey,” *IEEE Commun. Surv. Tutorials*, vol. 16, no. 1, pp. 493–512, 2013.

[33] C. Hao, Chang, and Y.-D. Lin, “OpenFlow Version Roadmap,” 2015. [Online]. Available:

http://speed.cis.nctu.edu.tw/~ydlin/miscpub/indep_frank.pdf.

[34] P. Gupta and N. Mckeown, “Algorithms for Packet Classification,” *IEEE Netw.*, vol. 15, no. 2, pp. 24–32, 2002.

[35] J. Sanders and E. Kandrot, *CUDA by Example: An Introduction to General-Purpose GPU Programming*, 1st ed. Addison-Wesley Professional, 2010.

[36] M. Arora, “The Architecture and Evolution of CPU-GPU Systems for General Purpose Computing.” University of California, San Diago, 2012.

[37] S. Singh, S. Singh, V. Banga, and C. Durlabh, “CUDA for GPGPU Applications – A Survey,” *Natl. Conf. Contem-poraryTechniques Technol. Electron. Eng.*, pp. 1–4, 2013.

[38] M. Mukerjee, D. Naylor, and B. Vavala, “Packet Processing on the GPU.” [Online]. Available: http://www.cs.cmu.edu/~bvavala/misc/project740/15-740_Project_files/Report.pdf.

[39] S. Han, K. Jang, K. Park, and S. Moon, “PacketShader : a GPU-Accelerated Software Router,” in *Proceedings of the ACM SIGCOMM 2010 conference*, 2010, pp. 195–206.

[40] G. Vasiliadis, S. Antonatos, M. Polychronakis, and P. Evangelos, “Gnort : High Performance Network Intrusion Detection Using Graphics Processors,” in *Proceedings of the 11th international symposium on Recent Advances in Intrusion Detection*, 2008, pp. 116–134.

[41] Y. Zhu and Y. Chen, “Hermes : An Integrated CPU / GPU Microarchitecture for IP Routing,” in *Design Automation Conference (DAC)*, 2011, pp. 1044–1049.

[42] R. Garg and S. Mittal, “Optimization by genetic algorithm,” *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 4, no. 4, pp. 587–589, 2014.

[43] D. E. Taylor and J. S. Turner, “Scalable packet classification using distributed crossproducing of field labels,” *Proc. IEEE 24th Annu. Jt. Conf. IEEE Comput. Commun. Soc.*, vol. 1, pp. 1–12, 2005.

[44] D. Yuan, X. Yang, X. Shi, B. Tang, and Y. Liu, “Multi-protocol query structure for SDN switch based on parallel bloom filter,” in *International Conference on ICT Convergence*, 2014, pp. 206–211.

[45] ب. جعفریان, "د سته‌بندی ب سته‌ها در شبکه‌های سرعت بالا," پایان‌نامه کار شنا سی ار شد, دان شگاه تهران, ۱۳۹۱.

[46] C. Thomas H, C. E. Leiserson, R. Ronald L, and S. Chiffoed, *Introduction to Algorithms*, 3rd ed. London, England: MIT Press, 1988.

[47] P. Gupta, “Algorithms for routing lookups and packet classification,” STANFORD, 2000.

[48] M. M. Buddhikot, S. Suri, and M. Waldvogel, “Space decomposition techniques for fast layer-4 switching,” in *Protocols for High-Speed Networks VI*, 2000, pp. 25–41.

[49] F. Baboescu, S. Singh, and G. Varghese, “Packet classification for core routers: is there an alternative to CAMs?,” in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*, 2003, vol. 1, pp. 53–63.

[50] B. Pfaff, J. Pettit, T. Koponen, E. Jackson, A. Zhou, J. Rajahalme, J. Gross, A. Wang, J. Stringer, P. Shelar, K. Amidon, A. Networks, and M. Casado, “The Design and Implementation of Open vSwitch,” in *Proceedings of the 12th USENIX Conference on Networked Systems Design and Implementation*, 2015, pp. 117–130.

فصل ۹: واژه‌نامه انگلیسی به فارسی

Abstraction	تجرید
Action	عمل
Appliance	دستگاه
Asynchronous	ناهمگام
Autonomous	خودمختار
Batch	دسته
Batch Delay	تاخیر دسته‌ای
Benchmarking	محک‌زنی
Biased Distribution	توزیع جهت‌دار
Bit Vector	بردار بیتی
Block	بلوک
Bus	گذرگاه
Centralized	متمرکز شده
Component	جزء

Constant Memory	حافظه ثابت
Control plane	سطح کنترل
Core	هسته
Crossover	تقاطع
Cut	برش
Cycle	چرخه
Data plane	سطح داده
Default	پیش‌فرض
Deterministic	قطعی
Diversity	تنوع
Driver	راه‌انداز
Duplicate	تکرار
Effective	موثر
Entry	مدخل
Exact Match	انطباق دقیق
False Positive	مثبت کاذب
Fitness	برازش
Flow	جریان
Full Match	تطابق کامل
Fully Efficient	به طور کامل کارآمد

Granularity	دانه‌بندی
Independence	استقلال
Inertia	لختی
Infrastructure	زیرساختار
Lane	خط
Line rate	نرخ خط
Logical	منطقی
Logically	منطقی، به طور منطقی
Mask	ماسک
Maximum	بیشینه
Merge	ادغام
Metadata	فراداده
Minimum	کمینه
Mutation	جهش
Mutually exclusive	دو به دو ناسازگار
Network Access Point	نقطه دسترسی شبکه
Network state	حالت شبکه
North-bound Interface	واسط شمالی
Offline	برون‌خط
Online	برخط

Out-of-order	خارج از نوبت
Performance	کارآیی
Pipeline	خط لوله
Platform	سکو
Pool	مخزن
Pre-filtering	پیش‌تصفیه
Prefix Match	انطباق پیشوندی
Preprocessing	پیش‌پردازش
Program Counter	شمارنده برنامه
Programmable	قابل برنامه‌ریزی
Range Match	انطباق بازه‌ای
Regular Expression	عبارت منظم
Replication Ratio	نسبت تکرار
Rule Programming	برنامه‌ریزی قانون
Scalability	مقیاس پذیری
Scheduler	زمان‌بند
Scheduling	زمان‌بندی
Scope	گستره
Segmentation	قطعه‌بندی
Slot	شکاف

Smoothing	هموارسازی
South-bound Interface	واسط جنوبی
Sparsity	خلوتی
Streaming Multiprocessor	چندپردازنده جریانی
String	رشته
Subrange	زیرمحدوده
Subset	زیرمجموعه
Switching	راه‌گزینی
Synthetic	مصنوعی
Target Rule	قانون هدف
Thread	ریسمان
Throughput	گذردهی
Tuple	چندتایی
Uniform	یکنواخت
Unifying	یک‌دست‌سازی
Zone	منطقه

فصل ۱۰: واژه‌نامه فارسی به انگلیسی

ادغام	Merge
استقلال	Independence
انطباق بازه‌ای	Range Match
انطباق پیشوندی	Prefix Match
انطباق دقیق	Exact Match
برازش	Fitness
برخط	Online
بردار بیتی	Bit Vector
بریدن	Cut
برنامه‌ریزی قانون	Rule Programming
برون‌خط	Offline
بلوک	Block
به طور کامل کارآمد	Fully Efficient
بیشینه	Maximum
پیش‌پردازش	Preprocessing

Pre-filtering	پیش‌تصفیه
Default	پیش‌فرض
Batch Delay	تاخیر دسته‌ای
Abstraction	تجرید
Full Match	تطابق کامل
Crossover	تقاطع
Duplicate	تکرار
Diversity	تنوع
Biased Distribution	توزیع جهت‌دار
Flow	جریان
Component	جزء
Mutation	جهش
Cycle	چرخه
Streaming Multiprocessor	چندپردازنده جریانی
Tuple	چندتایی
Constant Memory	حافظه ثابت
Network state	حالت شبکه
Out-of-order	خارج از نوبت
Lane	خط
Pipeline	خط لوله

Sparsity	خلوتی
Autonomous	خودمختار
Granularity	دانه‌بندی
Appliance	دستگاه
Batch	دسته
Mutually exclusive	دو به دو ناسازگار
Driver	راه‌انداز
Switching	راه‌گزینی
String	رشته
Thread	ریسمان
Scheduler	زمان‌بند
Scheduling	زمان‌بندی
Infrastructure	زیرساختار
Subset	زیرمجموعه
Subrange	زیرمحدوده
Data plane	سطح داده
Control plane	سطح کنترل
Platform	سکو
Slot	شکاف
Program Counter	شمارنده برنامه

Regular Expression	عبارت منظم
Action	عمل
Metadata	فراداده
Programmable	قابل برنامه‌ریزی
Target Rule	قانون هدف
Segmentation	قطعه‌بندی
Deterministic	قطعی
Performance	کارآیی
Minimum	کمینه
Throughput	گذردهی
Bus	گذرگاه
Scope	گستره
Inertia	لختی
Mask	ماسک
Centralized	متمرکز شده
False Positive	مثبت کاذب
Benchmarking	محک‌زنی
Pool	مخزن
Entry	مدخل
Synthetic	مصنوعی

Scalability	مقیاس‌پذیری
Zone	منطقه
Logical	منطقی
Logically	منطقی، به طور منطقی
Effective	موثر
Asynchronous	ناهمگام
Line rate	نرخ خط
Replication Ratio	نسبت تکرار
Network Access Point	نقطه دسترسی شبکه
South-bound Interface	واسط جنوبی
North-bound Interface	واسط شمالی
Core	هسته
Smoothing	هموارسازی
Unifying	یک‌دست‌سازی
Uniform	یکنواخت

Abstract:

In Software Defined Networking, a flow can be defined using arbitrary set of header fields of each layer. For example, In OpenFlow 1.3.1, total of 15 fields may be used to define a flow. Allowing to define flows with more fields, on the one hand, causes the length of flow table entries to be enlarged; On the other hand, it enables finer control on network traffic which in turn, causes the number of flow table entries to be increased. Consequently, design of a solution for packet classification in high performance environments has been turned out to be a big challenge. To address this problem in soft switches, researchers try to reduce the number of candidate rules that a tuple must match with them. In the MC-SBC algorithm, a two-level trie based structure is proposed for lookup tables in which a set of effective bits are detected through a statistical method, in order to partition the ruleset and construct the trie structure.

In this dissertation, genetic algorithm has been applied to find the effective bit sets for ruleset partitioning. The results show that the proposed method partitions the rulesets in more balanced subsets, such that, both the replication ratio and the maximum number of rules in a leaf node of the trie are decreased. Moreover, the results of running our prototype on a single NVIDIA GPU, shows that packet classification can be performed up to 10 percent faster for large rule sets.

Keywords: Sotware Defined Networking, Packet Classification, OpenFow Switch, Graphics Processing Unit, Genetic Algorithm

University of Tehran

College of Engineering

Faculty of Electrical and Computer Engineering

Packet Classification in Software Defined Networking

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

By:
Mohammad Reza Piroozi

Supervisor:
Dr. Nasser Yazdani