



دانشگاه تهران  
پردیس دانشکده های فنی  
دانشکده مهندسی کامپیوتر

## بهبود مدل های مبتنی بر ویژگی های پنهان پایگاه های دانش با استفاده از قوانین انجمنی

پایان نامه جهت دریافت درجه کارشناسی ارشد  
در رشته مهندسی کامپیوتر گرایش نرم افزار

نگارش:

مسعود هاشمیان

استاد راهنما:

دکتر ناصر یزدانی

مرداد ماه ۱۳۹۶



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ





دانشگاه تهران  
پردیس دانشکده های فنی  
دانشکده مهندسی کامپیوتر

## بهبود مدل های مبتنی بر ویژگی های پنهان پایگاه های دانش با استفاده از قوانین انجمنی

پایان نامه جهت دریافت درجه کارشناسی ارشد

در رشته مهندسی کامپیوتر گرایش نرم افزار

نگارش:

مسعود هاشمیان

استاد راهنما:

دکتر ناصر یزدانی

مرداد ماه ۱۳۹۶



دانشگاه تهران  
پردیس دانشکده‌های فنی  
دانشکده مهندسی برق و کامپیوتر

## گواهی دفاع از پایان‌نامه کارشناسی ارشد

هیأت داوران پایان‌نامه کارشناسی ارشد آقای / خانم ..... به شماره  
دانشجویی ..... در رشته ..... گرایش ..... را در تاریخ  
..... با عنوان .....

به عدد به حروف

--	--

با نمره نهایی

و درجه ارزیابی کرد.

--

ردیف	مشخصات هیأت داوران	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا مؤسسه	امضاء
۱	استاد راهنما استاد راهنمای دوم (حسب مورد)				
۲	استاد مشاور				
۳	استاد داور داخلی				
۴	استاد مدعو				
۵	نماینده کمیته تحصیلات 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 هاشمیان تأیید می‌کنم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب است و به دستاوردهای پژوهشی دیگران که در این نوشته از آنها استفاده شده است مطابق مقررات ارجاع گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم سطح یا بالاتر ارائه نشده است.

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

نام و نام خانوادگی دانشجو :

امضای دانشجو :





تقدیم به پدر، مادرم و همسر عزیزم



## چکیده

یادگیری ماشین برای رابطه‌های بین داده‌هایی حجیمی که روزانه توسط بازیابی اطلاعات، محاسبات زیست‌شناسی، پردازش زبان طبیعی و ... تولید می‌شوند مساله‌ی سختی تبدیل شده است. روش‌های زیادی برای کار با این داده‌ها معرفی شده است که پایگاه‌های دانش بزرگی را تحلیل و روابط موجود در آن‌ها را استخراج می‌کند. یک دسته از این روش‌ها روش‌های مبتنی بر ویژگی‌های پنهان است که مساله را وارد یک فضای برداری چند بعدی کرده و پس از آن سعی می‌کنند با تعداد پارامتر کم مساله را حل کنند. قدرت این روش‌ها در سادگی آموزش مدل، تعداد پارامتر کمی که نیاز است آموزش دیده شوند و کار روی پایگاه‌های دانش با مقیاس زیاد است. این مدل‌ها روابط موجود بین موجودیت‌ها را به خوبی تشخیص داده و پس از آموزش به دسته‌بندی خوبی از این موجودیت‌ها دست پیدا می‌کنند اما در مورد پیوندهای بین موجودیت‌ها اینطور نیست و روابط بین پیوندها بخوبی تشخیص داده نمی‌شود. در این پژوهش قصد داریم به کمک قوانین انجمنی روابط بین پیوندها را نیز تشخیص دهیم و در آموزش مدل دخیل کنیم و همچنین چارچوبی ارائه دهیم که به کمک آن بتوان نقاط ضعف و قوت روش‌های موجود را بررسی کرد.

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



## فهرست مطالب

فصل ۱: مقدمه	۱
۱-۱- تعریف مسئله	۳
۱-۲- روش انجام پژوهش	۴
۱-۳- کاربردهای پژوهش	۴
۱-۴- ساختار پایان نامه	۵
فصل ۲: پیشینه پژوهش	۶
۲-۱- زمینه‌های تحقیق	۷
۲-۱-۱- شبکه‌های همگون و ناهمگون	۷
۲-۱-۲- پیش‌بینی پیوند	۹
۲-۱-۳- فرض جهان‌باز و فرض جهان‌بسته	۱۰
۲-۱-۴- گرادیان نزولی تصادفی (SGD)	۱۱
۲-۲- روش‌های مبتنی بر ویژگی‌های پنهان	۱۲
۲-۳- تقسیم‌بندی داده‌های پایگاه دانش	۱۴
۲-۴- مراحل آموزش روش‌های مبتنی بر ویژگی‌های پنهان	۱۶
۲-۴-۱- مرحله‌ی آموزش	۱۷
۲-۴-۲- مرحله‌ی ارزیابی	۱۷
۲-۴-۳- مرحله‌ی آزمون	۱۹
۲-۵- بررسی روش‌های موجود	۲۰
۲-۵-۱- روش‌های خطی	۲۰

۲۱	۲-۵-۲- روش‌های دوخطی - رسکال
۲۶	۲-۵-۳- روش‌های ادراک چندلایه‌ای
۳۱	2-5-4- شبکه‌های عصبی تنسور
۳۳	۲-۵-۵- روش‌های فاصله‌ی پنهان
۳۳	2-5-6- Structured embedding مدل
۳۵	2-5-7- TransE مدل
۳۹	2-5-8- TransH مدل
۴۲	فصل ۳: قوانین انجمنی
۴۲	۳-۱- معیارهای اطمینان
۴۳	۳-۱-۱- معیار پشتیبانی قانون:
۴۴	۳-۱-۲- معیار پوشش سر
۴۴	۳-۱-۳- معیار اطمینان استاندارد
۴۸	۳-۱-۴- معیار اطمینان با فرض نیمه‌کامل
۴۹	۳-۲- الگوریتم‌ها
۵۲	۳-۳- قوانین هدف
۵۳	۳-۳-۱- R-subsumption قانون
۵۳	۳-۳-۲- R-equivalence قانون
۵۴	۳-۳-۳- 2-hope translation قانون
۵۴	۳-۳-۴- Triangle alignment قانون
۵۵	۳-۳-۵- Specific R-subsumption قانون
۵۶	۳-۴- جمع‌بندی

فصل ۴: روش پیشنهادی	۵۷
۴-۱- چالش‌ها	۵۷
۴-۲- عملکرد روش‌های موجود	۵۷
۴-۲-۱- روش رسکال	۵۸
۴-۲-۲- روش NTN	۵۸
4-2-3- Structured Embedding روش	۵۹
۴-۲-۴- روش TransE	۶۰
۴-۲-۵- روش TransH	۶۰
۴-۳- نقاط قوت و ضعف روش‌های موجود	۶۱
۴-۴- بررسی عمل کرد روش‌ها در یک نگاه	۶۸
۴-۵- استفاده از قوانین انجمنی برای بهبود نتایج	۷۱
۴-۵-۱- قانون R-subsumtion	۷۲
4-5-2- R-equivalence قانون	۷۲
۴-۵-۳- قانون تعدی (2-hope translation)	۷۲
4-5-4- Triangle alignment قانون	۷۴
4-5-5- Specefic R-subsumtion قانون	۷۴
فصل ۵: ارزیابی	۷۵
۵-۱- مقدمه	۷۵
۵-۲- آزمایش‌ها	۷۵
۵-۳- جمع‌بندی	۸۳

فصل ۶: نتیجه گیری .....	۸۷
۶-۱- نتیجه گیری .....	۸۷
۶-۲- کارهای آینده .....	۸۷
فصل ۷: مراجع .....	۹۱
فصل ۸: واژه‌نامه انگلیسی به فارسی .....	۹۲
فصل ۹: واژه‌نامه فارسی به انگلیسی .....	۹۷



## فهرست اشکال

- شکل ۱-Error! No text of specified style in document. ..... **Error!**
- شکل ۲-۲ شبکه ناهمگون ..... **Bookmark not defined.**
- شکل ۳-۲ پیش‌بینی پیوند در شبکه ناهمگون ..... **Error! Bookmark not defined.**
- شکل ۱-۳ کاربرد قوانین انجمنی در پیش‌بینی پیوند ..... **Error! Bookmark not defined.**
- شکل ۵ نمونه‌ی سوال و پاسخ از مدل ..... ۱۸
- شکل ۶ نحوه‌ی مقایسه‌ی ویژگی‌ها در روش‌های خطی ..... ۲۰
- شکل ۷ نحوه‌ی نگاشت ویژگی‌ها در روش‌های دوخطی ..... ۲۱
- شکل ۸ تبدیل ماتریس به تنسور در روش RESCAL ..... ۲۴
- شکل ۹ تقسیم رابطه‌ی رسکال به دو لایه ..... ۲۷
- شکل ۱۰ روش ER-MLP ..... ۳۰
- شکل ۱۱ افزایش فاصله‌ی دو موجودیت که رابطه‌ی  $k$  بین آنها برقرار نیست پس از نگاشت توسط بردار انتقال  $k$  در روش Structred Embedding ..... ۳۴
- شکل ۱۲ کاهش فاصله‌ی دو موجودیت که رابطه‌ی  $k$  بین آنها برقرار است پس از نگاشت توسط بردار انتقال  $k$  در روش Structred Embedding ..... ۳۵
- شکل ۱۳ بازنمایی شبکه‌ی ناهمگون در فضای برداری ..... ۳۶
- شکل ۱۴-ب روش TransH ..... ۴۰
- شکل ۱۴-الف روش TransE ..... ۴۰
- شکل ۱۶ نمونه گراف ناهمگون با دو نوع رابطه ..... ۴۶
- شکل ۱۷ حقیقت‌هایی که در گراف نمونه موجود نیست ..... ۴۷
- شکل ۱۸ حالت ناشناس برای پیوندهایی که اطلاعی در مورد آن نداریم ..... ۴۷
- شکل ۱۹ نحوه‌ی استخراج قوانین انجمنی از حقیقت‌های پایگاه دانش ..... ۴۹
- شکل ۲۰ بازنمایی بردار ۵۰ هزار موجودیت که به روش TransE آموزش دیده شده است ..... ۶۳
- شکل ۲۱ بازنمایی قسمتی از موجودیت‌ها که در مورد ژانر مجموعه‌های تلوزیونی هستند ..... ۶۴
- شکل ۲۲ بازنمایی قسمتی از موجودیت‌ها که در مورد ایالت‌های آمریکا هستند ..... ۶۵
- شکل ۲۳ بازنمایی قسمتی از موجودیت‌ها که در مورد نام سریال‌های تلوزیونی آمریکا هستند ..... ۶۵
- شکل ۲۴ بازنمایی بردار روابط پایگاه دانش freebase15k که توسط روش TransE آموزش دیده شده است ..... ۶۷
- شکل ۲۵ بزرگنمای قسمتی از بردارهای روابط موجود در پایگاه دانش freebase15k ..... ۶۸

شکل ۲۶ درصد hit@10 در روش‌های مورد بررسی .....	۷۰
شکل ۲۷-چپ تاثیر اعمال قانون R-Subsumption بر hit@10 -راست تاثیر اعمال قانون R-Subsumption بر رتبه‌ی میانگین .....	۷۷
شکل ۲۸-راست تاثیر اعمال قانون SR-Subsumption بر رتبه‌ی میانگین - چپ تاثیر اعمال قانون SR-Subsumption بر hit@10 .....	۷۸
شکل ۲۹-چپ تاثیر اعمال قانون R-equivalence بر hit@10 - راست تاثیر اعمال قانون R-equivalence بر رتبه‌ی میانگین .....	۷۹
شکل ۳۰-چپ تاثیر اعمال قانون 2-hope translation بر hit@10 - راست تاثیر اعمال قانون 2-hope translation بر رتبه‌ی میانگین .....	۸۰
شکل ۳۱-چپ تاثیر اعمال قانون Triangle alignment بر hit@10 - راست تاثیر اعمال قانون Triangle alignment بر رتبه‌ی میانگین .....	۸۲
شکل ۳۲-چپ تاثیر اعمال همه‌ی قانون‌های هدف بر hit@10 - راست تاثیر اعمال همه‌ی قانون‌های هدف بر رتبه‌ی میانگین .....	۸۳

## فهرست جداول

جدول ۱ تقسیم‌بندی پایگاه‌های دانش .....	۱۴
جدول ۲ نمونه‌ی پایگاه دانش متشکل از دو رابطه .....	۴۳
جدول 3 قوانین انجمنی هدف .....	۵۵
جدول ۴ نمونه‌ی سوال از مدل TransE و ده پاسخ اول .....	۶۲
جدول ۵ تعداد پارامترهای هر روش .....	۶۹
جدول 6 مقدار رتبه‌ی میانگین و hit10 در روش‌های مورد بررسی .....	۷۰
جدول 7 تعداد قوانین استخراج شده روی freebase15k به تفکیک قوانین هدف .....	۷۶
جدول 8 نتایج اعمال قوانین انجمنی هدف بر روی معیار hti@10 روش‌های مورد بررسی .....	۸۳
جدول 9 نتایج اعمال قوانین انجمنی هدف بر روی معیار رتبه‌ی میانگین روش‌های مورد بررسی .....	۸۴



## فصل ۱: مقدمه

---

در سال های اخیر شاهد رشد بسیار زیادی در شبکه های اجتماعی بوده ایم و مطالعات زیادی روی این شبکه ها انجام گرفته است. داده های شبکه های اجتماعی یکی از ابزارهای محبوب برای مدل کردن رابطه و رفتار افراد و جامعه یا گروهی که در آن عضو هستند به شمار می رود. این داده ها معمولاً به صورت گرافی نمایش داده می شود که در آن گره ها افراد و لبه ها روابط بین این افراد می باشد. همچنین یادگیری ماشین مدت زیادی است که در علوم کامپیوتر جایگاه خود را پیدا کرده و به عنوان ابزار قدرتمندی برای کمک به انسان در زمینه های مختلف محسوب می شود و ماشین را بیش از پیش در خدمت انسان در آورده است.

در روش های اولیه یادگیری ماشین، عمدتاً از داده ها و متن های خام در زمینه ی یادگیری استفاده می شد. اخیراً از طرف برخی شرکت ها و موسسات بزرگ، همچون گوگل، ای بی ام، مایکروسافت و ... پایگاه های دانشی معرفی شده است که انجام راهکارهای مختلف یادگیری ماشین را ساده تر و کاربردی تر کرده است. در این پایگاه های دانش اطلاعات مورد نیاز برای عملیات های مختلف یادگیری به صورت منظم و نیمه منظم موجود است و دغدغه ی نرمال کردن و رفع خطا و استخراج حقایق را به مقدار زیادی کم کرده است. این پایگاه های دانش عمدتاً به صورت یک شبکه از موجودیت ها و روابط بین آن ها که می توان آن را به صورت یک گراف داده نمایش داده به این صورت که گره ها موجودیت ها و یال های بین گره ها نشانگر روابط بین آن ها باشند، که این یال ها می توانند از یک نوع باشند یعنی گراف نشانگر یک شبکه تک-رابطه ای باشد (همگون) یا هر یال با یک برچسب، نشانگر نوع رابطه باشد و شبکه نشانگر یک گراف چند رابطه ای (ناهمگون) باشد.

یکی از مسائل بنیادی و اساسی در یادگیری ماشین روی شبکه های اجتماعی، پیش بینی پیوند در شبکه های همگون و ناهمگون است به این معنی که از اطلاعات موجود در گراف دانش استفاده کرده و وجود یا عدم وجود یک یال را پیش بینی کرد، یا به عبارت دیگر مسأله ی پیش بینی پیوند این است که نمایی از یک شبکه به ما داده می شود و ما مایل هستیم که بدانیم در آینده نزدیک، احتمالاً چه تراکنش هایی میان اعضای فعلی شبکه روی می دهد و یا اینکه کدام یک از تراکنش های موجود را از دست می دهیم.

این راهکار در زمینه های مختلف یادگیری ماشین مورد استفاده قرار می گیرد و کاربرد گسترده ای در زندگی انسان پیدا کرده است. برای مثال از این راهکار در سیستم های توصیه گر در فروشگاه های اینترنتی، سیستم های تشخیص پزشکی، جواب گویی به سوال و ... استفاده می شود. اکثر مطالعات انجام شده در این زمینه روی شبکه های تک رابطه ای بوده است. به این معنا که روابط بین موجودیت ها از یک نوع است و این روابط به صورت دوتایی های مرتب مورد استفاده قرار می گیرند. برای مثال اگر در یک شبکه اجتماعی رابطه را دوستی بین افراد در نظر بگیریم یال های گراف شبکه به صورت «الف، ب» خواهد بود به این معنی که شخص الف با شخص ب رابطه دوستی دارد.

پایگاه های دانشی که اخیراً معرفی شده اند عمدتاً داده ها را به صورت داده های چند رابطه ای ذخیره می کنند و اطلاعات بیشتری از یک رابطه دوتایی بلی یا خیر به ما می دهند. منظور از داده های چند رابطه ای گراف جهت داری است متشکل از موجودیت ها و روابط بین آن ها که بصورت «مبدا h، رابطه r، مقصد t» نمایش داده می شود، به این معنی که یک رابطه r بین موجودیت های h و t وجود دارد. برای مثال سه تایی «تهران، واقع در، ایران» این اطلاع را به ما می دهد که استان تهران داخل کشور ایران قرار دارد. در این نوع پایگاه دانش هم انواع مختلف موجودیت وجود دارد و هم انواع مختلف رابطه بین موجودیت ها. پایگاه های دانشی مانند Graph Google Knowledge، FreeBase و WordNet وجود دارند که شامل تعداد زیادی نمونه چندرابطه ای می باشند و تعداد زیادی موجودیت و روابط بین آن ها را می توان در آن ها یافت و از آن برای یادگیری مدل استفاده کرد.

## ۱-۱- تعریف مسئله

روش های مختلفی برای حل مساله ی پیش بینی پیوند در پایگاه های دانش ناهمگون ارائه شده است که از رویکردهای مختلفی سعی به حل این مساله می کنند از جمله روش های آماری، روش های ویژگی های پنهان، روش های ویژگی های گراف و ... تمرکز ما در این مقاله بر روی روش های پیش بینی پیوند مبتنی بر ویژگی های پنهان خواهد بود.

در روش های مبتنی بر ویژگی های پنهان، با استفاده از ویژگی هایی که در موجودیت ها و روابط بین آنها وجود دارد سعی می شود میزان ارتباط بین موجودیت ها 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 ویژگی های مشترکی دارند، مثلاً هر دو اهل یک شهر هستند، هر دو در یک رشته ی دانشگاهی تحصیل کرده اند، خصوصیات اخلاقی یکسانی دارند و ... در رابطه ی همکار بودن به هیچ یک از این ویژگی ها به طور مستقیم اشاره نشده است و این ویژگی ها به طور ضمنی در این موجودیت ها قرار دارند که با استفاده از آن می توانیم وجود یا عدم وجود رابطه ی همکار بودن را حدس بزنیم. پس هر موجودیت می تواند تعداد زیادی ویژگی پنهان داشته باشد که رابطه ها به این ویژگی ها وزن می دهند، مثلاً در رابطه ی همکاری احتمال اینکه رشته ی تحصیلی دو شخص در همکار شدن آنها تاثیرگذار باشد بیشتر از ویژگی رنگ پوست دو شخص است، پس وزن ویژگی رشته ی تحصیلی در این رابطه بیشتر از ویژگی رنگ پوست است.

روش های زیادی برای حل مساله ی پیش بینی پیوند مبتنی بر ویژگی های پنهان ارائه شده است. همه ی این روش ها از یک روال ثابتی برای حل مساله استفاده می کنند و هر کدام با نوآوری هایی که داشته اند بهبودهایی در نتایج بدست آمده حاصل کرده اند. این روش ها در بخش ۵-۲ به طور کامل معرفی خواهند شد.

در این پژوهش قصد داریم که با استفاده از قوانین انجمنی موجود در پایگاه های دانش به دو هدف برسیم،

۱. بهبود نتایج در روش های معرفی شده ی موجود: در ادامه پنج روش از روش های معروف

مبتنی بر ویژگی های پنهان که به حل مساله ی پیش بینی پیوند پرداخته اند را انتخاب

می کنیم و در بخش ۲-۴ نشان می دهیم که استفاده از قوانین انجمنی در فرایند

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

۲. ارائه ی چارچوبی برای سنجش نقاط ضعف و قوت روش های ارائه شده: همچنین با دسته بندی کردن قوانین انجمنی در بخش ۳-۳- نشان می دهیم که با اعمال جداگانه ی دسته های مختلف قوانین انجمنی می توانیم میزان قدرت و ضعف روش ها را در قانون های مختلف بررسی کنیم که با برطرف کردن نقاط ضعف می توان روش های قوی تری 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 روش های معرفی شده فقط از روابط موجود در پایگاه دانش در فرایند آموزش استفاده شده است. مثلاً اگر در پایگاه دانش داشته باشیم که «علی، پدر، حسن» به این معنی که علی پدر حسن است می توانیم به این نتیجه برسیم که «حسن، فرزند، علی» و از آن در فرایند آموزش استفاده کنیم اما در این روش ها این روابط مورد استفاده قرار نگرفته اند. در این پژوهش قصد داریم که اینگونه قوانین را به کمک روش هایی که در بخش ۲-۳- توضیح داده می شوند از پایگاه دانش استخراج کنیم و به کمک معیارهای کیفیت که در بخش ۱-۳- معرفی می شوند قوانینی که کیفیت مناسب دارند را انتخاب کنیم و در فرایند آموزش از آن ها استفاده کنیم، نحوه ی استفاده از این قوانین در مدل های موجود در بخش ۳-۳- توضیح داده شده است.

## ۳-۱- کاربردهای پژوهش

مسائل پیش بینی پیوند در زمینه های زیادی قابل به کارگیری هستند و همین امر باعث شده است که در سال های گذشته بسیار مورد توجه و تحقیق قرار بگیرند. به عنوان مثال در ادامه تعدادی از این کاربردها را مطرح خواهیم کرد:

- پیش بینی پیوندهای احتمالی در شبکه های اجتماعی، برای مثال در شبکه های 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 به عنوان سیستم های توصیه گر، برای مثال کاربران و کالاها یا اشیاء موجودیت ها هستند و خریدن، امتیاز دادن، بررسی کردن و ... رابطه ها هستند که می توان از بررسی این روابط و موجودیت ها اطلاعاتی نظیر کاربر  $X$  کالای  $Y$  را خواهد خرید یا خیر یا اینکه کالای  $Y$  به تعداد بالا فروش خواهد رفت یا خیر بدست آورد.

- استفاده در سیستم های تشخیص پزشکی، برای مثال شبکه ی بین بیماران، بیماری ها، داروها و ... را در نظر بگیرید، با بررسی دقیق این شبکه می توان علل و درمان های بیماری ها را به کمک ماشین بدست آورد.

علاوه بر کاربردهایی که در بالا برای پیش بینی پیوند اشاره شد، از روش ارائه شده در این پژوهش می توان برای کامل تر کردن پایگاه های داده نیز استفاده کرد، به این صورت که روابطی که از قوانین انجمنی استخراج شده از پایگاه دانش بدست می آید و کیفیت لازم را دارد را نیز به پایگاه دانش اضافه کنیم و آن را کامل تر کنیم.

## ۴-۱- ساختار پایان نامه

ادامه ی این پایان نامه به این صورت است که در فصل ۲ پیشینه پژوهش و زمینه های تحقیق را شرح می دهیم که توضیح می دهیم که از چه ابزاری برای انجام این پژوهش استفاده شده است و همچنین روش های مختلف حل این مساله را توضیح می دهیم. در فصل ۳ مروری داریم بر قوانین انجمنی و معیارهایی برای سنجش کیفیت قوانین معرفی می کنیم و نحوه ی استخراج قانون ها از پایگاه های دانش را توضیح می دهیم و در انتهای این فصل پنج قانون هدفی که قصد استفاده از آن ها داریم را توضیح می دهیم. در فصل ۴ روش پیشنهادی برای بهبود روش های مبتنی بر ویژگی پنهان را شرح می دهیم و نحوه ی اعمال آن بر روی همه ی متدهای معرفی شده در فصل ۲ را توضیح می دهیم. در فصل ۵ آزمایش ها و نتایج آزمایش های انجام شده را آورده ایم و نهایتاً در فصل ۶ نتیجه گیری انجام شده و کارهایی که در آینده برای بهبود بیشتر این روش ها قابل انجام هست را توضیح می دهیم.

## فصل ۲: پیشینه پژوهش

در این بخش به معرفی روش های موجود که به حل مساله ی پیش بینی پیوند در شبکه های ناهمگون می پردازند، خواهیم پرداخت. روش هایی با راهکارهای مختلفی در حوزه های مختلف تلاش به حل این مساله کرده اند. این روش ها می توانند به سه دسته ی مختلف تقسیم می شوند: ۱- روش های مبتنی بر ویژگی های گراف ۲- روش های مدل تصادفی مارکوف ۳- روش های مبتنی بر ویژگی های پنهان. در ادامه توضیح مختصری در مورد هر دسته از روش ها می دهیم.

- در روش های مبتنی بر ویژگی های گراف از روی ویژگی های ساختاری گراف داده ها استفاده می شود مانند دسته بندی گره ها، دسته بندی نوع یال ها، تعداد گره های مشابه و ...
- روش های مدل تصادفی مارکوف که در آن دید بالایی از گراف داده نداشته و سعی می کنیم مساله را به صورت محلی حل کنیم به این صورت که روابط هر موجودیت را با موجودیت های اطراف آن بررسی می کنیم و جواب های محلی را بدست می آوریم.
- روش های مبتنی بر ویژگی های پنهان که در این روش ها هر موجودیت و نوع رابطه بین آنها به صورت برداری از ویژگی های پنهان تعریف می شود که ویژگی های پنهان نام دارد. برای مثال ویژگی هایی که یک موجودیت می تواند داشته باشد، محل به وجود آمدن آن، سن آن، جاندار یا بی جان بودن آن و ... است.

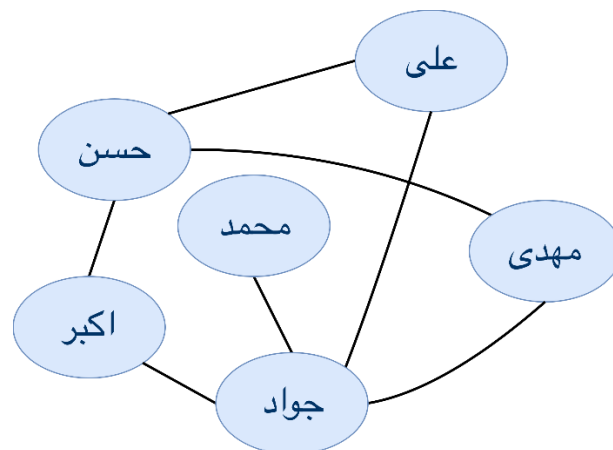
همان طور که در بخش ۲-۱-۲- گفتیم تمرکز ما در این تحقیق روی روش های مبتنی بر ویژگی های پنهان است. در ادامه این روش ها را به صورت کامل توضیح داده و تعدادی از آن ها را به طور مختصر معرفی کرده و نحوه ی کار آن ها را توضیح می دهیم و نتایج به دست آمده از آن ها را توضیح می دهیم.

## ۲-۱- زمینه های تحقیق

در این بخش مطالبی راجع به مباحث پایه ای که در ادامه تحقیق از آن ها استفاده شده است را مطرح خواهیم کرد.

### ۲-۱-۱- شبکه های همگون و ناهمگون

در سال های اخیر شبکه های اجتماعی پی شرفت زیادی داشته است و در زمینه های مختلفی شبکه سازی شده است. عمدتاً در این شبکه ها روابط خاصی مدنظر و قابل استخراج است، مثل روابط دوستی، همکاری و... که اگر گراف این گونه شبکه ها را رسم کنیم در این گراف ها یال ها از یک نوع است و نشانگر یک رابطه ی خاص است. مثلاً گرافی هست که همه ی گره های آن انسان ها هستند و یال های بین گره ها نشان دهنده ی وجود یا عدم وجود دوستی بین اشخاص است. این گونه شبکه ها که در آن ها یال و گره ها از یک نوع است را شبکه های همگون می نامیم [X]. در شکل ۱ یک نمونه شبکه ی همگون که در آن اشخاص و رابطه ی دوستی بین آن ها به تصویر کشیده شده است را مشاهده می کنیم.

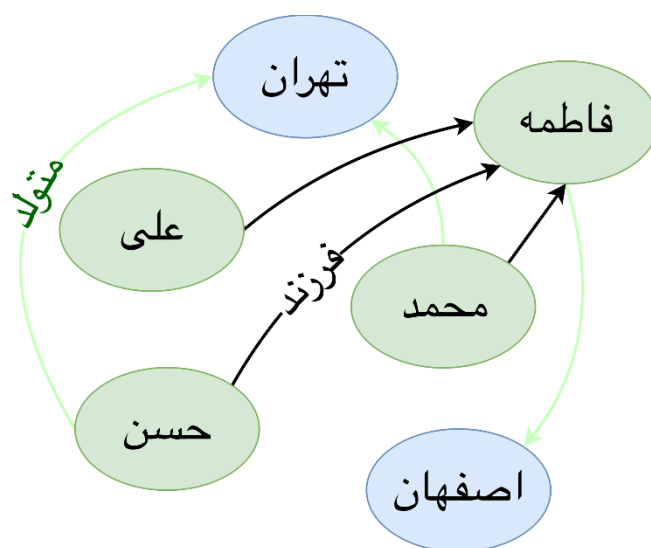


شکل ۱ شبکه ی همگون

اما همانطور که در مقدمه نیز اشاره شد، اخیراً شبکه های بزرگتر و پیچیده تری معرفی شده است که فقط یک نوع رابطه را پوشش نمی دهد و روابط مختلفی را بین موجودیت های مختلف در بر می گیرد، به این شبکه ها به علت یکسان نبودن نوع روابط و موجودیت ها شبکه های ناهمگون می گوییم، همچنین به خاطر وجود دانشی که در این شبکه ها نهفته و قابل استخراج است، آن را پایگاه دانش<sup>۱</sup> نیز می نامیم. در شکل ۲ یک قسمت کوچک از یک شبکه ی ناهمگون را مشاهده می کنیم که موجودیت های آن از دو نوع انسان و شهر هستند و روابط موجود در این شبکه از دو نوع «والد بودن» و «متولد شهر ... بودن» است.

---

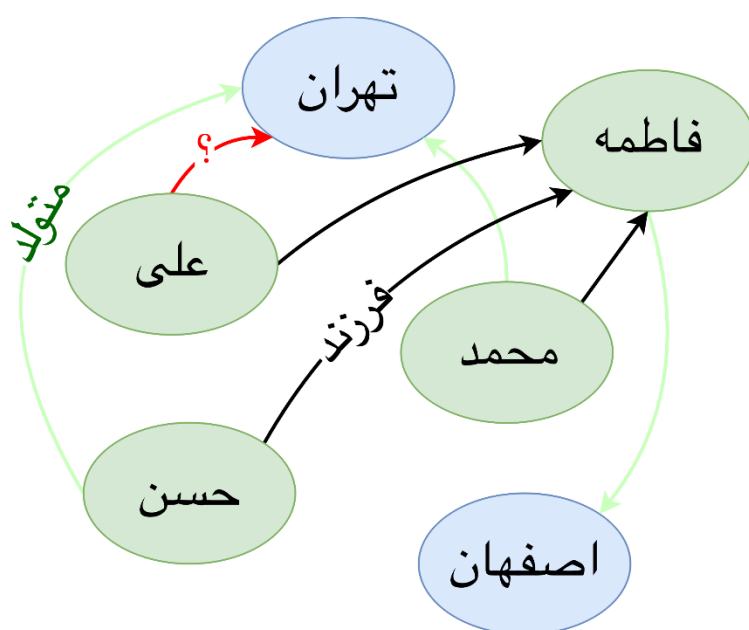
<sup>1</sup> Knowledgebase



شکل ۲ شبکه ناهمگون

## ۲-۱-۲- پیش بینی پیوند

یکی از روش های یادگیری ماشین که در زمینه های مختلف به کمک انسان آمده است و کارهای انسانی را تسهیل کرده است پیش بینی پیوند است. در پیش بینی پیوند یک گراف از روابط بین موجودیت ها به عنوان ورودی مساله دریافت می کنیم و وجود یا عدم وجود یک یال بین دو موجودیت را پیش بینی می کنیم. در شکل ۳ یک گراف از یک شبکه ی ناهمگون را مشاهده می کنیم. مساله ی پیش بینی لینک تلاش می کند که بررسی کند که رابطه ی «تولد» بین گره های «علی» و «تهران» قرار دارد یا خیر. این پیش بینی عمدتاً از روی دیگر روابط مرتبط بین موجودیت ها انجام می شود و با بررسی شباهت ها و معیارهایی که در ادامه ی این پژوهش توضیح خواهیم داد تصمیم می گیریم که این پیوند برقرار هست یا خیر.



شکل ۳ پیش بینی پیوند در شبکه ناهمگون

راهکارهای مختلفی برای حل این مساله مطرح شده است که به طور کلی می توان این راهکارها را به سه دسته ی ۱- روش های مبتنی بر ویژگی های گراف؛ ۲- روش های مدل تصادفی مارکوف؛ و ۳- روش های مبتنی بر ویژگی های پنهان تقسیم کرد که در این پژوهش تمرکز ما روی دسته ی سوم یعنی روش های مبتنی بر ویژگی های پنهان خواهد بود و در بخش ۴-۲ این روش ها را به تفصیل توضیح خواهیم داد.

### ۳-۱-۲- فرض جهان باز<sup>۱</sup> و فرض جهان بسته<sup>۲</sup>

فرض های جهان باز و جهان بسته در سیستم رسمی منطق<sup>۳</sup> برای بازنمایی دانش<sup>۴</sup> مورد استفاده قرار می گیرد. در فرض جهان بسته در نظر می گیریم که اگر داده ای در پایگاه دانش نبود، آن داده را غلط

<sup>۱</sup> Open world assumption (OWA)

<sup>۲</sup> Closed world assumption (CWA)

<sup>۳</sup> Formal system of logic

<sup>۴</sup> Knowledge representation

فرض می کنیم  $[X]$  . برای مثال اگر در پایگاه دانش حقیقت « $X, r, y$ » که به معنی این است که  $X$  با  $y$  رابطه ی  $r$  را دارد وجود نداشته باشد، می توانیم در نظر بگیریم که این حقیقت اشتباه است و مطمئنیم که  $X$  با  $y$  رابطه ی  $r$  ندارد.

اما در فرض جهان باز اینگونه نیست و اگر حقیقتی در پایگاه دانش وجود نداشته باشد نمی توانیم با اطمینان بگوییم که آن حقیقت اشتباه است، و ممکن است صحیح باشد  $[X]$  . وجود این فرض از کامل نبودن پایگاه های دانش ناشی می شود که وقتی نمی توانیم همه ی اطلاعات موجود در مورد موضوع مربوط به پایگاه دانش را جمع آوری و در آن قرار دهیم پس نمی توانیم در مورد حقیقت هایی که در پایگاه دانش نیست نظری بدهیم و این حقیقت را ناشناس<sup>۱</sup> در نظر می گیریم.

#### ۴-۱-۲- گرادیان نزولی تصادفی<sup>۲</sup> (SGD)

گرادیان نزولی تصادفی یک راهکار ساده و در عین حال موثر برای مسائل کمینه سازی یا بیشینه سازی یک تابع هدف است که به صورت مجموع یک تابع مشتق پذیر نوشته می شود. در کل SGD سعی می کند نقطه ی کمینه یا بیشینه را به کمک تکرار پیدا کند. این روش می تواند به صورت دسته ای روی پایگاه های دانش بزرگ اعمال شود که مشکل حافظه برای عملیات یادگیری روی پایگاه های دانش بزرگ که در حافظه ی اصلی جا نمی گیرند را حل می کند. همچنین دسته ای بودن این روش این قابلیت را به ما می دهد که داده های جدیدی که به صورت برخط به مجموعه اضافه می شوند را نیز بتوانیم وارد فرایند آموزش کنیم و نیاز به اجرای مجدد همه ی مراحل آموزش نباشد. نکته روش گرادیان تصادفی نزولی این است که نیاز نیست در هر تکرار کل مجموعه داده مورد بررسی قرار بگیرد و این روش با یک یا چند نمونه از مجموعه ی داده در هر تکرار قابل اجرا است. بروزرسانی پارامتر بصورت زیر انجام می پذیرد که در آن  $Q$  تابع هدف،  $w$  پارامتری که با تغییر آن قصد داریم تابع هدف را بهینه کنیم و  $\eta$  نرخ یادگیری است.

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

<sup>1</sup> unknown

<sup>2</sup> Stochastic gradient descent

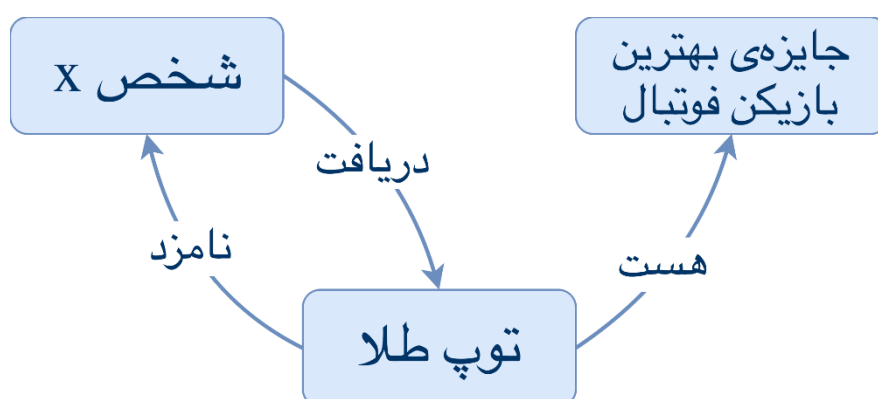
## ۲-۲- روش های مبتنی بر ویژگی های پنهان

روش های مبتنی بر ویژگی های پنهان از جدید ترین راهکارهایی است که در حوزه ی پیش بینی پیوند روی پایگاه های دانش موجود معرفی شده است. همانطور که در بخش قبل گفتیم این روش از ویژگی هایی که در نگاه اول از پایگاه دانش برداشت نمی شود استفاده می کند که ویژگی های پنهان نام دارند. در همه ی روش های مبتنی بر ویژگی پنهان این ویژگی ها را به صورت برداری تعریف می کنیم که هر مولفه از این بردار نشان دهنده ی یک ویژگی می باشد.

برای مثال در رابطه ی دوستی فاکتورهایی تاثیر گذار هستند و اگر در پایگاه دانش همه ی رابطه های دوستی موجود را بررسی کنیم به یک بردار از فاکتورها می رسیم که بردار رابطه ی دوستی را تشکیل می دهد. مثلا فاکتورهایی مانند شهر محل زندگی، سن، دانشگاه، رشته ی دانشگاهی، جنسیت، مذهب و ... در شکل گیری رابطه ی دوستی می تواند موثر باشد، اما اینکه هر کدام از این روابط چقدر در ایجاد رابطه ی دوستی تاثیر دارند و اهمیت هر کدام چقدر است و این میزان اهمیت را چگونه در تشخیص این رابطه تاثیر دهیم به مدل یادگیری بستگی دارد که در ادامه در معرفی هر یک از روش ها 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 می کنیم و میزان ارتباط این شخص را با معیارهای بازیکن خوب فوتبال بودن بررسی می کنیم. در مثال شکل ۴ شخص X هم برای توپ طلا نامزد شده و هم توپ طلا را دریافت کرده و از اطلاعات دیگر پایگاه دانش می دانیم که توپ طلا به بهترین بازیکن فوتبال هر سال داده می شود. پس شخصی که این جایزه را دریافت کرده بازیکن فوتبال خوبی است.





شکل ۴ کاربرد قوانین انجمنی در پیش بینی پیوند

روش های زیادی برای مدل های مبتنی بر ویژگی های پنهان معرفی شده است، روش هایی که در این تحقیق مورد مطالعه و بررسی قرار گرفته این به صورت زیر دسته بندی می شوند.

- روش های خطی
- روش های دو خطی
  - RESCAL
- روش های ادراک چندلایه ای
  - Neural Tensor Network (NTN)
- روش های فاصله ی پنهان
  - Structured Embedding
  - Translating Embedding (TransE)
  - Translating on Hyperplane (TransH)

در ادامه ابتدا پایگاه دانش Freebase که مطالعات روی آن انجام می گیرد و نحوه ی تقسیم بندی آن را توضیح خواهیم داد. سپس روش کلی آموزش مدل های مبتنی بر ویژگی های پنهان را شرح می دهیم و در ادامه روش هایی که در بالا نام برده شدند را توضیح داده و نقاط قوت و ضعف آن ها را بررسی می کنیم و نتایج بدست آمده از هر یک از این روش ها را نیز بررسی خواهیم کرد و در فصل بعد تلاش بر بهبود این روش ها خواهیم کرد.

### ۳-۲- تقسیم بندی داده های پایگاه دانش

برای آموزش دادن مدل های نام برده شده در بخش قبل از پایگاه دانش Freebase که قسمتی از گراف دانش تولید شده توسط گوگل است استفاده می کنیم. در این پایگاه دانش ۸۰ میلیون موجودیت، ۲۰ هزار نوع رابطه مختلف و ۱,۲ میلیارد حقیقت وجود دارد. حقیقت هایی که در این پایگاه دانش وجود دارد به صورت سه تایی مرتب

(subject, predicate, object)

مشخص شده اند به این صورت که موجودیت subject رابطه ی predicate دارد با موجودیت object. به این نوع ذخیره ی داده اصطلاحاً SPO گفته می شود. 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» می باشد.

این پایگاه دانش شامل تعداد زیادی حقیقت است که عملیات آموزش روی آن هزینه ی زیادی از نظر زمان و منابع خواهد داشت. برای سادگی و تسریع کار از یک نمونه ی نرمال کوچک این پایگاه دانش به نام Freebase15k استفاده می کنیم که در آن ۱۴۹۵۱ موجودیت، ۱۳۴۵ رابطه ی مختلف و ۵۹۲۲۱۳ حقیقت وجود دارد.

جدول ۱ تقسیم بندی پایگاه های دانش

آزمون	ارزیابی	آموزش	رابطه ها	موجودیت ها	پایگاه دانش
-------	---------	-------	----------	------------	-------------

FB15K	۱۴,۹۵۱	۱,۳۴۵	۴۸۳,۱۴۲	۵۰,۰۰۰	۵۹,۰۷۱
FB1M	$1 \times 10^6$	۲۳,۳۸۲	$17/5 \times 10^6$	۵۰,۰۰۰	۱۷۷,۴۰۴

روش های مبتنی بر ویژگی های پنهان به صورت تکراری انجام می شوند و نیاز است در هر تکرار بررسی کنیم که به آستانه ی مناسب برای قطع تکرار الگوریتم رسیده ایم 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.

- داده های آموزش: قسمت عمده ی داده ها برای عملیات آموزش استفاده می شود که الگوریتم اصلی هر روش روی آن اعمال می شود و سعی می کنیم پارامترهایی که همان ویژگی های پنهان هستند را تنظیم کنیم.

- داده های ارزیابی: قسمتی از داده که داده های ارزیابی نام دارند برای بررسی میزان بهبود یا تخریب مدل در هر تکرار استفاده می شوند، این داده ها کاملاً از داده های آموزش جدا هستند و در مرحله ی آموزش اصلاً به مدل نشان داده نمی شود و مدل تحت تاثیر این داده ها قرار نمی گیرد. در انتهای هر مرحله به کمک این داده ها بررسی می شود که تغییراتی که روی پارامترهای این مدل انجام شده باعث بهتر یا بدتر شدن این مدل شده است، در صورتی که بهبودی مشاهده شود تغییرات انجام شده در این مرحله نگه داشته می شود و به سراغ مراحل بعدی می رویم اما اگر نتایج بدتر شده باشد پارامترها را به مقادیر قبلی برگردانده و مرحله ی بعدی را شروع می کنیم.

- داده های آزمون: قسمت دیگری از داده که داده های آزمون نام دارند برای بررسی کیفیت عملکرد کلی مدل به کار می روند. این داده ها نیز کاملاً از داده های آموزش تفکیک شده اند و در زمان آموزش روی مدل تاثیری نمی گذارند و کلاً در هیچ یک از مراحل آموزش استفاده نمی شوند و تنها پس از آموزش مدل استفاده می شوند تا کیفیت مدل آموزش دیده شده را بررسی کنند.
- در این تحقیق پایگاه دانش Freebase15k به صورت شکل ۵ تقسیم و استفاده شده است، به این صورت که برای قسمت آموزش ۴۸۳,۱۴۲ حقیقت استفاده کرده ایم، برای قسمت ارزیابی ۵۰ هزار حقیقت استفاده کرده ایم و برای قسمت آزمون ۵۹ هزار حقیقت استفاده کرده ایم.



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

## ۴-۲- مراحل آموزش روش های مبتنی بر ویژگی های پنهان

همانطور که در بخش قبل گفته شد پایگاه دانش را به سه قسمت آموزش، ارزیابی و آزمون تقسیم می کنیم. نحوه ی آموزش کلی همه ی روش های مبتنی بر ویژگی های پنهان در سه مرحله انجام می شود:

۱- مرحله ی آموزش ۱- مرحله ی ارزیابی ۳- مرحله ی آزمون. مراحل آموزش و ارزیابی به صورت تکراری و معمولاً با تکرار بالا انجام می شوند و در هر تکرار بررسی می شود که بهبودی اتفاق افتاده است یا خیر، اگر بهبودی داشتیم نتایج این مرحله تکرار را نگه داشته و مرحله ی تکرار بعدی را شروع می کنیم و اگر بهبودی اتفاق نیافتاده بود نتایج این مرحله را تاثیر نمی دهیم و مرحله ی تکرار بعدی را شروع می کنیم.

در ادامه این سه مرحله را توضیح می دهیم.

### ۱-۴-۲- مرحله ی آموزش

همانطور که قبل تر توضیح داده شد آموزش به صورت تکراری انجام می شود و در هر تکرار الگوریتم آموزش روش مورد نظر، روی قسمتی یا همه ی داده های مجموعه ی آموزش انجام می شود. در این قسمت سعی می شود که پارامترهای همه ی ویژگی ها جهت دهی شده و آموزش ببینند تا کمترین خطا در پاسخ به سوالاتی که از مدل پرسیده می شود را داشته باشند.

### ۲-۴-۲- مرحله ی ارزیابی

پس از هر تکرار مدل آموزش دیده شده را روی داده های ارزیابی اجرا می کنیم و معیارهایی که برای بررسی کیفیت مدل در نظر گرفته ایم را بدست می آوریم و از روی آن میزان بهبود مدل در این تکرار را بررسی می کنیم.

برای مثال فرض کنید که حقیقت زیر در مجموعه داده های ارزیابی وجود دارد و در مرحله ی 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 مرحله ی ارزیابی موجودیت اول یا آخر این حقیقت را حذف می کنیم و قسمت حذف شده را از مدل سوال می پرسیم و انتظار داریم که قسمت حذف شده را حدس بزند. سوالی که از این مدل پرسیده می شود به این صورت است:

(WALL-E, has\_genre, ?)

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

در مدل های مبتنی بر ویژگی های پنهان پاسخ به اینگونه سوالات به صورت مجموعه ی مرتب شده ی همه موجودیت هاست. یعنی این مدل میزان نزدیکی همه ی موجودیت ها با پاسخ این سوال را بررسی می کند و به ترتیب نزدیک به دور مرتب می کند و به عنوان پاسخ به ما می دهد. برای مثال پاسخ به سوال بالا به صورت شکل ۶ می باشد.

(WALL-E, has\_genre, Fantasy)

WALL-E has the genre?!	1- Animations
(WALL-E, has_genre, ?)	2- Computer Animation
	3- Comedy film
	4- Adventure film
	5- Science Fiction
	<b>6- Fantasy</b>
	7- Stop motion
	8- Satire
	...

شکل ۶ نمونه ی سوال و پاسخ از مدل

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

در مرحله ی 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:

- رتبه ی میانگین (Mean rank): میانگین رتبه ی جواب های درستی که مدل داده است.
- حدس زیر ۱۰: درصد سوالاتی که پاسخ درست مدل به آن زیر رتبه ی ۱۰ بوده است که در ادامه به آن  $hit@10$  می گوئیم.

همانطور که قبل تر توضیح داده شد ۵۰ هزار حقیقت در دسته ی ارزیابی وجود دارد، ما پس از هر تکرار الگوریتم دو معیار بالا را به دست آورده و میزان بهبود مدل را اندازه می گیریم. پس از پرسیدن این ۵۰ هزار سوال میانگین رتبه ای که جواب های درست داشته معیار اول را به ما می دهد و درصد سوالاتی که جواب درست آن زیر رتبه ی ۱۰ بوده است معیار دوم را به ما می دهد.

### ۳-۴-۲- مرحله ی آزمون:

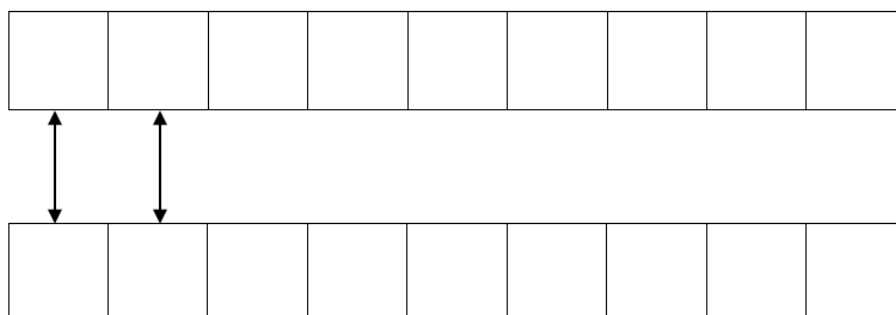
پس از انجام کامل مراحل آموزش و ارزیابی و متوقف شدند الگوریتم از داده ای آزمون که در بخش ۲-۳ توضیح دادیم و معیارهای کیفیت که در بخش ۱-۴-۲- توضیح دادیم استفاده می کنیم و کیفیت مدل را بررسی می کنیم. در این بخش هم مانند بخش ارزیابی داده ها به مدل در حال آموزش نشان داده نشده و برای مدل جدید هستند. در این مرحله هم یک قسمت از هر حقیقت موجود در داده های آزمون را حذف کرده و آن را از مدل سوال می پرسیم، دقیقا به مانند مرحله ی ارزیابی. پس از پرسیدن سوالات دو معیار رتبه ی میانگین و  $hit@10$  را بدست می آوریم که این دو معیار نشان دهنده ی میزان کیفیت و دقت روش است.

## ۵-۲- بررسی روش های موجود

در ادامه نحوه ی عملکرد روش های موجود را توضیح داده و بررسی می کنیم.

### ۱-۵-۲- روش های خطی

همانطور که در بخش ۲-۲- گفتیم در روش های مبتنی بر ویژگی های پنهان موجودیت ها و رابطه ها به صورت بردارهایی در یک فضای  $n$  بعدی تبدیل می شوند که به کمک معیارهای مختلف شباهت، رابطه ی بین دو موجودیت را بدست می آوریم. در روش های خطی در زمان آموزش و بررسی میزان شباهت بردارها را به صورت خطی با یکدیگر مقایسه می کنیم، به این صورت که هر اندیس از بردار موجودیت اول را با اندیس متناظر آن در بردار رابطه یا موجودیت دیگر بررسی می کنیم، شکل ۷.



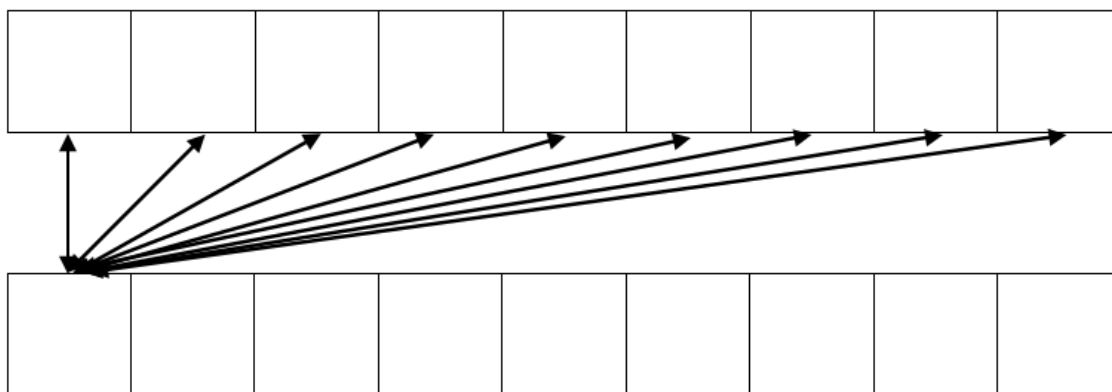
شکل ۷ نحوه ی مقایسه ی ویژگی ها در روش های خطی

با توجه به نتایج ضعیفی که روش های این دسته در آموزش مدل و پیش بینی پیوند بدست آورده اند به این روش ها نمی پردازیم و به همین معرفی اکتفا می کنیم.



## ۲-۵-۲- روش های دوخطی - رسکال

روش RESCAL [] یکی از روش های ویژگی های پنهان رابطه ای<sup>۱</sup> است که حقیقت های پایگاه دانش را به صورت تراکنش های بین جفت ویژگی های پنهان در نظر می گیرد. یعنی بر خلاف روش های خطی هر ویژگی پنهان از هر بردار را با همه ی ویژگی های پنهان دیگر بردار بررسی می کند، به صورت شکل ۸. به همین دلیل این روش را روش دوخطی نیز می نامیم.



شکل ۸ نحوه ی نگاشت ویژگی ها در روش های دوخطی

در این روش امتیاز هر سه تایی را از رابطه ی (۲-۱) بدست می آوریم که در آن  $H_e$  تعداد ابعاد بردار در نظر گرفته شده برای موجودیت ها است (تعداد ویژگی های 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 رابطه ی  $k$  چقدر با هم کنش می کنند.

<sup>1</sup> Relational latent factor

$$f_{ijk}^{RESICAL} := e_i^T W_k e_j = \sum_{a=1}^{He} \sum_{b=1}^{He} w_{abk} e_{ia} e_{jb} \quad (1-2)$$

همانطور که در رابطه ی (۱-۲) مشاهده می شود 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 رابطه ی 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 (۱-۲) به ما می دهد بدست می آوریم. این امتیاز به این صورت محاسبه می شود که بردار موجودیت a در ماتریس مربوط به k ضرب شده و پس از آن در بردار b ضرب می شود که نتیجه ی آن یک مقدار حقیقی است که امتیاز این حقیقت را به ما می دهد.

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

آموزش رابطه ای<sup>۱</sup> از طریق بازنمایی مشترک<sup>۲</sup>: در رابطه ی + هر موجودیت به صورت یک بردار بازنمایی شده است بدون توجه به اینکه در قسمت اول حقیقت می آید یا قسمت دوم آن. همچنین این موجودیت ها برای همه ی رابطه ها یک بازنمایی مشترک دارند و در هر رابطه نیاز به تعریف جدید ندارند. برای مثال موجودیت i در قسمت اول حقیقت  $x_{ijk}$  با رابطه ی k آمده است و همین موجودیت در رابطه ی  $x_{piq}$  به عنوان موجودیت دوم در رابطه ی q ظاهر شده است. هر دو تابع امتیاز  $f_{ijk} = e_i^T W_k e_j$  و  $f_{piq} = e_p^T W_q e_i$  از یک بازنمایی برای موجودیت i (e<sub>i</sub>) استفاده می کنند. بنابراین همه ی پارامترها به صورت مشترک<sup>۳</sup> آموزش دیده می شوند و این بازنمایی مشترک باعث می شود که اطلاعات روی همه ی

<sup>1</sup> Relational Learning

<sup>2</sup> Shared representations

<sup>3</sup> jointly

حقیقت ها به وسیله ی بازنمایی موجودیت ها و ماتریس وزن دار رابطه ها پخش شوند و بتوانیم وابستگی های جهانی<sup>۱</sup> در داده ها را تشخیص دهیم [1].

ارتباط معنایی بردارها<sup>۲</sup>: خاصیت بازنمایی مشترک در این روش کمک می کند که میزان شباهت موجودیت ها در فضای رابطه ای<sup>۳</sup> نیز بدست بیاید. برای مثال موجودیت هایی که با رابطه های مشابه به موجودیت های 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 موجودیت ها با تعداد زیادی رابطه ی مشترک بازنمایی یکسانی خواهند داشت. این خصوصیت می تواند در بازنمایی<sup>۴</sup> و خوشه بندی های با مقیاس بالای<sup>۵</sup> موجودیت ها روی داده های رابطه ای مورد استفاده قرار گیرد. [63,64,1]

ارتباط با عامل بندی تنسور<sup>۶</sup>: این روش شباهت زیادی به روش های استفاده شده در سیستم های توصیه گر [66] و عامل بندی تنسور سنتی دارد [۶۷]. ضرب ماتریس که در معادله ی + معرفی کردیم می تواند به صورت  $F_k = EW_kE^T$  نوشته شود که در آن  $F \in R^{Ne*Ne}$  ماتریسی است که همه ی امتیازات مربوط به رابطه ی  $k$  را در خود جا داده است و سطر  $i$  ام از ماتریس  $E \in R^{Ne*He}$  بازنمایی موجودیت  $e_i$

---

<sup>1</sup> global dependency

<sup>2</sup> Semantic embeddings

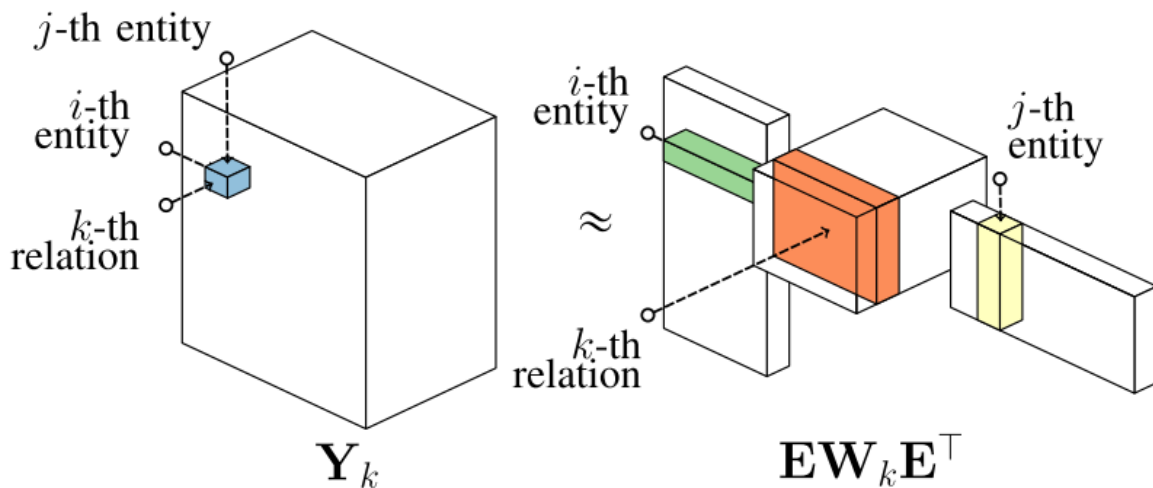
<sup>3</sup> relational domain

<sup>4</sup> resoluton

<sup>5</sup> large-scale hierarchical clustering

<sup>6</sup> Tensor factorization

است که برداری از ویژگی های پنهان این موجودیت است. در شکل ۹ این تبدیل ماتریس به تنسور نمایش داده شده است. [1]



شکل ۹ تبدیل ماتریس به تنسور در روش RESCAL

برازش مدل: اگر بخواهیم از یک مدل آماری استفاده کنیم، پارامترهای این روش می توانند به صورت یک مدل کمینه سازی بر مبنای گرادیان<sup>۱</sup> تخمین زده شوند مانند گرادیان نزولی تصادفی (SGD)<sup>۲</sup> [68]. این روش می توانیم به صورت یک روش بر مبنای امتیاز حل کرد که می تواند از پارامترهای مدل را به صورت بسیار بهینه ای تخمین بزند: با توجه به ساختار تنسور که در بالا توضیح دادیم و همچنین با توجه به تنک بودن داده های موجود، نشان می دهیم که روش رسکال می تواند به کمک توالی<sup>۳</sup> ای از بروزرسانی های بسته ی کارآمد<sup>۴</sup> محاسبه شود [63,64]. به صورت تحلیلی می توان نشان داد که در این

<sup>1</sup> gradient-based minimization

<sup>2</sup> stochastic gradient descent

<sup>3</sup> sequence

<sup>4</sup> efficient close-form update

راهکار با هر بروزسانی در  $E$  و  $W_k$  به صورت خطی با تعداد موجودیت ها  $N_e$ ، تعداد رابطه ها  $N_r$  و تعداد حقیقت های مشاهده شده توسط مدل رشد می کند. [64]

پیش بینی مجزا: در رابطه ی (۱-۲) احتمال وجود یک رابطه از روی یک ضرب ماتریسی ساده از مرتبه ی  $O(H_e^2)$  بدست می آید. بنابراین، زمانی که پارامترهای مدل تخمین زده شدند، پیچیدگی محاسبات برای پیش بینی امتیاز یک حقیقت فقط به تعداد ویژگی های پنهان وابسته است و مستقل از اندازه ی کل گراف است. با این حال به لطف بازنمایی مشترک که قبل تر توضیح داده شده، این مدل می تواند در زمان تخمین پارامترها، وابستگی های جهانی بین موجودیت ها و رابطه ها را فهمیده و در فرایند آموزش تاثیر دهد. [1]

نتایج یادگیری رابطه ای: رسکال در زمینه های مختلف مدل های یادگیری مدرن<sup>۱</sup> مورد استفاده قرار گرفته است. برای مثال در [63] نشان داده شده است که رسکال موفق شده در پیش بینی رابطه نتایج نزدیک یا بهتر روی چندین مجموعه داده ی معیار نسبت به روش های [70] Markov Logic Networks و [71, 72] Bayesian Clustered و the Infinite (Hidden) Relational model و [73] Tensor Factorization بدست آورد. همچنین رسکال برای پیش بینی پیوند روی کل پایگاه دانش مانند YAGO و DBpedia مورد استفاده قرار گرفته است [64, 74]. فارق از پیش بینی پیوند، رسکال در روش های یادگیری رابطه ای تصادفی (SRL)<sup>۲</sup> مانند بازنمایی موجودیت ها<sup>۳</sup> و خوشه بندی بر مبنای پیوند<sup>۴</sup> عملکرد موفقی داشته است. برای مثال این روش عملکرد موفقی در دسته بندی

---

<sup>1</sup> state-of-the-art

<sup>2</sup> stochastic relational learning

<sup>3</sup> entity resolution

<sup>4</sup> link-based clustering

نویسندگان، ناشران و سالن های انتشار روی مجموعه داده های ناشران داشته است [63,65]. علاوه بر این، ارتباط معنایی موجودیت ها که در این روش محاسبه شد در ایجاد طبقه بندی<sup>۱</sup> به کمک خوشه بندی سلسله مراتبی<sup>۲</sup> روی داده های دسته بندی نشده<sup>۳</sup> به کار برده شده است [75].

### ۳-۵-۲- روش های ادراک چند لایه ای<sup>۴</sup>

می توانیم رسکال را به صورت مدلی که برای هر حقیقت یک بازنمایی تولید می کند و از روی این بازنمایی ها وجود یا عدم وجود این حقیقت ها را پیش بینی می کند تفسیر کنیم. به طور خاص می توانیم رسکال را به صورت رابطه های (۲-۲) و (۳-۲) باز نویسی کنیم.

$$f_{ijk}^{RESICAL} := w_k^T \phi_{ij}^{RESICAL} \quad (2-2)$$

$$\phi_{ij}^{RESICAL} = e_j \otimes e_i \quad (3-2)$$

در روابط (۱-۲) ضرب داخلی  $e_i$  و  $e_j$  که در محاسبه ی مجموع امتیازها شرکت می کردند را از مجموع بیرون کشیده و به صورت ضرب خارجی نوشتیم. بنابراین رسکال بازنمایی جفت موجودیت  $i$  و  $j$  را به صورت ضرب تنسور ویژگی های پنهان این دو موجودیت بدست آورده است. می توان احتمال وجود حقیقت  $x_{ijk}$  را از روی ضرب داخلی بازنمایی جفت موجودیت ها ( $\phi_{ij}$ ) و ماتریس وزن دار رابطه ( $w_k^T$ )

---

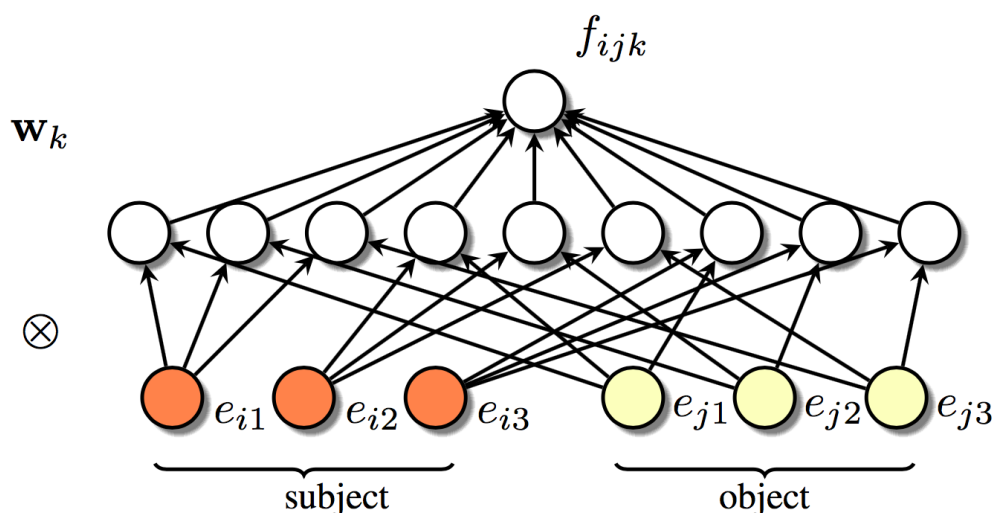
<sup>1</sup> taxonomies

<sup>2</sup> hierarchical clusterings

<sup>3</sup> uncategorized data via

<sup>4</sup> Multi-layer perceptrons

بدست آورد. این تقسیم بندی رابطه ی رسکال در شکل ۱۰ به تصویر کشیده شده است. توضیح بیشتر در مورد ایجاد بازنمایی پنهان به وسیله ی ضرب تنسور در [88, 89, 90] آمده است.



شکل ۱۰ تقسیم رابطه ی رسکال به دو لایه

از آنجایی که ضرب تنسور تراکنش بین همه ی جفت موجودیت ها را مدل می کند، وقتی تعداد ویژگی های پنهان زیاد باشد، رسکال در این راهکار به تعداد زیادی پارامتر نیاز پیدا خواهد کرد. این موضوع می تواند باعث مشکل در مقیاس پذیری روش روی پایگاه های دانش بزرگ با تعداد زیاد رابطه شود.

در ادامه ی این بخش به روش های ادراک چندلایه ای که به شبکه های عصبی پیش خور<sup>۱</sup> نیز معروف هستند می پردازیم. این راهکار این امکان را به ما می دهد که مدل های جایگزینی برای ساختن بازنمایی حقیقت ها در نظر بگیریم و همچنین بتوانیم از توابع غیر خطی در پیش بینی وجود پیوندها استفاده کنیم.

<sup>1</sup> feedforward neural networks

مدل<sup>۱</sup> E-MLP:

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

$$f_{ijk}^{E-MLP} := w_k^T g(h_{ijk}^a) \quad (4-2)$$

$$h_{ijk}^a = A_k^T \phi_{ij}^{E-MLP} \quad (5-2)$$

$$\phi_{ij}^{E-MLP} = [e_i; e_j] \quad (6-2)$$

در روابط فوق  $g(u) = [g(u_1), g(u_2), \dots]$  یک تابع  $g$  است که بر روی تک تک المان های بردار  $u$  اعمال می شود. که می تواند یک تابع غیر خطی باشد، مانند  $g(u) = \tanh(u)$ .

در این روابط  $h_a$  یک لایه مخفی<sup>۲</sup> اضافه شده است که ماتریس وزن دار دیگر روی بازنمایی موجودیت ها اعمال می کند. در اصل ما در رابطه ی + دو موجودیت  $e_i$  و  $e_j$  را ترکیب کردیم و هیچ کنشی بین این دو موجودیت محاسبه و تاثیر داده نشده است، بنابراین نیاز به یک ماتریس وزن دار برای محاسبه ی تاثیر این دو موجودیت نیاز بود و  $h_a$  وارد معادله شد.

---

<sup>1</sup> Entity-MLP

<sup>2</sup> hidden layer



بزرگترین تفاوتی که این روش با روش های ضرب تنسور مانند رسکال دارد این است که در اینجا بجای محاسبه ی همه ی تراكش های ممکن بین دو موجودیت، فقط تراكش های موجود در  $A_k$  مورد بررسی قرار می گیرند. این راهکار به طور قابل ملاحظه ای تعداد پارامتری که باید آموزش دیده شوند را کاهش می دهد.

### روش<sup>۱</sup> ER-MLP:

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

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

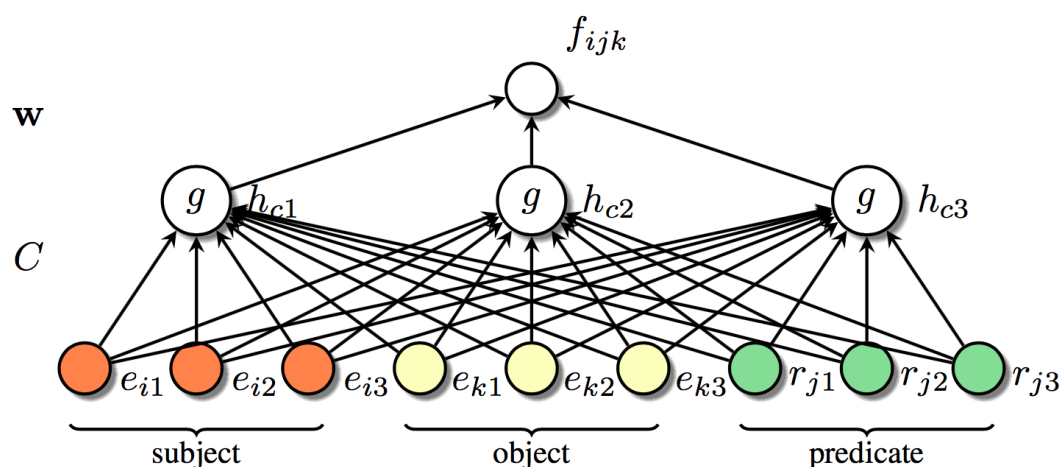
$$h_{ijk}^c = C^T \phi_{ijk}^{ER-MLP} \quad (۸-۲)$$

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

---

<sup>1</sup> Entity-Relation-MLP

توجه کنید که در این روش از یک بردار وزن دار جهانی برای همه روابط استفاده شده است و بردار  $C$  تعریف شده مستقل از رابطه ای  $r_k$  است که باعث کاهش تعداد پارامترهای مساله می گردد. نحوه ی کار روش ER-MLP در شکل ۱۱ نمایش داده شده است.



شکل ۱۱ روش ER-MLP

در [91] نشان داده شده است که روش های MLP کلماتی که قرابت معنایی<sup>۱</sup> دارند را به درستی نزدیک به یکدیگر تشخیص می دهند در حالی که برای چنین کاربردی آموزش دیده نشده اند. همچنین در [28] این ویژگی نشان داده شده است، برای مثال به کمک MLP مساله ی نزدیک ترین همسایه<sup>۲</sup> برای بازنمایی پنهان چندین رابطه ی انتخاب شده از پایگاه دانش Freebase را حل کرده اند. در نتایج می توان مشاهده کرد که روابطی که ارتباط معنایی دارند نزدیک به یکدیگر قرار گرفته اند.

<sup>1</sup> semantically similar

<sup>2</sup> nearest neighbors

۴-۵-۲- شبکه های عصبی تنسور<sup>۱</sup>

در [92] با ترکیب روش های ادراک چند لایه ای (MLP) و روش های دوخطی (Bilinear) روش جدیدی به نام شبکه های عصبی تنسور (NTN) معرفی شده است. روابط محاسبه ی تابع امتیاز این روش به صورت (۲-۱۰) و (۲-۱۱) و (۲-۱۲) است.

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

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

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

در اینجا  $B_k$  یک تنسور است، که در آن میزان رابطه ی بین جفت موجودیت ها نگاشت شده است.  $h_{ijk}^b$  را یک لایه ی مخفی دوخطی<sup>۲</sup> می نامیم، زیرا هم به صورت یک لایه ی مخفی از مدل های MLP هست و هم به کمک یک تابع وزن دار میزان کنش جفت موجودیت ها را مشخص می کند مانند آنچه در روش رسکال توضیح دادیم.

با توجه به معادله های معرفی شده در (۲-۱۰) و (۲-۱۱) و (۲-۱۲) مشخص است که این روش مشکلات مقیاس پذیری که در هر دو روش E-MLP و RESCAL وجود داشت را دارد. همچنین در [95] و

---

<sup>1</sup> Neural tensor networks

<sup>2</sup> bilinear hidden layer

[28] نشان داده شده است که این روش به بیش برآزش<sup>1</sup> میل می کند (حداقل روی مجموعه داده هایی که در این مقالات استفاده شده است) [1].

---

<sup>1</sup> overfit

۵-۵-۲- روش های فاصله ی پنهان<sup>۱</sup>

کلاس دیگری از مدل ها، مدل های فاصله ی پنهان هستند (که در تحلیل شبکه های اجتماعی به مدل های فضای پنهان معروف هستند) که احتمال وجود رابطه بین بین موجودیت ها را از فاصله ی بین بازنمایی پنهان آن ها در فضا می سنجد به این صورت که: موجودیت ها با یکدیگر رابطه دارند اگر بازنمایی پنهان آن ها با یک معیار فاصله نزدیک به هم باشد [۱].

در [96] مدلی برای داده های تک-رابطه ای<sup>۲</sup> برای اولین بار راهکاری در این زمینه معرفی شده است.

این راهکار احتمال وجود پیوند در شبکه های اجتماعی را با تابع امتیاز  $f(e_i, e_j) = d(e_i, e_j) - \text{محاسبه می کند، که در آن } d(.,.)$  یک تابع اندازه گیری فاصله ی دلخواه مانند فاصله ی اقلیدسی<sup>۳</sup> است. در ادامه سه روش Structured Embedding (SE)، TransE و TransH را به تفصیل توضیح می دهیم.

## ۶-۵-۲- مدل Structured embedding

در [93] مدلی به نام Structured Embedding (SE) معرفی شده که در آن ایده ای که در بند قبل توضیح داده شد برای داده های چند-رابطه ای<sup>۴</sup> گسترش داده شده است. در این روش تابع امتیاز برای حقیقت های  $X_{ijk}$  به صورت  $(2-13)$  مدل شده است.

---

<sup>1</sup> Latent distance models

<sup>2</sup> uni-relational data

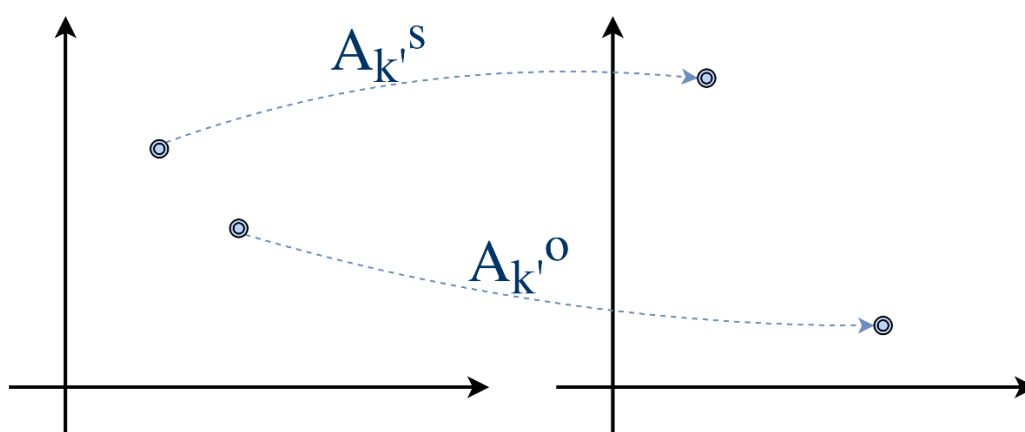
<sup>3</sup> Euclidean distance

<sup>4</sup> multi-relational

$$f_{ijk}^{SE} := -\|A_k^s e_i - A_k^o e_j\|_1 = -\|h_{ijk}^a\|_1 \quad (13-2)$$

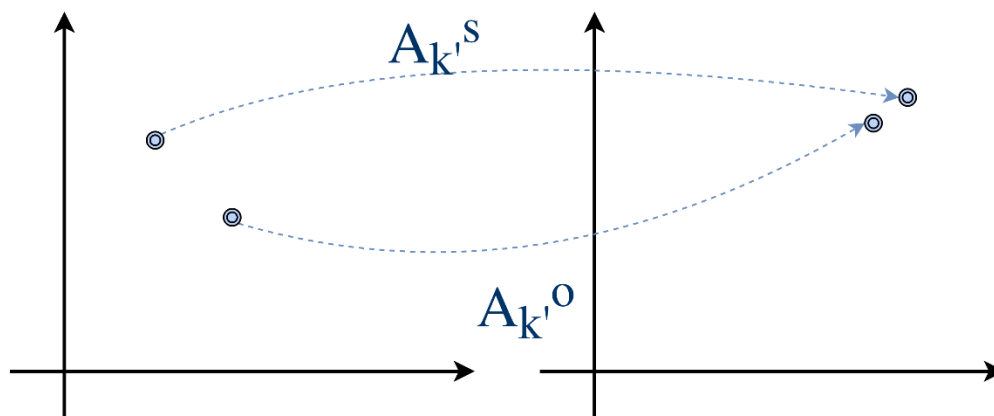
در رابطه ی (۱۳-۲)  $A_k$  از ماتریس های مربوط به موجودیت های اول و دوم  $[A_k^s, A_k^o]$  تشکیل شده است. ماتریس های  $A_k^o$  و  $A_k^s$  بازنمایی ویژگی های پنهان موجودیت ها را به فضای مخصوص به رابطه ی  $k$  منتقل می کند. این انتقال به صورتی آموزش دیده می شود که جفت رابطه هایی که برقرار هستند، نسبت به جفت رابطه هایی که برقرار نیستند به یکدیگر نزدیکتر باشند.

در شکل ۱۲ مشاهده می شود که در صورتی که بین دو موجودیت  $i$  و  $j$  رابطه ی  $k$  برقرار باشد، پس از نگاشت این دو موجودیت توسط ماتریس های نگاشت  $A_k^o$  و  $A_k^s$  این دو موجودیت به یکدیگر نزدیکتر شده اند. همینطور در شکل ۱۳ مشاهده می شود که در صورتی که رابطه ی  $k'$  بین دو موجودیت  $i$  و  $j$  برقرار نباشد، پس از نگاشت این دو موجودیت توسط ماتریس های نگاشت  $A_k^o$  و  $A_k^s$  دو موجودیت در فاصله ی بیشتری از یکدیگر قرار می گیرند.



شکل ۱۲ افزایش فاصله ی دو موجودیت که رابطه ی  $k'$  بین آنها برقرار نیست پس از نگاشت توسط بردار انتقال  $k'$

در روش Structred Embedding



شکل ۱۳ کاهش فاصله ی دو موجودیت که رابطه ی  $k'$  بین آنها برقرار است پس از نگاشت توسط بردار انتقال  $k'$

در روش Structred Embedding

یکی از نکات منفی که در این روش به چشم می خورد یادگیری ماتریس های وزن دار جدا برای موجودیت های اول و دوم است، که باعث افزایش تعداد پارامتر مورد نیاز برای آموزش است.

## ۷-۵-۲- مدل $\text{TransE}^1$

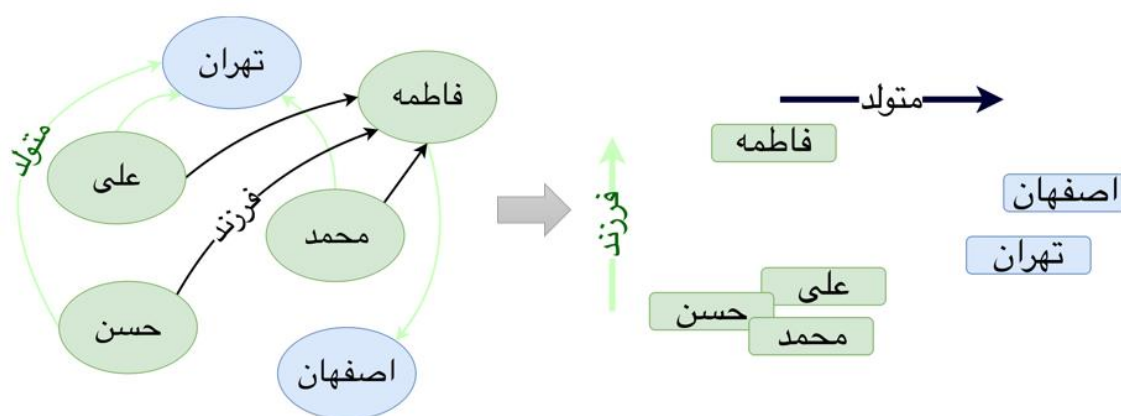
در [94] مدلی برای حل مشکل تعداد پارامتر زیاد در روش SE مطرح شده است که بجای اینکه از ماتریس های  $A_{k'o}$  و  $A_{k's}$  برای تاثیر دادن رابطه ها استفاده شود، رابطه را یک بردار همانند بردار موجودیت ها در نظر گرفته شده و به عنوان یک آفست در کنار موجودیت اول در محاسبات از آن استفاده می کند. [1] به طور خاص امتیاز حقیقت  $x_{ijk}$  از رابطه ی (۲-۱۴) بدست می آید.

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

<sup>1</sup> Translation Embedding

در این روش رابطه ها به صورت یک بردار انتقال استفاده می شوند، به این صورت که فقط روی موجودیت اول اعمال می شوند و در صورت وجود رابطه بین این دو موجودیت، موجودیت اول را به موجودیت دوم نزدیک می کنند. در شکل + یک گراف را مشاهده می کنیم که در آن روابط فرزندی و مکان تولد بین ۶ موجودیت نمایش داده شده است. در شکل ۱۳ یک مثال از اعمال روش TransE روی این گراف را مشاهده می کنیم. مشاهده می شود که موجودیت ها بردارهایی ثابت در نظر گرفته شده اند و بردارهای رابطه به صورت یک بردار انتقال به تصویر کشیده شده است که موجودیت ها را به هدف های مورد نظر نزدیک می کنند.

مثلا اگر «علی» را توسط بردار «فرزند» منتقل کنیم به موجودیت «فاطمه» نزدیک می شود که حقیقت (علی، فرزند، فاطمه) را تشکیل می دهد. اما اگر همین موجودیت را توسط بردار رابطه ی «متولد» منتقل کنیم، به موجودیت «تهران» نزدیک می شود که حقیقت (علی، متولد، تهران) را تشکیل می دهد.



شکل ۱۴ بازنمایی شبکه ی ناهمگون در فضای برداری

روش TransE این مساله را به صورت یک مساله ی کمینه سازی حل می کند به این صورت که سعی می کند فاصله ی مجموع بردارهای موجودیت اول و رابطه  $e_1 + r$  را با موجودیت دوم  $e_2$  کمینه کند.



همانطور که قبل تر گفتیم تابع امتیاز در این روش ها یک تابع فاصله است، پس در اینجا هدف کاهش فاصله بین  $e_1 + r$  و  $e_2$  است که بصورت (۲-۱۵) نوشته می شود.

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

برای رسیدن به نتیجه ی بهینه نوآوری دیگری که در این روش معرفی شده است این است که تابع کمینه سازی را به گونه ای تغییر داده است که فاصله ی نمونه های اشتباه را نیز زیاد کرده است.

در زمانی آموزش پارامترها علاوه بر در نظر گرفتن این نکته که باید تابع فاصله ی نمونه های مثبت موجود در پایگاه دانش کمینه شود، سعی شده تا فاصله ی نمونه های منفی را نیز افزایش دهد. از انجایی که پایگاه های دانش مورد استفاده از فرض جهان باز پیروی می کنند وقتی حقیقت  $(e_1.r.e_2)$  در پایگاه دانش موجود نیست نمی توانیم نتیجه بگیریم که این سه گانه غلط است و آن را نمونه ی منفی تلقی کنیم.

در این روش برای ساختن نمونه های منفی، نمونه های مثبت مورد استفاده قرار گرفته اند 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 نمونه ی مثبت دو نمونه ی منفی نیز تولید شده است (رابطه ی ۲-۱۶). در رابطه ی (۲-۱۷) مشاهده می شود که علاوه بر کاهش فاصله ی موجودیت های مثبت، یک جریمه هم برای نمونه های منفی در نظر گرفته شده است و همچنین یک حاشیه  $\gamma$  بین نمونه های مثبت و منفی اضافه شده است.

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

$$\sum_{pos} \sum_{neg \in S'} [\gamma + \|s + r - o\|_2^2 - \|s' + r - o'\|_2^2] \quad (17-2)$$

در این روش نیز مانند روش رسکال از SGD استفاده شده است که هم امکان آموزش به کمک نمونه برداری دسته ای<sup>۱</sup> را فراهم می کند و هم از مشکل گیر کردن در نقاط بهینه ی محلی جلوگیری می کند. الگوریتم کامل این روش در الگوریتم ۱ آمده است.

```

1: input: Training set  $S = \{(\text{sub}, \text{rel}, \text{obj})\}$ , margin  $\gamma$ , learning rate  $\lambda$ 
2: initialize  $\mathbf{r} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$  for each rel
3:        $\mathbf{r} \leftarrow \ell / \|\ell\|$  for each  $\ell$ 
4:        $\mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{j}}, \frac{6}{\sqrt{k}})$  for each entity ent(sub or obj)
5: loop
6:    $\mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$  for each entity ent
7:    $S_{batch} \leftarrow \text{sample}(S, b)$  //sample minibatch of size  $b$ 
8:    $T_{batch} \leftarrow \emptyset$  //initialize set of pairs
9:   for  $(\text{sub}, \text{rel}, \text{obj}) \in S_{batch}$  do
10:     $(\text{sub}', \text{rel}, \text{obj}') \leftarrow \text{sample}(S'(\text{sub}, \text{rel}, \text{obj}))$  //sample negative triplet
11:     $T_{batch} \leftarrow T_{batch} \cup \{((\text{sub}, \text{rel}, \text{obj}), (\text{sub}', \text{rel}, \text{obj}'))\}$ 
12:   end for
13:   Update embeddings w.r.t.  $\sum_{T_{batch}} \nabla [\gamma + \|\mathbf{s} + \mathbf{r} - \mathbf{o}\|_2^2 - \|\mathbf{s}' + \mathbf{r} - \mathbf{o}'\|_2^2]_+$ 
14: end loop

```

الگوریتم ۱ الگوریتم روش TransE

<sup>1</sup> Batch Sampling

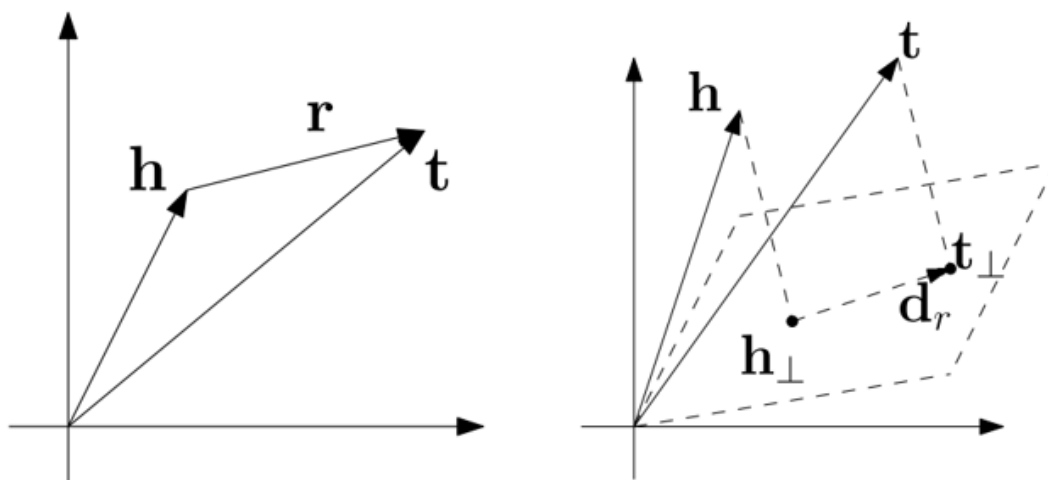
<sup>2</sup> local optimom

## ۸-۵-۲- مدل TransH

در بخش قبل مشاهده کردیم که TransE روش مناسبی را ارائه داد با تعداد پارامتر کم و مقیاس پذیر که قادر به آموزش مدل است. در [۱] روشی معرفی شده است که به بررسی برخی از نگاشته ها مانند یک به چند، چند به یک، چند به چند و انعکاسی پرداخته است. در این روش رابطه ها به صورت یک ابرصفحه و یک بردار انتقال روی آن در نظر گرفته می شود، از همین رو به این روش روش انتقال روی ابر صفحه یا به اختصار TransH گفته می شود.

در این روش موجودیت ها همچون روش قبل به صورت یک بردار از ویژگی های پنهان تعریف می شوند ولی رابطه ها به صورت دو بردار تعریف می شوند، یک بردار برای انتقال فضای مساله به ابرصفحه آن رابطه و دیگری بردار انتقالی است که در روش TransE نیز داشتیم و موجودیت های اول و دوم را به یکدیگر وصل می کرد. در روش قبل هدف این بود که دو موجودیت مشارکت کننده در یک حقیقت درست به وسیله ی 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 می شود که نگاشته ی از موجودیت های اول و دوم در آن قرار دارد. به کمک این روش ساده می توانیم نگاشته های یک به چند، چند به یک، چند به چند و انعکاسی را نیز در عملیات آموزش تاثیر دهیم در حالی که پیچیدگی و هزینه ی این روش TransE نداریم. با هزینه و پیچیدگی کمی مانند روش TransE، را نیز در نظر گرفت.

<sup>1</sup> Translating on Hyperplanes



شکل ۱۵-ب روش TransH

شکل ۱۶-الف روش TransE

همانطور که در شکل ۱۶-الف روش TransE مشاهده می شود روش TransE بردارهای  $h$  و  $t$  را به کمک بردار  $r$  به یکدیگر متصل می کرد اما در روش TransH همانطور که در شکل ۱۵-ب روش TransH نشان داده شده است ابتداء بردارهای  $h$  و  $t$  به صفحه ای مربوط به رابطه ای مورد نظر منتقل شده اند و توسط بردار  $d_r$  که بردار رابطه است به یکدیگر متصل می شوند.

مقادیر  $h_{\perp}$  و  $t_{\perp}$  به صورت معادله ای (۲-۱۹) بدست می آیند که در آن بردار انتقال به ابرصفحه ای مربوط به رابطه ای  $r$  است و  $d_r$  معادل بردار  $r$  در ابرصفحه ای رابطه است.

$$h_{\perp} = h - w_r^T h w_r. \quad t_{\perp} = t - w_r^T t w_r \quad (۲-۱۸)$$

پس تابع امتیاز این روش به صورت زیر خواهد بود:

$$f_r(h, t) = ||(h - w_r^T h w_r) + d_r - (t - w_r^T t w_r)||_2^2 \quad (۲-۱۹)$$

در این روش نیز مانند روش TransE برای کمینه کردن خطا از حقیقت های صحیح و غلط و یک فاصله بین آن ها استفاده می شود که قصد دارد فاصله ی بین بردار  $h + r$  حقیقت های صحیح را با  $t$  کم و همین فاصله را در حقیقت های غلط زیاد کند. انتخاب حقیقت های غلط در این روش بمانند روش قبل است اما در این روش بجای استفاده از تابع امتیاز  $f_r$  که در معادله ی (۲-۱۷) مطرح کردیم از تابع امتیاز  $f_r$  که در (۲-۱۹) معرفی کردیم استفاده می کنیم و تابع کمینه سازی به صورت (۲-۲۰) خواهد شد.

$$f_r = \sum_{(h.r.t) \in \Delta} \sum_{(h'.r'.t') \in \Delta'_{(h.r.t)}} [f_r(h.t) + \gamma - f_{r'}(h'.t')]_+ \quad (2-20)$$

## فصل ۳: قوانین انجمنی

در این فصل به معرفی قوانین انجمنی پرداخته و توضیح می دهیم چگونه از این قوانین برای بهبود مدل هایی که در فصل قبل معرفی شد استفاده می کنیم. در ابتدا برای سنجش کیفیت قوانین انجمنی معیارهای اطمینان را تعریف می کنیم، در ادامه نحوه ی استخراج این قوانین از پایگاه دانش را توضیح می دهیم و در انتها نیز قوانین هدفی که قصد استفاده از آن ها برای بهبود مدل داریم را مشخص می کنیم.

### ۱-۳- معیارهای اطمینان

برای اینکه بتوانیم به قانون هایی که استخراج کرده ایم اعتماد کنیم و از آن ها در آموزش مدل ها استفاده کنیم، نیاز به یک معیار اطمینان داریم. معیارهای رایج برای اندازه گیری دقت یک قانون میزان پشتیبانی قانون<sup>۱</sup>، پوشش سر<sup>۲</sup> و معیار اطمینان استاندارد<sup>۳</sup> است.

---

<sup>۱</sup> support rule

<sup>۲</sup> head coverage

<sup>۳</sup> Standard confidence

## ۱-۳- معیار پشتیبانی قانون:

این معیار برای میزان اهمیت یک قانون استفاده می شود و به این صورت تعریف می شود: تعداد قوانین یافت شده در پایگاه دانش که یک حقیقت را نتیجه می دهند. برای مثال در (۱-۳) قانون R معرفی شده است که مشخص می کند هر شخص x که در شهر y زندگی می کند، در آن شهر متولد شده است. به تعداد دفعاتی که این دو حقیقت در پایگاه دانش اتفاق بیافتد و قانون R را نقض نکند، پشتیبان این قانون گفته می شود.

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

این معیار به صورت رابطه ی (۲۱-۳) تعریف می شود.

$$\text{supp}(\vec{B} \Rightarrow r(x, y)) := \#(x, y) : \exists z_1, \dots, z_m : \vec{B} \wedge r(x, y) \quad (۲-۳)$$

در رابطه ی (۲-۳)، B مجموعه ای از حقیقت ها است که حقیقت  $r(x, y)$  را نتیجه می دهد، به تعداد باری که این اتفاق می افتد معیار پشتیبانی قانون گفته می شود. در جدول ۲ نمونه ی پایگاه دانش متشکل از دو رابطه ی پایگاه دانش متشکل از دو رابطه و ۵ حقیقت وجود دارد. قانون (۱-۳) 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 می کنند [AMIE].

جدول ۲ نمونه ی پایگاه دانش متشکل از دو رابطه

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

## ۲-۱-۳- معیار پوشش سر

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

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

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

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

## ۳-۱-۳- معیار اطمینان استاندارد

معیارهای قبل اهمیت قانون را بررسی می کردند و فقط پیش بینی های درست از قانون را در نظر می گرفتند، و پیش بینی های غلطی که این قانون تولید می کند را در نظر نمی گیرند. 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 بدنه ی این قانون در پایگاه دانش دیده شده است تقسیم می کنیم. این کار باعث می شود که معیاری داشته باشیم برای اینکه بدانیم در



چند درصد مواقع که قسمت بدنه ی این قانون برقرار بوده است منجر به تولید قسمت نتیجه شده است.

$$conf(\vec{B} \Rightarrow r(x, y)) := \frac{supp(\vec{B} \Rightarrow r(x, y))}{\#(x, y) : \exists z_1, \dots, z_m : \vec{B}} \quad (5-3)$$

رابطه ی بالا به زبان ساده برابر است با تعداد رخداد قانون بخش بر تعداد دفعاتی که می توانست رخ دهد. برای مثال در پایگاه دانش معرفی شده در جدول ۲ معیار اطمینان استاندارد برابر  $\frac{1}{3}$  است. زیرا  $wasBornIn(Adam, Paris)$  به عنوان نمونه ی مثبت<sup>۱</sup> در نظر گرفته شده و  $wasBornIn(Adam, Rome)$  و  $wasBorn(Bob, Zurich)$  که در پایگاه دانش وجود ندارند نمونه ی منفی<sup>۲</sup> در نظر گرفته شده است. که معیار اطمینان استاندارد برابر می شود با تعداد نمونه ی مثبت یعنی ۱ بخش بر تعداد کل نمونه ها یعنی ۳.

این معیار در پایگاه های دانش فرض جهان بسته معیار خوبی است و به خوبی دقت قانون استخراج شده را مشخص می کند زیرا همان طور که در مثال بالاتر هم مشاهده کردیم، عدم وجود یک نمونه ی مثبت از حقیقت در پایگاه دانش دلیلی بر غلط بودن آن حقیقت بوده است. اما همانطور که در بخش ۳-۱-۲ اشاره کردیم فضای این مساله فرض جهان باز است و حقیقت هایی که در این پایگاه های دانش وجود ندارند لزوماً غلط نیستند [AMIE]. برای درک بیشتر این موضوع به مثال زیر دقت کنید:

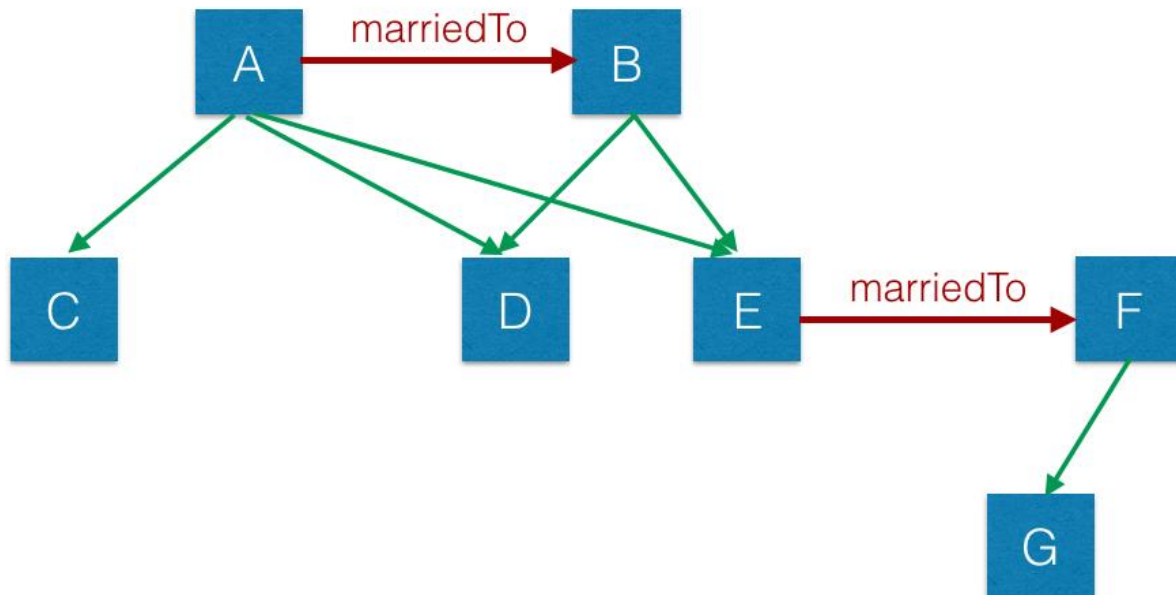
به گراف ارائه شده در شکل ۱۷ دقت کنید. در این گراف رابطه های افقی رابطه ی ازدواج (marriedTo) و روابط عمودی رابطه ی داشتن فرزند (hasChild) را مشخص می کنند. در این گراف قصد داریم که قانون R با تعریف زیر را بررسی کنیم:

$$hasChild(y.x).marriedTo(y.z) \Rightarrow hasChild(z.x) \quad (6-3)$$

<sup>1</sup> positive example

<sup>2</sup> negative example

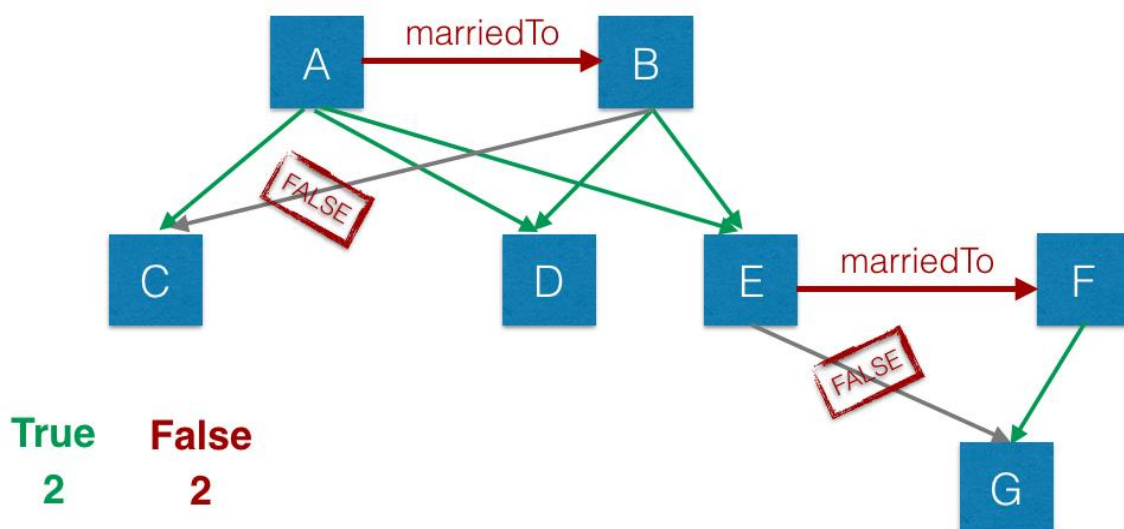
این قانون به این معناست که اگر شخص Y فرزندی به نام X داشته باشد و همین شخص با شخص دیگری به نام Z ازدواج کرده باشد، میتوان نتیجه گرفت که شخص Z هم فرزندی به نام X دارد.



شکل ۱۷ نمونه گراف ناهمگون با دو نوع رابطه

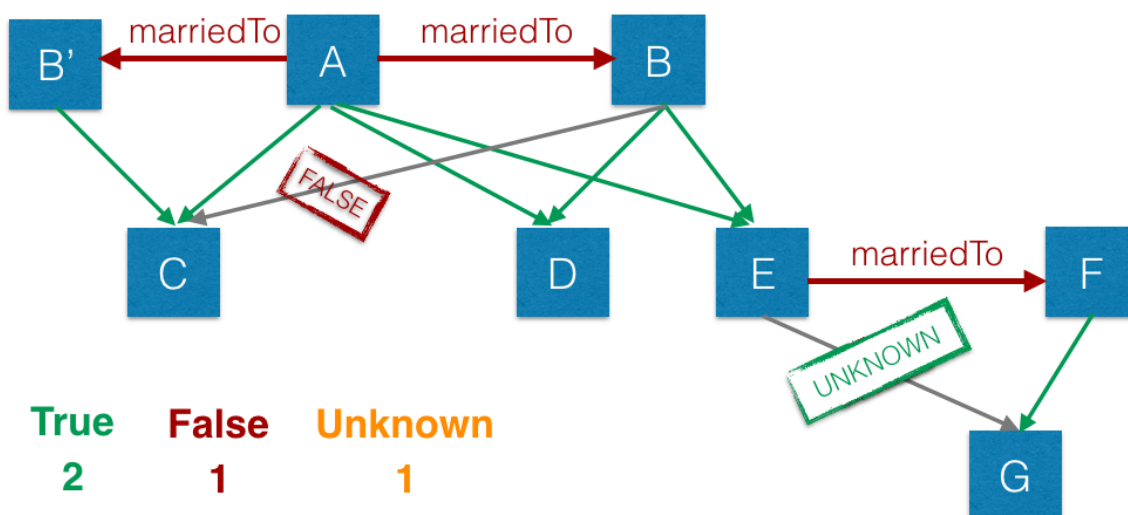
اگر بخواهیم معیار اطمینان استاندارد در این گراف را بررسی کنیم، مشاهده می شود که این قانون ۲ بار در این گراف صدق کرده است در حالی که می توانست ۴ بار اتفاق بیافتد. در شکل ۱۸ دفعاتی که این قانون باید اتفاق می افتاده است و اتفاق نیافتاده است نمایش داده شده و به عنوان نمونه ی منفی در نظر گرفته شده است. با این اوصاف معیار اطمینان استاندارد قانون R در گراف معرفی شده برابر ۰,۵ می شود.

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



شکل ۱۸ حقیقت هایی که در گراف نمونه موجود نیست

اما همانطور که توضیح دادیم عدم وجود حقایق در پایگاه های دانش فرض جهان باز دلیل بر غلط بودن این حقیقت ها نیست. برای مثال فرض کنید که در گراف معرفی شده یک گرهی  $B'$  داشته باشیم که رابطه ای ازدواج داشته باشد با گرهی  $A$  و رابطه ای فرزندی داشته باشد با گرهی  $C$  پس می دانیم که رابطه ای فرزندی بین گرهی  $B$  و  $C$  برقرار نیست اما در مورد وجود و عدم وجود این رابطه بین  $E$  و  $G$  اطلاعی نداشته باشیم (شکل ۱۹).



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

پس معیار اطمینان استاندارد برای پایگاه های دانش جهان باز معیار مناسبی نیست و نیاز به تعریف معیار بهتری داریم. در [AMIE] معیار اطمینان با فرض نیمه کامل<sup>۱</sup> (PCA-Confidence) را معرفی کرده است.

#### ۴-۱-۳- معیار اطمینان با فرض نیمه کامل

در این معیار سعی شده نمونه هایی که در پایگاه دانش وجود ندارند و در معیار اطمینان استاندارد آنها را نمونه ی غلط در نظر می گرفتیم 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، بر سی می شود که  $y'$  وجود دارد که برای آن حقیقت  $r(x,y')$  برقرار باشد یا خیر، اگر وجود داشته باشد در نظر می گیرد که  $r(x,y)$  غلط بوده و در غیر این صورت این حقیقت را ناشناخته فرض می کند.

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

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

یا در گرافی که در شکل ۱۹ معرفی شده است، در معیار اطمینان استاندارد هر دو رابطه ی بین  $(B,C)$  و  $(E,G)$  را غلط در نظر گرفتیم، اما در اینجا داریم که  $C$  مادری به نام  $B'$  دارد و می توانیم مطمئن شویم که رابطه ی فرزندی بین  $B$  و  $C$  برقرار نیست و این رابطه را غلط در نظر بگیریم. اما برای رد رابطه ی  $E$  و  $G$  هیچ مدرکی نداریم و این رابطه را غلط در نظر نمی گیریم بلکه ناشناخته در نظر

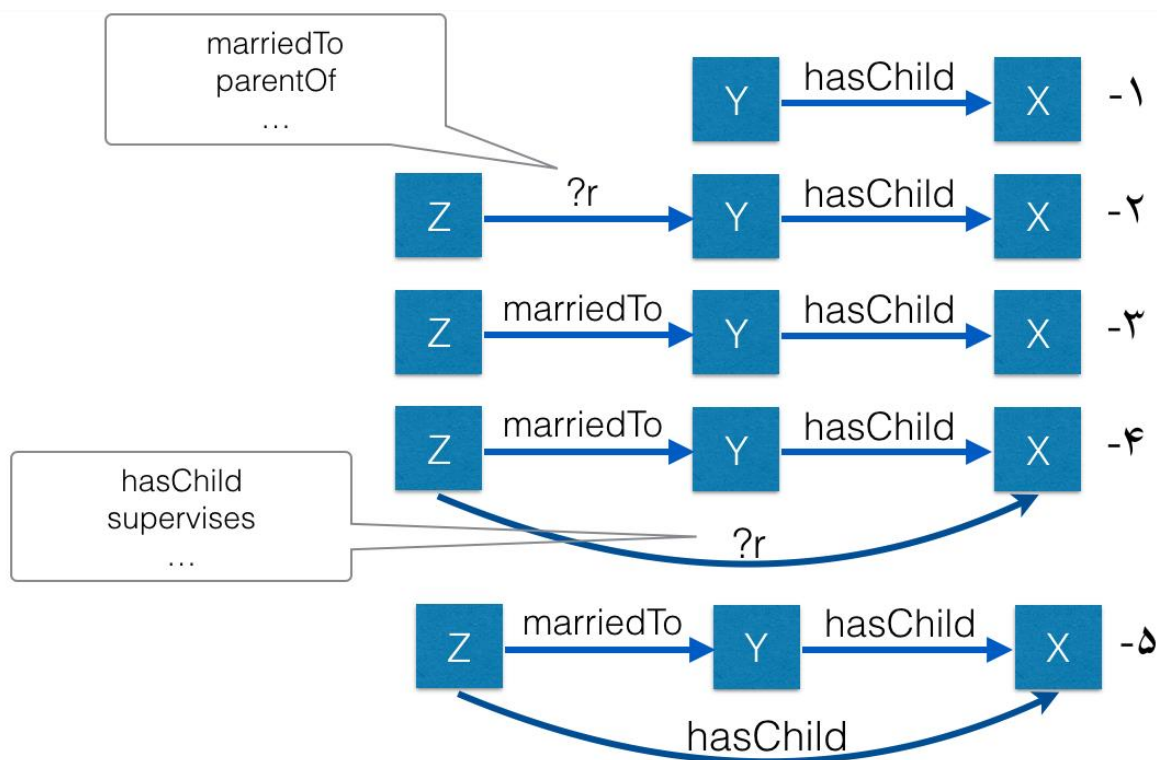
<sup>1</sup> Partial Completeness Assumption

می گیریم و از آن در فرومول PCA-confidence استفاده نمی کنیم. پس PCA-Confidence در این مثال برابر با X است.

## ۳-۲- الگوریتم ها

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

نحوه ی استخراج قوانین در شکل ۲۰ نشان داده شده است که هر مرحله را در ادامه توضیح خواهیم داد.



شکل ۲۰ نحوه ی استخراج قوانین انجمنی از حقیقت های پایگاه دانش

به ازای هر حقیقت این روند یک بار تکرار می شود:

- ۱- حقیقت انتخاب شده را در نظر می گیریم. در این مثال حقیقت (Y, hasChild, X) در نظر گرفته شده است.
  - ۲- تمامی روابطی که بین یکی از موجودیت ها و موجودیت دیگری وجود دارد را بررسی می کنیم. در این مثال روابط موجود برای موجودیت اول یعنی Y نمایش داده شده است، برای مثال روابط ازدواج و والد بودن با این موجودیت مورد بررسی قرار گرفته است.
  - ۳- همه ی روابطی که در مرحله ی ۲ کاندید شده بودن را در کنار Y قرار می دهیم و یک زنجیر از قوانین را ایجاد می کنیم. زنجیر ایجاد شده در این قسمت به صورت رابطه ی ازدواج بین Z و Y و رابطه ی داشتن فرزند بین Y و X است.
  - ۴- در این قسمت همه ی روابطی که بین موجودیت اضافه شده به زنجیر و موجودیت اول را بررسی می کنیم یعنی روابط بین Z و X. روابط کاندید در این مثال روابط “داشتن فرزند” و “ناظر بودن” است.
  - ۵- در این مرحله روابطی که کاندید شده اند را بررسی می کنیم و معیارهای اطمینان و اهمیت که در بخش قبل معرفی کردیم را برای آن ها بدست می آوریم، در صورتی که به معیارها مقدار قابل قبولی داشته باشند قانون تولید شده را ذخیره می کنیم و در غیر این صورت از آن رد می شویم.
  - ۶- پس از بررسی همه ی روابط کاندید تولید شده در بخش ۴ کل زنجیر تولید شده را به مرحله ی ۱ ارسال می کنیم و با اضافه کردن یک حقیقت دیگر به ابتدا یا انتهای آن روند رشد زنجیر را تا آستانه ی تعریف شده ادامه می دهیم.
- به کمک الگوریتم معرفی شده در بالا همه ی قانون های ممکن در پایگاه دانش با توجه به معیارهای کیفیت و اهمیت استخراج خواهند شد. اما با توجه به اندازه ی بزرگ پایگاه های دانش مورد استفاده این روش برای بررسی همه ی حالت های ممکن بسیار وقت گیر خواهد بود، پس بهتر است در

مرحله ی ۲ و ۴ که به انتخاب کاندید می پردازیم بجای بررسی همه ی حقیقت های موجود در پایگاه داده یک عملیات هرس<sup>۱</sup> روی کاندیدها با توجه به همرخدای با حقیقت موجود انجام شود و همه ی حقیقت ها مورد بررسی قرار نگیرند.

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

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

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



?a /soccer/current\_team/team ?b

در مثال دیگر داریم که اگر بازیگر سینمای **a** جایزه ی **b** را برنده شده باشد، نتیجه می گیریم که این بازیگر برای جایزه ی **b** نامزد شده است.

?a /award/awards\_won ?b



?a /award/award\_nomination/nominated\_for ?b

---

<sup>1</sup> Pruning

قوانین پیچیده تری نیز استخراج شده است که بدنه ی قانون<sup>۱</sup> از چند حقیقت تشکیل شده است که یک نمونه از این قوانین به صورت زیر است. در این قانون داریم که اگر a یک موسسه ی آموزشی باشد که در مکان f قرار دارد و داشته باشیم که f در مکان b واقع شده است، می توانیم نتیجه بگیریم که موسسه ی آموزشی a نیز در مکان b قرار دارد. علی رقم پیچیدگی ای که این قانون نسبت به دو قانون قبلی دارد، اما معیار اطمینان PCA این قانون ۰,۹۳ بوده و بسیار قابل اعتماد است و می توانیم از آن در آموزش مدل ها 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 مدل های مبتنی بر ویژگی های پنهان استفاده خواهیم کرد.

### ۳-۳- قوانین هدف

در بخش قبل مشاهده شد که چگونه قوانین از پایگاه های دانش استخراج می شوند و چند نمونه از این قوانین را مرور کردیم. در این بخش دسته بندی از قوانینی که استخراج می شوند را معرفی می کنیم و در ادامه بررسی های انجام شده روی روش های موجود را روی این دسته بندی ها اعمال می کنیم و نقاط قوت و ضعف هر روش را نسبت به این دسته بندی ها می سنجیم.

<sup>1</sup> rule body



## ۱-۳-۳- R-subsumption قانون

این قانون به صورت زیر تعریف می شود، به این معنی که اگر  $x$  و  $y$  رابطه ی  $r$  را باهم داشته باشند، رابطه ی  $r'$  نیز بین آن ها برقرار است.

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

برای مثال اگر در پایگاه دانش رابطه ی پدر بودن را داشته باشیم، می توانیم از آن رابطه ی والد بودن را نتیجه بگیریم، مثلاً اگر حقیقت «علی، پدر، حسن» به معنی «علی پدر حسن است» را داشته باشیم، می توانیم نتیجه بگیریم که حقیقت «علی، والد، حسن» نیز حقیقت درستی است.

## ۲-۳-۳- R-equivalence قانون

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

$$r(x, y) \Leftrightarrow r'(x, y)$$

این قانون در آموزش مدل ها کمک بسیاری می تواند بکند زیرا ماهیت این قانون به این صورت است که روابط یکسان (روابطی که به دو صورت در پایگاه دانش استفاده شده اند اما یک معنی می دهند) را شناسایی می کند، برای نمونه مثالی که در بخش قبل زدیم، دو رابطه ی `located_in` و `contained_by` در توالی یکدیگر آمده بودند که یک معنی را می دهند، پس می توانیم نتیجه بگیریم که اگر حقیقت  $(x, \text{located\_in}, y)$  را داشته باشیم، حقیقت  $(x, \text{contained\_by}, y)$  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 می شود.

## ۳-۳-۳ قانون 2-hope translation

این قانون، همان قانون تعدی است، به این صورت که اگر  $X$  و  $Y$  رابطه ی  $r_1$  را باهم داشته باشند و همچنین  $Y$  و  $Z$  نیز رابطه ی  $r_2$  را داشته باشند، می توانیم نتیجه بگیریم که  $X$  و  $Z$  نیز با یکدیگر در ارتباطند.

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

نمونه ای از این قانون را در مورد مناطق جغرافیایی در بخش قبل مشاهده کردیم. مثلاً می دانیم که شهر تهران در کشور ایران قرار دارد، و کشور ایران در منطقه ی خاورمیانه قرار دارد، پس می توانیم نتیجه بگیریم که شهر تهران در منطقه ی خاورمیانه قرار دارد.

## ۳-۳-۴ قانون Triangle alignment

این قانون نیز مانند قانون قبلی در سمت بدنه ی قانون دو حقیقت را بررسی می کند و از روی آن ها حقیقت جدید را نتیجه می گیرد.

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

مشاهده می شود که در این قانون اشتراک موجودیت های دوم مد نظر قرار گرفته است و در صورتی که دو حقیقت در موجودیت دوم مشترک باشند بررسی می شود که بین موجودیت های اول نیز رابطه ای 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، می توان نتیجه گرفت که  $married(x, y)$  یعنی  $X$  و  $Y$  زن و شوهر هستند.

## ۵-۳-۳- قانون Specific R-subsumption

این قانون حالت کامل تری از قانون اول یعنی R-sub است به این صورت که بجز بررسی رابطه ها، ویژگی های موجودیت اول را نیز بررسی می کند. برای مثال در قانون اول داشتیم اگر شخصی پدر X باشد می توان نتیجه گرفت که آن شخص والد X نیز هست، اما عکس این قضیه را نمی توان نتیجه گرفت، یعنی نمی توان 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 می شود.

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

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

در جدول 3 قوانین معرفی شده به اختصار آمده اند.

قوانین انجمنی هدف 3 جدول

Body rule		Target rule	name
$r(x, y)$	$\Rightarrow$	$r'(x, y)$	R-subsumption
$r(x, y)$	$\Leftrightarrow$	$r'(x, y)$	R-equivalence
$r_1(x, y), r_2(y, z)$	$\Rightarrow$	$r'(x, z)$	2-hope translation
$r(x, z), r(y, z)$	$\Rightarrow$	$r'(x, y)$	Triangle alignment
$r_1(x, y), r_2(x, v)$	$\Rightarrow$	$r'(x, y)$	Specific R-sub

## ۴-۳- جمع بندی

متن

## فصل ۴: روش پیشنهادی

متن

### ۴-۱- چالش ها

متن

### ۴-۲- عملکرد روش های موجود

در این بخش نتایج بدست آمده از روش های معرفی شده در بخش قبل را مورد بحث و بررسی قرار می دهیم. روش هایی که در این پایان نامه مورد بررسی و بهبود قرار گرفته اند ۱- روش RESCAL از روش های دوخطی ۲- روش NTN از روش های ادراک چندلایه ای ۳- روش SE ۴- روش TransE و ۵- روش TransH از روش های فاصله ی پنهان هستند.

در ادامه آزمایشاتی روی این ۵ روش انجام شده است و این روش ها از نظر مقیاس پذیری و میزان کیفیت خروجی که به کمک معیارهای رتبه ی میانگین و  $hit@10$  محاسبه می شود مورد بررسی قرار گرفته اند. همه ی این آزمایشات روی پایگاه دانش Freebase15k انجام شده است که شامل حدود ۱۵ هزار موجودیت، ۱۳۰۰ رابطه و در مجموع ۶۰۰ هزار حقیقت است. تقسیم بندی داده های آموزش، ارزیابی و آزمون به شکلی که در بخش ۳-۲ توضیح داده شد انجام شده است.

### ۴-۲-۱- روش رسکال

**تعداد پارامترها:** در این روش برای هر موجودیت یک بردار  $d$  بعدی و برای هر رابطه یک ماتریس وزن دار  $d*d$  بعدی در نظر گرفتیم. پس تعداد پارامترهایی که در این روش نیاز است که آموزش دهیم از مرتبه  $O(n_e d + n_r d^2)$  است. در آزمایشات تعداد ابعادی که بهترین نتیجه را بدست می دهد  $d=250$  است. اگر ابعاد مساله را ۲۵۰ در نظر بگیر باید به تعداد ۸۸ میلیون پارامتر آموزش داده شود

$$(n_e d + n_r d^2 = 15k * 250 + 1.3k * 250 * 250 = 88m)$$

**کیفیت خروجی:** پس از اعمال مدل آموزش دیده ی رسکال روی مجموعه داده ی آزمون ۵۰ هزار حقیقی، میزان ۴۲,۱٪ پاسخ های داده شده توسط این مدل زیر رتبه ی ۱۰ بوده ( $hit@10$ ) و میانگین رتبه ی کل پاسخ هایی که داده شده ( $mean rank$ ) ۶۸۳ است.

### ۴-۲-۲- روش NTN

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

کیفیت خروجی: مقدار  $hit@10$  در این روش ۲۷٪ و میانگین رتبه ی همه ی پاسخ های این روش ۱۶۴ است. همانطور که در معرفی این روش گفته شد این روش به سرعت به بیش برآزش میل می کند 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 می دهد روش رسکال برای سوالاتی که به میزان کافی نمونه ی آموزش ندیده است و نتایج خیلی پرتی بدست می دهد ولی برای ۴۲,۱٪ حقیقت که روابط بین آن ها به خوبی آموزش دیده شده است و پاسخ های درست در ۱۰ پاسخ اول بوده است.

### ۳-۲-۴- روش Structured Embedding

**تعداد پارامتر:** در این روش برای هر موجودیت یک بردار  $d$  بعدی و برای هر رابطه دو ماتریس  $d*d$  بعدی آموزش دیده می شود، یکی برای انتقال موجودیت اول و دیگری برای انتقال موجودیت دوم. بنابراین تعداد پارامتر این روش از مرتبه ی  $O(n_e d + 2n_r d^2)$  است. برای آموزش این مدل نیز بردارهایی با اندازه  $d=50$  کفایت می کند و برای آموزش این مدل باید حدود ۸ میلیون 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 می شود که علی رقم کاهش بسیار زیادی که در تعداد پارامترهای مساله نسبت به دو روش قبل داشته ایم، نتیجه ی خوبی حاصل شده است و در معیار  $hit@10$  با روش رسکال و در معیار میانگین رتبه با روش NTN رقابت می کند.

## TransE - ۴-۲-۴

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

**کیفیت خروجی:** این روش علی رقم کاهش چشم گیری که در تعداد پارامترها داشت و مقیاس پذیری بسیار بالایی که ایجاد کرده است، در نتایج بدست آمده نیز بهتر از روش های معرفی شده ی قبلی کار کرده است. در این روش  $hit@10$  برابر ۴۵,۱٪ و میانگین رتبه برابر ۱۲۵ است که مشاهده می شود در این روش هم جواب های درست بسیار بیشتر بوده و هم داده های پرت بسیار کمتر شده است و جواب سوال ها به جواب های منطقی نزدیک تر شده است.

## TransH - ۴-۲-۵

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



ابرفصلی مربوط به رابطه را بیشتر دارد که به ازای هر رابطه  $k$  پارامتر به عملیات آموزش اضافه می کند، پس پارامترهایی که باید آموزش دیده شوند از مرتبه  $O(n_k + 2n_r k)$  هستند. تعداد پارامتر این روش روی پایگاه دانش freebase15k حدود 880 خواهد بود  $(n_d + 2n_r d = 15k * 50 + 880k = 880k * 1.3 * 2)$ . مشاهده می شود که این روش افزایش چشمگیری در تعداد پارامترهای مساله نداشته اما نتایج بسیار دقیق تری بدست آورده است.

**کیفیت خروجی:** این روش با در نظر گرفتن روابط پیچیده تری مانند روابط یک به چند و چند به چند و چند به یک و بازگشتی توانسته است که دقت بهتری نسبت به روش های نام برده شده ی قبلی به دست آورد و بدون افزایش غیر منطقی تعداد پارامترها نسبت به روش TransE به مقدار  $\text{Hit}@10$  برابر با ۶۴,۴٪ و میانگین رتبه ی ۸۷ برسد.

### ۳-۴- نقاط قوت و ضعف روش های موجود

در این بخش به بررسی نقاط قوت و نقاط ضعف روش های موجود می پردازیم. در بخش ۲-۴- به تفصیل در مورد مقیاس پذیری و معیارهای کیفیت روش ها صحبت شد، در این بخش میزان همبستگی و ارتباط موجودیت ها و رابطه ها را بررسی می کنیم که چه مقدار این همبستگی و ارتباط ها در مدل های آموزش دیده شده درک شده است.

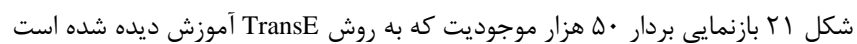
روش هایی که معرفی شدند در بهترین حالت تا ۶۴,۴٪ دقت در یافتن پاسخ در ۱۰ جواب اول ( $\text{hit}@10$ ) را داشتند. نمونه هایی از این سوال و ۱۰ جواب اول مدل TransE برای آن را در جدول ۴ مشاهده می کنیم.

جدول ۴ نمونه ی سوال از مدل TransE و ده پاسخ اول

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

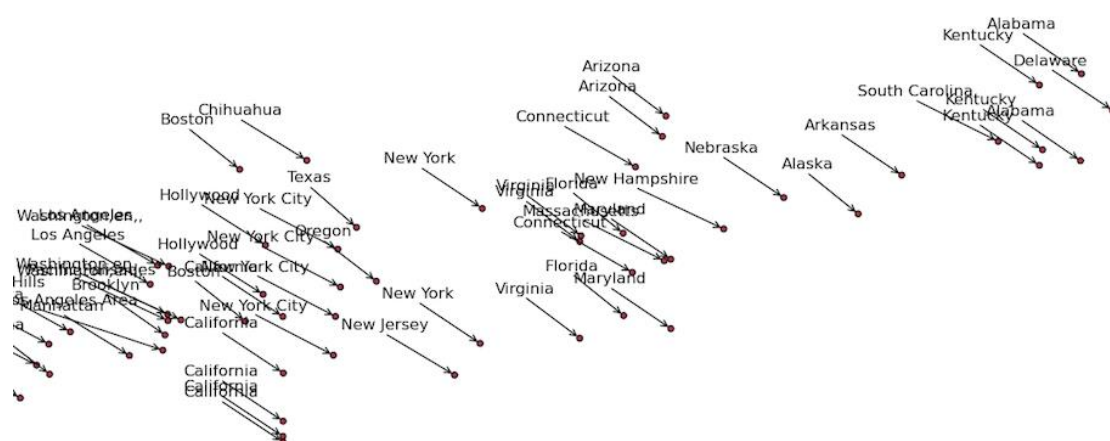
همانطور که مشاهده می شود جواب هایی که داده شده است همبستگی خوبی دارند، برای مثال در نمونه ی اول که از مدل پرسیده شده است Lil Wayne که یک خواننده ی آمریکایی است متولد کجاست، تمام ۱۰ جواب اول همانطور که انتظار می رود ایالت های مختلف آمریکا هستند و پاسخ اول یعنی New Orleans جواب درست می باشد و جواب ها منطقی 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، جواب ها همبستگی خوبی دارند و جواب پرت داخل آن ها نیست.

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



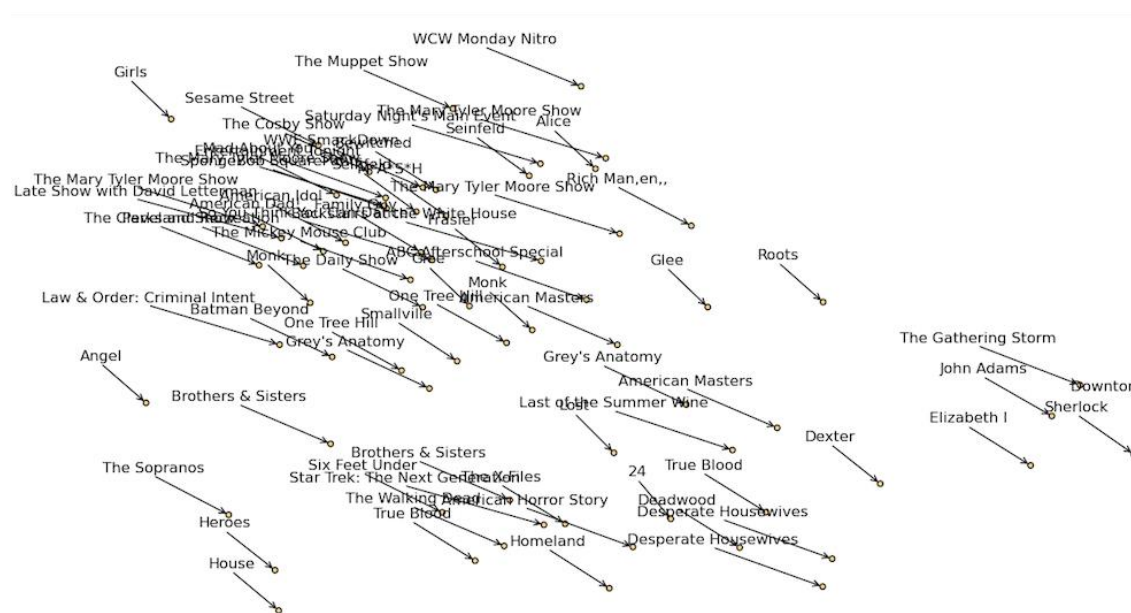
برای مطمئن شدن از این فرض چند جزیره را در شکل ۲۲ و شکل ۲۳ و شکل ۲۴ بزرگ‌نمایی کرده‌ایم.





شکل ۲۳ بازنمایی قسمتی از موجودیت ها که در مورد ایالت های آمریکا هستند

در شکل ۲۳ مشاهده می شود که ایالت های آمریکا یک جزیره را تشکیل داده اند، مانند آریزونا، کالیفرنیا، تگزاس و ...



شکل ۲۴ بازنمایی قسمتی از موجودیت ها که در مورد نام سریال های تلویزیونی آمریکا هستند

در شکل ۲۴ مشاهده می شود که نام سریال های تلویزیونی یک جزیره را تشکیل داده اند.

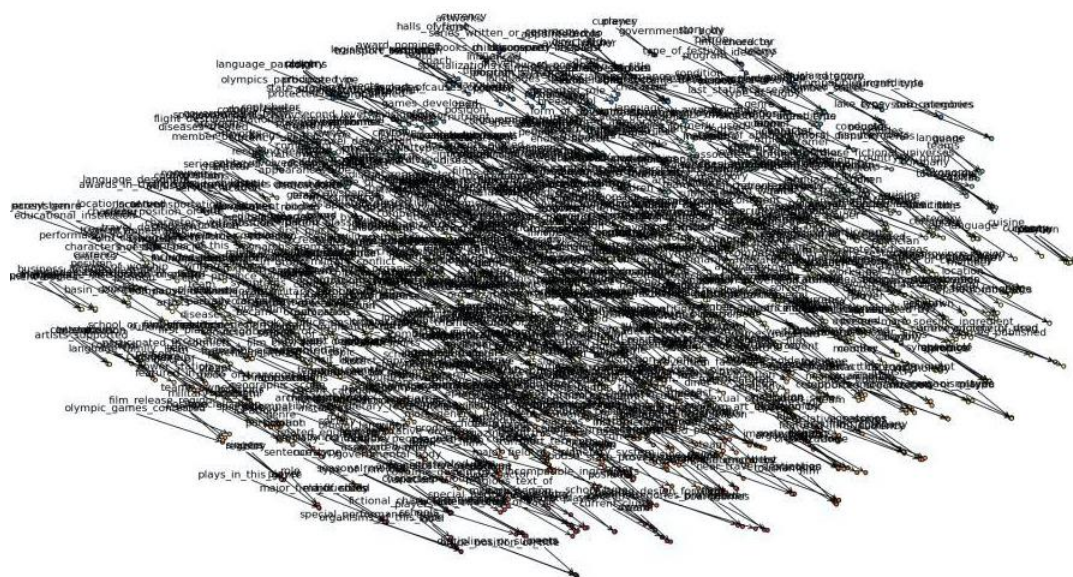
در مثال هایی که گفته شد، مشخص است که رابطه ی بین موجودیت ها به خوبی در این روش ها درک شده است و یک همبستگی بین بردارهای آموزش دیده شده وجود دارد.

می دانیم که رابطه ها هم مانند موجودیت ها می توانند ارتباطات و همبستگی های خود را داشته باشند. برای مثال رابطه ی “پدر بودن” نزدیک تر است به رابطه ی “فرزندی” نسبت به رابطه ی “نویسنده ی کتاب” بودن و انتظار داریم بردارها و ماتریس های آموزش دیده شده برای رابطه ها این همبستگی ها را درک کرده باشد و در فاصله ی کمی از یکدیگر قرار گرفته باشند و همان دسته بندی و جزیره شدنی که برای موجودیت ها اتفاق افتاده بود را اینجا نیز شاهد باشیم.

حال همین مساله را در بردارهای آموزش دیده شده بررسی می کنیم. برای بررسی این امر در شکل ۲۵ بردارهای همه رابطه های استفاده شده در Freebase15k را در دو بعد به نمایش در آورده ایم. همانطور که در شکل مشخص است، برای رابطه ها اتفاق جزیره ای شدن که در بازنمایی موجودیت ها اتفاق افتاده بود، به وجود نیامده است و بردارها روی صفحه بخش<sup>۱</sup> است. به نظر می رسد که در بردارهای آموزش دیده برای رابطه ها این همبستگی و ارتباط وجود ندارد.

---

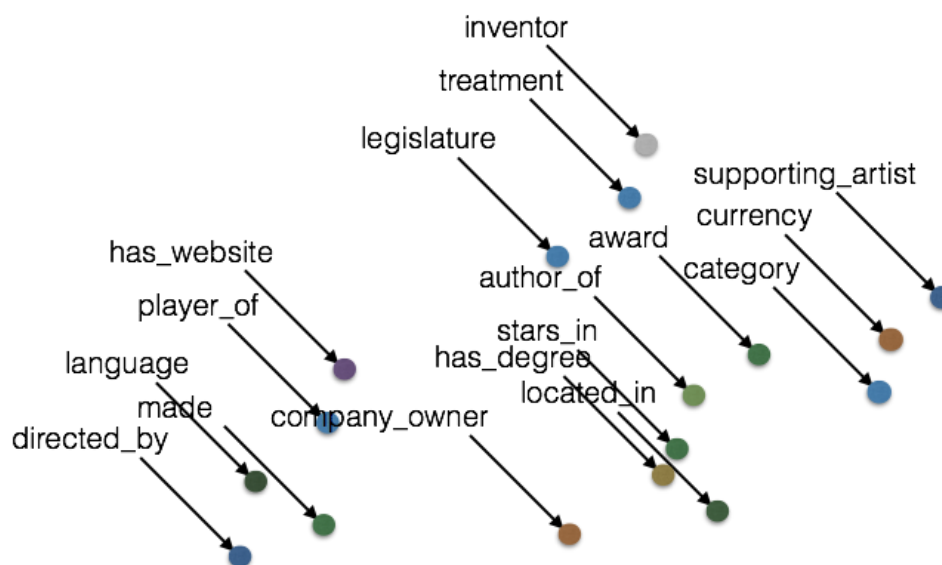
<sup>1</sup> Scatterd



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

برای بررسی بیشتر این موضوع چند قسمت از بازنمایی این بردارها را در شکل ۲۶ بزرگنمایی کرده ایم. همانطور که پیش بینی کرده بودیم رابطه هایی که در کنار هم قرار گرفته اند هیچ ارتباط و همبستگی ای با یکدیگر ندارند و کاملاً در صفحه پخش شده اند و به نظر می رسد که این مدل ها نتوانسته اند در این قسمت قوی عمل کنند و رابطه ها را درک کنند. در فصل بعد روی این مساله تمرکز می کنیم و سعی می کنیم به مساله را در آموزش مدل ها دخیل کنیم و همبستگی بین رابطه ها را نیز به مدل بفهمانیم و انتظار داریم نتیجه ی بهتری در معیارهای کیفیت بگیریم.





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

#### ۴-۴- بررسی عمل کرد روش ها در یک نگاه

در این قسمت روش های موجود را ...

در جدول ۵ تعداد پارامترهایی که هر یک از روش های بالا برای آموزش دیدن نیاز دارند، آورده شده است، همانطور که مشاهده می شود روش های Rescal و NTN تعداد پارامتر خیلی زیادی باید آموزش دهند و به مشکل مقیاس پذیری بر خواهند خورد و روی پایگاه دانش های بزرگ خیلی کند خواهند بود. در مقابل روش های SE و TransE و TransH با ایده هایی که برای کاهش تعداد پارامترهای مساله پیاده سازی کردن این تعداد را بسیار کاهش داده و مقیاس پذیری خوبی دارند، به صورتی که مدل TransE روی پایگاه دانش Freebase 1M در مدت تقریباً یک روز با  $hit@10$  برابر با ۳۴٪ به جواب رسیده است در صورتی که روش های RESCAL و NTN روی این پایگاه دانش به جواب نمی رسند، که این موضوع بر عدم مقیاس پذیری روش های RESCAL و NTN تاکید می کند.



جدول ۵ تعداد پارامترهای هر روش

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)

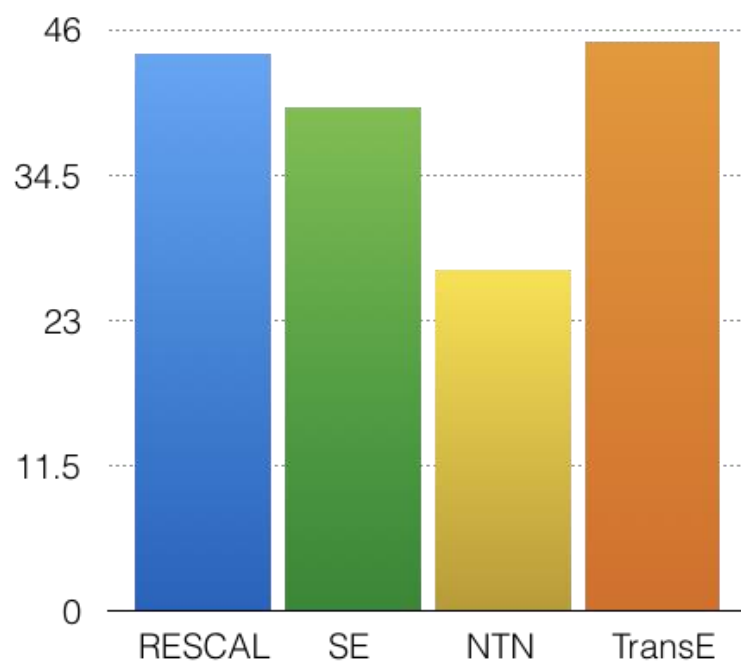
در جدول 6 میزان  $\text{Hit@10}$  و میانگین رتبه ی هر یک از روش های معرفی شده نمایش داده شده است که در این جدول مشاهده می شود که روش NTN در  $\text{hit@10}$  نتیجه ی مطلوبی نگرفته است که در معرفی این روش اشاره کردیم که این روش به سرعت به بیش برآزش میل می کند و همین امر باعث شده است که پاسخ های درستی در مورد مساله نداشته باشد.

نکته ی مهم دیگری که در این جدول دیده می شود کارایی خوب روش های SE و TransE هست که علاوه بر کاهش پارامتری که داشته اند، هم از نظر  $\text{hit@10}$  و هم از نظر میانگین رتبه نتایج خوبی داشته اند.

در روش RESCAL نیز مقدار  $\text{hit@10}$  خوب بوده است اما رتبه ی میانگین مقدار بسیار بالایی نسبت به روش های دیگر داشته که نشان از این دارد که علی رقم اینکه بسیاری از روابط را درست فهمیده است و نتایج خوبی در آن ها گرفته است، اما بسیاری از روابط را نیز اصلا درک نکرده و پاسخ هایی که به سوالات در مورد این روابط داده است جواب های تقریبا تصادفی ای داشته است و رتبه های پرتی گرفته اند که باعث شده میانگین رتبه ی این روش چنین افزایشی داشته باشد.

جدول 6 مقدار رتبه ی میانگین و hit10 در روش های مورد بررسی

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



شکل ۲۷ درصد hit@10 در روش های مورد بررسی

## ۴-۵- استفاده از قوانین انجمنی برای بهبود نتایج

در این فصل به طور کامل قوانین انجمنی و چگونگی استخراج آن ها را شرح دادیم و پنج قانون هدف نیز مشخص کردیم که به کمک آن ها تصمیم داریم مدل های موجود مبتنی بر ویژگی پنهان را بهبود دهیم. در این بخش به چگونگی انجام این بهبود می پردازیم.

در برخی از روش های معرفی شده در بخش + ابتدا همه ی این قانون ها را روی حقیقت های مجموعه ی آموزش اعمال کردیم و حقایق جدید بدست آمده را دوباره به پایگاه دانش اضافه کردیم و سپس از پایگاه دانش جدید که بزرگ تر، کامل تر و دقیق تر شده است استفاده کردیم.

اما در روش های TransE و TransH این قوانین را مستقیماً وارد فرایند آموزش کردیم. به این صورت که پایگاه دانش را گسترش ندادیم و با همان حدود ۵۰۰ هزار حقیقت اولیه عملیات آموزش را شروع کردیم اما در حین آموزش از این قوانین استفاده کردیم تا نتایج دقیق تری بگیریم که در ادامه به این موضوع خواهیم پرداخت که این قوانین چگونه استفاده شده اند.

عدم اعمال این قوانین روی پایگاه دانش باعث شد که هم مساله ی پیش پرداز برای اعمال قوانین روی پایگاه دانش را نداشته باشیم و هم مساله را با پایگاه دانش کوچک تری شروع کنیم که در دو مرحله باعث کاهش زمان محاسبات می شود.

در ادامه توضیحات استفاده از این قوانین را روی روش TransE می دهیم. در روش TransE هدف کاهش فاصله ی بین بردار موجودیت اول + بردار رابطه و بردار موجودیت دوم بود که برای حقیقت  $r(e_1, e_2)$  این رابطه را به صورت زیر می نوشتیم.

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

### ۱-۵-۴- قانون R-subsumtion

برای قانون R-subsumtion که به صورت  $r(x, y) \Rightarrow r'(x, y)$  است در عملیات آموزش هر جا عملیات کمینه سازی روی حقیقت  $r(x, y)$  انجام گرفت آن را روی روابط هم ارز آن که از این قانون بدست می آیند نیز انجام می دهیم، یعنی روابط  $r'(x, y)$ . پس در کمینه سازی ها علاوه بر کمینه کردن رابطه ی  $\|x + r - y\|$  رابطه ی  $\|x + r' - y\|$  را نیز کمینه می کنیم.

### ۲-۵-۴- قانون R-equivalence

در قانون R-equivalence که همانند رابطه ی قبل است با این تفاوت که قانون برگشت پذیر نیز هست، یعنی  $r(x, y) \Leftrightarrow r'(x, y)$  می توان بصورت بالا عمل کرد و هر قسمت (سمت چپ و راست قانون) از این قانون در حقیقت ها دیده شد قسمت دیگر را نیز در معادله ی کمینه سازی قرار دهیم. اگر معیار اطمینان این قانون بالا باشد، عملاً اتفاقی که می افتد این است که رابطه های  $r$  و  $r'$  به یک شکل آموزش دیده می شوند و به یک صورت عمل خواهند کرد، که با توجه به هم معنا بودن رابطه های مثل `located_in` و `contained_by` این اتفاق، اتفاق منطقی ای خواهد بود و از آن امید بهبود در مدل ها را داریم.

### ۳-۵-۴- قانون تعدی (2-hope translation)

قانون تعدی یا 2-hope translation که به صورت

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

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

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

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

$$r_1 + r_2 = r'$$

پس از عکس این رابطه نیز می توانی 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:

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

#### ۴-۵-۴ قانون Triangle alignment

در این قانون نیز مانند قانون قبل می توان عمل کرد و با پیدا کردن هر یک از حقیقت های موجود در بدنه ی قانون، دومین حقیقت را جستجو کرد و در صورت پیدا کردن آن، علاوه بر کمینه سازی که روی خود حقیقت های اصلی انجام می دهیم کمینه سازی را روی حقیقت بدست آمده از قانون نیز اعمال کنیم.

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

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

#### ۴-۵-۵ Specfic R-subsumption قانون

در این قانون همچون دو قانون قبل باید دو حقیقت که در سمت بدنه ی قانون آمده اند را یافته و پس از آن نتیجه ی قانون را در عملیات آموزش تاثیر دهیم. این قانون را به این صورت تاثیر می دهیم که اگر دو قانون  $r_1(x, y)$  و  $r_2(x, V)$  در حقایق وجود داشت، علاوه بر کمینه کردن فاصله ی بین این دو حقیقت عبارت نتیجه ی این قانون را نیز در عملیات کمینه سازی شرکت می دهیم به صورت

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

## فصل ۵: ارزیابی

### ۵-۱- مقدمه

در فصل قبل قوانین انجمنی و نحوه ی استخراج، و چگونگی استفاده از آن ها را توضیح دادیم. در این فصل این قوانین انجمنی را وارد عملیات آموزش کرده و نتایج بدست آمده را بررسی و مقایسه می کنیم. در بخش ۵-۲ به تشریح آزمایش ها و نتایج بدست آمده می پردازیم و آزمایش ها را به تفکیک قوانین هدف که در بخش ۳-۳ مطرح شد توضیح می دهیم. بخش ۵-۳ را به بررسی نتایج و مقایسه آن ها و جمع بندی این فصل اختصاص می دهیم و در فصل بعد نتیجه گیری خواهیم داشت و کارهایی که در آینده می توان برای بهبود بیشتر نتایج انجام داد را معرفی می کنیم.

### ۵-۲- آزمایش ها

در این تحقیق هدف فقط بهبود نتایج روش های مبتنی بر ویژگی های پنهان نبوده و به دنبال ارائه ی چارچوبی هستیم که به کمک آن بتوانیم قدرت و نقاط قوت و ضعف روش ها را نیز مشخص کنیم. برای ارائه ی این چارچوب از بخش بندی ای که در بخش ۳-۳ روی قوانین انجمنی داشتیم استفاده می کنیم. و بررسی می کنیم که هر یک از این دسته قوانین چه مقدار توسط مدل درک شده اند و خوب جواب می دهند.

برای انجام آزمایش ها باید پایگاه دانش و قوانین انجمنی که می خواهیم عملیات آموزش را روی آن ها انجام دهیم را مشخص کنیم. پایگاه دانش استفاده شده همان پایگاه دانش Freebase15k است که در بخش ۳-۲ معرفی شد (حدود ۶۰۰ هزار حقیقت، ۱۵ هزار موجودیت و ۱۳۰۰ رابطه) است. در ادامه قوانین انجمنی را روی این پایگاه دانش اجرا می کنیم و قوانین بدست آمده را دسته بندی می کنیم.

همانطور که در بخش ۴-۳ اشاره شد، پس از اجرای قوانین انجمنی روی پایگاه دانش Freebase15k ۴۱۱۹۶ قانون استخراج شد اما همه ی این قانون ها از نظر معیار کیفیت و اطمینان، شرایط استفاده در عملیات آموزش را ندارند، نیاز است تا قوانین با کیفیت را مشخص کنیم و فقط از آن ها 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 محدودیت ها تعداد قوانین کاندید بدست آمده برابر با ۴۱۹۳ قانون شد. این قوانین را بر اساس تقسیم بندی های بخش ۳-۳ دسته بندی کردیم که این دسته بندی در جدول ۷ مشخص شده است.

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

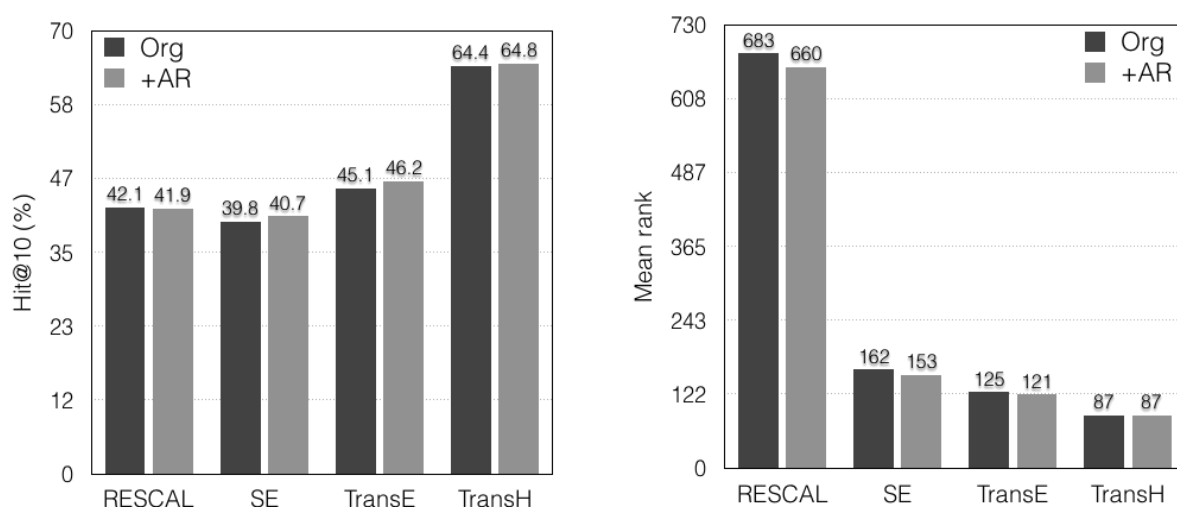
Rules		
name	rule	#rule
R-Subsumption	$r(x, y) \Rightarrow r'(x, y)$	۱۱۲۷
R-Equivalence	$r(x, y) \Leftrightarrow r'(x, y)$	۷۸۲
2-hope	$r1(x, y), r2(y, z) \Rightarrow r'(x, z)$	۸۱۰
Triangle alignment	$r(x, z), r(y, z) \Rightarrow r'(x, y)$	۶۹۵
SR-Subsumption	$r_1(x, y), r_2(x, v) \Rightarrow r'(x, y)$	۷۷۹



all rules	۴۱۹۳
-----------	------

همانطور که قبل تر هم اشاره شد تصمیم داریم که علاوه بر بهبود نتایج یک چارچوب برای سنجش کیفیت روش ها نیز ارائه دهیم، برای این کار طبق آنچه در بخش ۴-۲- گفته شد این قوانین را به صورت دسته ای به روش های آموزش اضافه می کنیم و نتایج را مشاهده می کنیم و در آخر نیز همه ی قوانین را باهم به روش ها اضافه می کنیم و میزان بهبود انجام شده توسط این قانون ها را در روش های مختلف بررسی می کنیم.

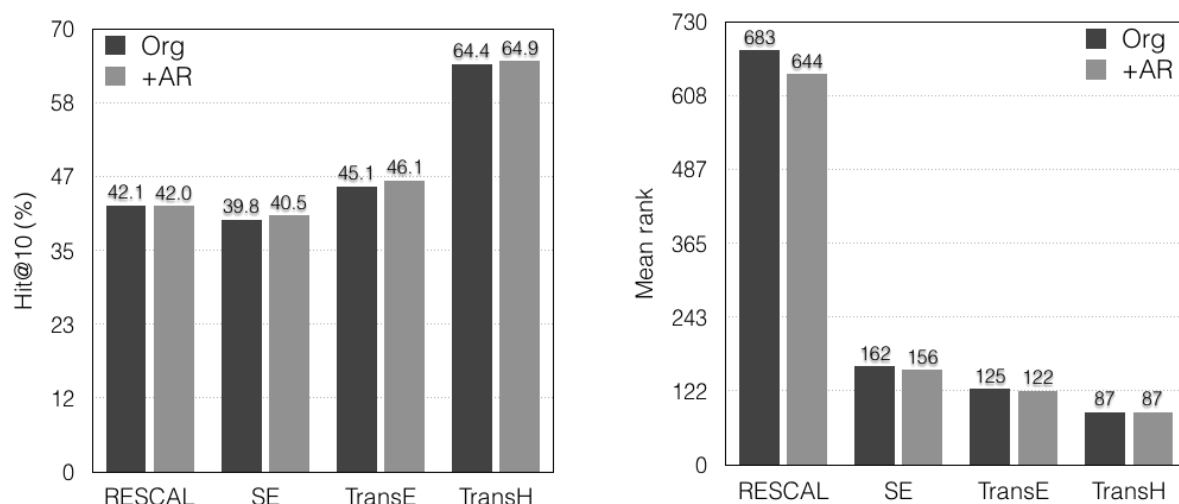
در نمودار شکل ۲۸-چپ معیار  $hit@10$  و در نمودار **Error! Reference source not found.** راست معیار میانگین رتبه برای روش های معرفی شده در ۵-۲-، قبل و بعد از اضافه کردن قوانین استخراج شده در دسته ی R-Subsumption را مشاهده می کنیم.



شکل ۲۸-چپ تاثیر اعمال قانون R-Subsumption بر  $hit@10$  -راست تاثیر اعمال قانون R-Subsumption بر رتبه ی میانگین

در نمودارها مشاهده می شود که اضافه کردن قوانین R-subsumption در هیچ یک از معیارها تاثیر چندان چشم گیری نداشته و بهبودهای جزئی روی نتایج روش ها داشته است. این مساله نشانگر این است

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

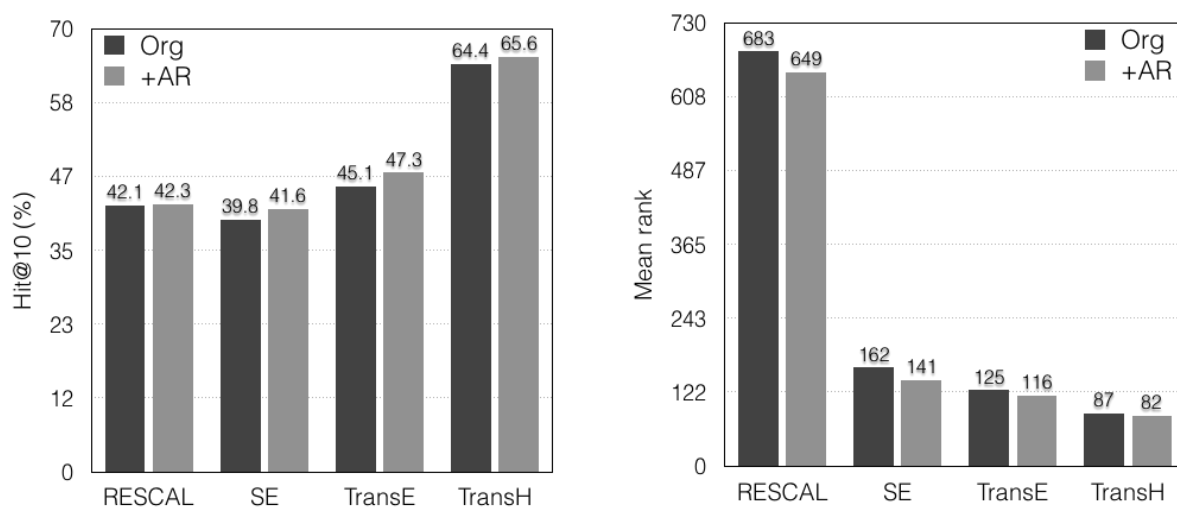


شکل ۲۹-راست تاثیر اعمال قانون SR-Subsumption بر رتبه ی میانگین - چپ تاثیر اعمال قانون SR-

Subsumption بر hit@10

در نمودارهای شکل ۲۹-راست و **Error! Reference source not found.** چپ نتایج معیارهای کیفیت روش های آموزش را قبل و بعد از اضافه کردن قوانین SR-subsumption مشاهده می کنیم. این قانون ها نیز مانند قانون های R-subsumption تاثیر چندانی در معیارهای hit@10 و میانگین رتبه نداشته است.

می توان نتیجه گرفت که مدل های مبتنی بر ویژگی های پنهان در کل قوانینی که به طور مستقیم از روی یکدیگر نتیجه گرفته می شوند را به خوبی درک می کنند و مدل آموزش دیده این روابط رو به خوبی تشخیص می دهد و در پاسخ به سوالات به اشتباه نمی افتد.

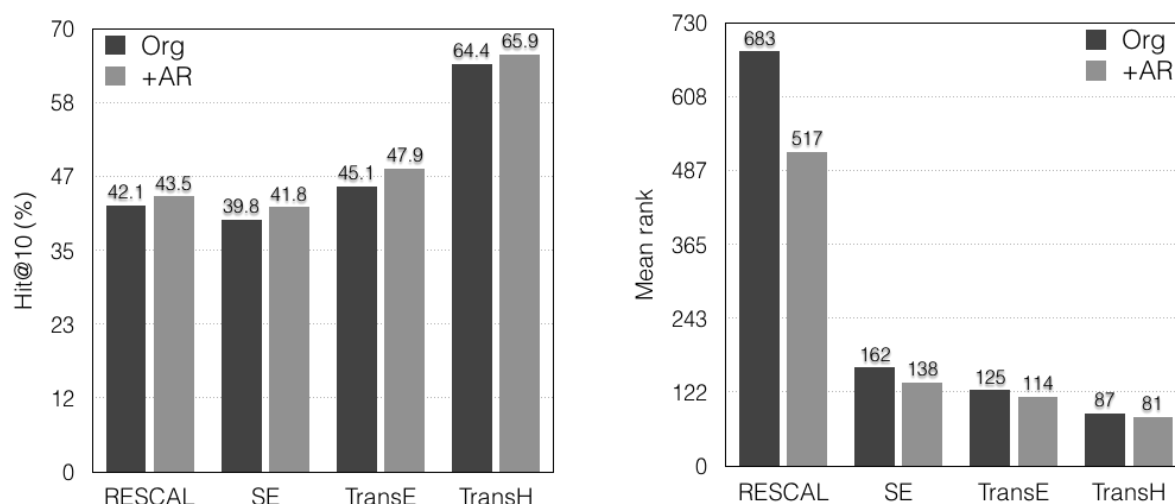


شکل ۳۰- چپ تاثیر اعمال قانون R-equivalence بر hit@10 - راست تاثیر اعمال قانون R-equivalence بر رتبه‌ی میانگین

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

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



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

در نمودارهای شکل ۳۱-راست و شکل ۳۱-چپ با اضافه کردن قوانین 2-hope translation نیز در هر دو معیار بهبود محسوسی دیده می شود. مشاهده می شود که در روش RESCAL نیز که با اضافه کردن قانون های قبلی تغییر چندانی نکرده بود، با اضافه کردن این قانون بهبود بسیار زیادی داشته ایم و مقدار hit@10 به ۴۳,۵ و مقدار میانگین رتبه با ۱۶۶ رتبه کاهش به ۵۱۷ رسیده است که بهبود چشم گیری است. همچنین در دیگر مدل ها بهبودهای زیادی را داشته ایم، در روش های SE و TransE و TransH هر کدام 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 می شود.

می توانیم نتیجه بگیریم که هیچ یک از این مدل ها قادر به شناسایی قوانین غیر مستقیم پیچیده نبوده است. در آزمایش های قبلی دیدیم که قوانینی غیر مستقیم که با استنتاج از روی یک رابطه ساخته شده بودند (در قسمت بدنه ی قانون فقط یک حقیقت وجود داشت) را مدل ها بهتر درک کرده بودن و اضافه کردن قوانین مربوط به این آزمایش ها تاثیر چندانی در نتیجه نداشت، اما اضافه کردن قوانین غیرمستقیم پیچیده تری مانند قانون 2-hope translation (که در سمت بدنه ی قانون از دو حقیقت استفاده شده

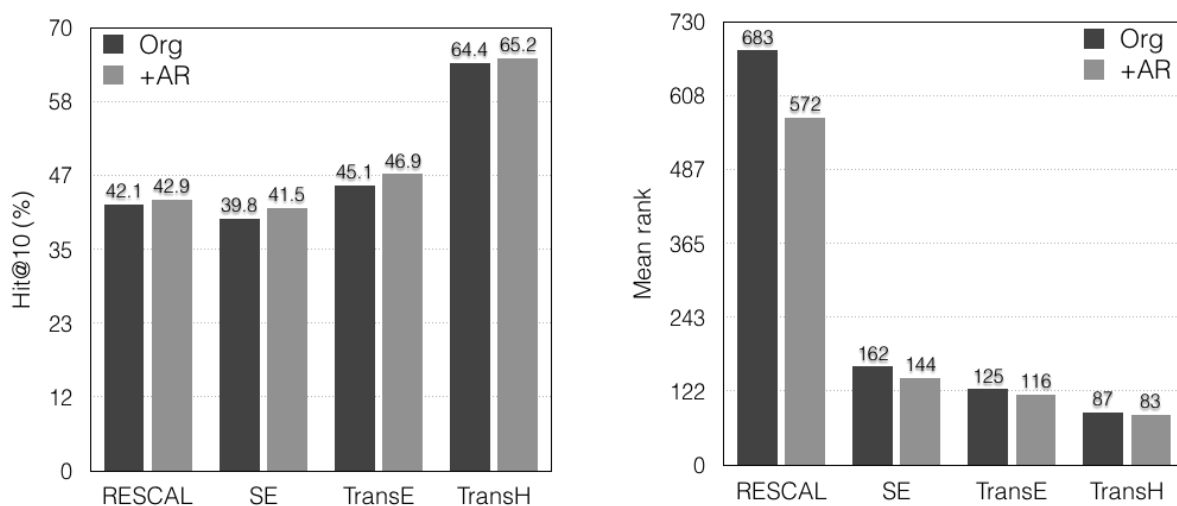
است) بهبودهای زیادی را در همه ی مدل ها اعمال کرده است. پس به کمک این قانون ها توانستیم چند قدم جلوتر از آنچه مدل ها قادر به دیدن آن بودند را به آن ها نشان دهیم و قوانین پیچیده ی موجود در پایگاه های دانش را نیز در امر آموزش دخیل کنیم.

بهبود در معیار  $hit@10$  نشان می دهد که اضافه کردن این قوانین باعث شده که در جواب های دقیق که به سوالات داده شده است بهبود داشته باشیم و تعداد جواب هایی صحیح که زیر ۱۰ کاندید اول بودند را افزایش دهیم و از طرفی بهبود در معیار میانگین رتبه نشان می دهد که در رابطه هایی که مدل برای آن ها درست آموزش ندیده است و جواب های پرتی برای آن ها در نظر گرفته است، اوضاع پاسخ ها بهتر شود.

به طور خاص یکی از دلایل بالا بودن معیار میانگین رتبه این است که برای برخی روابط حقیقت های کمی در پایگاه دانش موجود است و مدل ها از روی این تعداد کم رابطه قادر به شناسایی الگو برای پاسخگویی به سوالات در مورد این حقیقت ها و روابط نیستند. همین امر باعث می شود که پاسخ هایی که به سوالات در مورد این حقیقت ها می دهند به صورت تصادفی باشد و رتبه های پرتی بگیرد. این رتبه های پرت باعث افزایش زیادی در معیار میانگین رتبه می شود، در روش RESCAL این مساله را به خوبی مشاهده می کنیم که با وجود اینکه قریب به ۴۲٪ جواب ها زیر رتبه ی ۱۰ قرار می گیرند اما میانگین رتبه ای که برای این روش اعلام شده است ۶۸۳ است.

این مشکل عدم وجود تعداد نمونه ی کافی برای آموزش مدل را تا حدی می توان با اضافه کردن نمونه های غیر مستقیم موجود در پایگاه دانش حل کرد، به این معنی که برای خیلی از روابط نمونه هایی وجود دارد که از روی بقیه ی حقایق درون پایگاه دانش می توان به آن ها پی برد و از آن ها استفاده کرد. در اینجا هم ما با اضافه کردن قانون های 2-hope translation چنین کاری صورت داده ایم و حقیقت هایی که از به وسیله ی قانون تعدی قابل شناسایی بوده اند را به پایگاه دانش اضافه کرده ایم و در عملیات

آموزش تاثیر داده ایم و نهایتا نتایج خوبی نیز در پاسخ به سوالاتی که از مدل پرسیده می شود بدست آمده است.

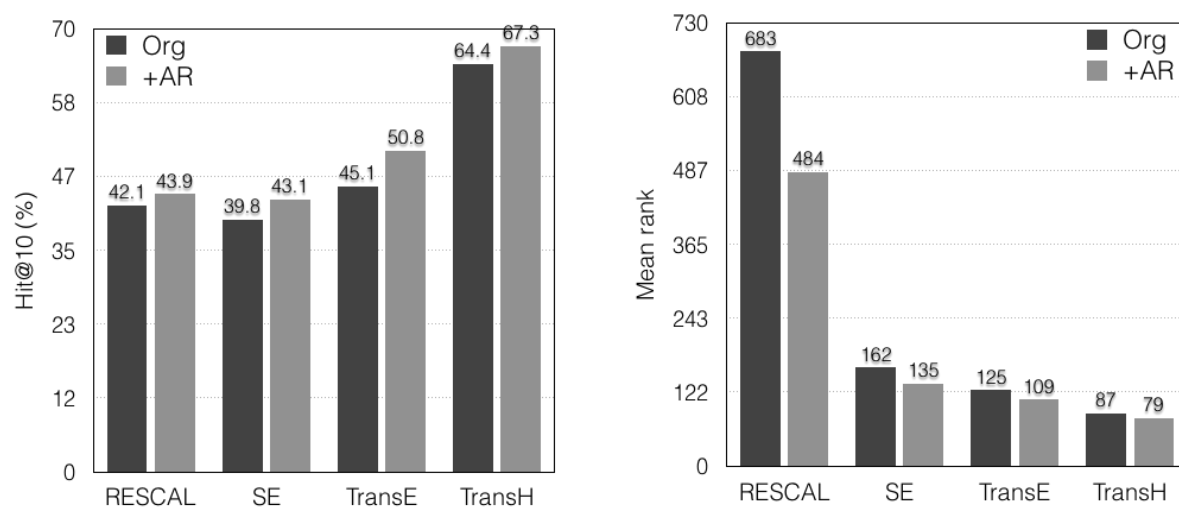


شکل ۳۲- چپ تاثیر اعمال قانون Triangle alignment بر hit@10 - راست تاثیر اعمال قانون Triangle

alignment بر رتبه ی میانگین

در نمودارهای شکل ۳۲-راست و شکل ۳۲-چپ نیز تاثیر استفاده از قوانین Triangle alignment را مشاهده می کنیم 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 می کند و باعث بهبود نتایج شده است.

در ادامه نتایج اعمال کل قانون هایی که در قسمت ۳-۳ معرفی کردیم را روی مدل های معرفی شده را بررسی می کنیم. نتایج بدست آمده در نمودارهای + و + نشان داده شده است.



شکل ۳۳- چپ تاثیر اعمال همه ی قانون های هدف بر hit@10 - راست تاثیر اعمال همه ی قانون های هدف بر رتبه ی میانگین

مشاهده می شود که اعمال همه ی قانون ها در کنار هم نیز باعث بهبود در نتایج همه ی روش های معرفی شده، شده است و علاوه بر چارچوبی که برای سنجش نقاط قوت و ضعف روش ها معرفی کردیم توانستیم در همه ی روش ها بهبودهای قابل قبولی نیز اعمال کنیم.

### ۳-۵- جمع بندی

در جدول ۸ و جدول ۹ جمع بندی ای داریم روی نتایج اعمال قوانین روی روش های 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 داده ایم.

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

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

<b>RESCAL</b>	42.1	41.9	42.3	43.5	42.9	42.0	<b>43.9 (1.8%)</b>
<b>NTN</b>	27	-	-	-	-	-	-
<b>SE</b>	39.8	40.7	41.6	41.8	41.5	40.5	<b>43.1 (3.3%)</b>
<b>TransE</b>	45.1	46.2	47.3	47.9	46.9	46.1	<b>50.8 (5.7%)</b>
<b>TransH</b>	64.4	64.8	65.6	65.9	65.2	64.9	<b>67.3 (2.9%)</b>

جدول 9 نتایج اعمال قوانین انجمنی هدف بر روی معیار رتبه‌ی میانگین روش های مورد بررسی

Mean rank on Freebase 15K							
Method	original	R-Subsumption	R-Equivalence	2-hope	Triangle alignment	SR-Subsumption	all rules (~improve)
<b>RESCAL</b>	683	660	649	517	572	644	<b>484 (30%)</b>
<b>NTN</b>	164	-	-	-	-	-	-
<b>SE</b>	162	153	141	138	144	156	<b>135 (17%)</b>
<b>TransE</b>	125	121	116	114	116	122	<b>109 (13%)</b>
<b>TransH</b>	87	87	82	81	83	87	<b>79 (10%)</b>

نکاتی که در این جداول قابل توجه هستند:



- اعمال همه ی قوانین روی هر چهار روش مورد آزمایش باعث بهبود قابل قبول نتایج گشته است.
- قوانین پیچیده تر مانند قانون 2-hope و قانون Triangle alignment که به طور مستقیم از روی داده های موجود در پایگاه دانش قابل برداشت نیستند توسط مدل ها به خوبی درک نشده اند و از نقاط ضعف روش ها به شمار می روند که در جدول مشاهده می کنیم اضافه کردن این قوانین بهبود خوبی در نتایج داشته است.
- قانون R-equivalence مانند دو قانون بند قبل قانون پیچیده ای نیست، اما به حل یکی از مشکلات اساسی پایگاه های 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 می دهد نیز باعث تاثیر خوبی در نتایج همه ی روش ها بجز روش RESCAL شده است که در بخش ۴-۱-۲ دلیل عدم بهبود برای روش RESCAL را توضیح دادیم.
- قوانین ساده تر مانند R-subsumption و SR-subsumption تقریباً توسط همه ی مدل ها به خوبی درک شده است و اضافه کردن این قانون ها بهبود چندانی را در نتایج حاصل نکرده است.
- به طور خاص در روش RESCAL بهبود در معیار  $hit@10$  نسبت به دیگر روش ها 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 می دهد که روش RESCAL برای روابطی که به میزان کافی نمونه برای آموزش دیدن داشته است خوب عمل کرده و جواب حدود ۴۰٪ از سوال ها را در رتبه ی 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 نمونه ی کافی از آن ها موجود بوده است مفید

واقع نشده است، اما کاهش بسیار زیاد میانگین رتبه ی پاسخ ها نشان می دهد که کمک بسیاری به روابطی که نمونه ی کافی نداشته اند شده است و جواب های بیشتری به سمت منطقی شدن پیش رفته است.

- متاسفانه پیاده سازی روش NTN کامل نشده و نتایج اعمال این قوانین روی این روش را برای مقایسه در اختیار نداریم.

در این تحقیق ابتدا روش های موجود در زمینه ی پیش بینی پیوند که از ویژگی های پنهان استفاده می کردند معرفی شد و سپس با معرفی و استفاده از قوانین انجمنی سعی شد تا در این روش ها بهبودهایی ایجاد شود. با توجه به نتایجی که مشاهده کردیم و نکاتی که گفته شد در کل دست آوردهای این تحقیق را می توان به صورت زیر خلاصه کرد:

- استفاده از قوانین انجمنی در بهبود روش های مبتنی ویژگی های پنهان
- ارایه ی چارچوبی برای مقایسه و بررسی عملکرد و نقاط ضعف و قوت روش های موجود مبتنی بر ویژگی های پنهان که قابل گسترش به روش های دیگر در زمینه ی پیش بینی پیوند نیز هست.

## فصل ۶: نتیجه‌گیری

### ۶-۱- نتیجه‌گیری

متن

### ۶-۲- 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  $hit@1$  را در نظر بگیریم، زیرا فقط یک جواب صحیح مدنظر است و دیگر جواب‌ها فاقد اعتبارند و جواب‌های درست به شدت کاهش پیدا می‌کند.

با این که این روش ها در حال حاضر در آخرین تکنولوژی های شرکت های بزرگ مانند گوگل و IBM در حال استفاده هستند، اما این روش ها هنوز باید بسیار دقیق تر شوند تا در کاربردهایی مثل پاسخگویی به سوالات نیز قابل استفاده باشند.

بهبودهایی که می توان روی این مساله داد می توانند از طریق راهکارهای زیر باشند:

- استفاده از روشی که در این تحقیق ارائه شده و یافتن نقاط ضعف روش ها و انتخاب راه حل برای حل این ضعف ها
- در این تحقیق فقط قوانین انجمنی با پیچیدگی ۱ و ۲ (قوانینی که در قسمت بدنه ی خود دو حقیقت را بررسی می کردند) در نظر گرفته شدند، مطمئنا درک قوانین با پیچیدگی های بیشتر نیز برای مدل های موجود سخت بوده و ممکن است نقاط ضعف این روش ها باشند، پس اضافه کردن قوانین با پیچیدگی بیشتر از ۲ می تواند بهبودهایی را حاصل کند.
- راهکار دیگری برای پایین آوردن معیار میانگین رتبه، فیلتر کردن نتایج روش ها است. می دانیم پاسخ هایی که روش های موجود به سوالات می دهند به صورت لیستی از جواب های مرتب شده است که ممکن است جواب درست سوال مورد ۵۰ام باشد. با بررسی جواب ها مشخص می شود که خیلی از پاسخ هایی که در ۴۹ پاسخ غلط وجود داشته کلا بی ارتباط با سوال است. مثلا سوال شده که «نویسنده ی کتاب X چه شخصی است؟» و انتظار داریم جواب های دریافتی نام اشخاص باشد، اما پاسخ هایی از جنس های دیگر مانند نام کشورها، وضعیت های آب و هوا و ... در پاسخ های غلط پیش از پاسخ درست وجود دارد. برای حل این مشکل و فیلتر کردن پاسخ های بی ربط می توان از روش های کشف جامعه<sup>۱</sup> استفاده کرد و موجودیت ها را دسته بندی کرد، و به مدل ها آموزش داد که جواب هر سوال باید از جنس چه مجموعه موجودیتی باشد و پس از

<sup>1</sup> community detection

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



## فصل ۷: مراجع

---

- [1] M. Gholizadeh, F. Taghiyareh, and S. Alvandkoochi, "Determining the Interrelationship Among Player Engagement Features in Game-Based Vocabulary," in *European Conference on Games Based Learning*, 2017.

## فصل ۸: واژه‌نامه انگلیسی به فارسی

---

Abstraction	تجريد
Action	عمل
Appliance	دستگاه
Asynchronous	ناهمگام
Autonomous	خودمختار
Batch	دسته
Batch Delay	تاخير 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	پیش فرض
Deterministic	قطعی
Diversity	تنوع
Driver	راه انداز
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	برون خط
Online	بر خط
Out-of-order	خارج از نوبت
Performance	کارایی
Pipeline	خط لوله
Platform	سکو
Pool	مخزن
Pre-filtering	پیش تصفیه
Prefix Match	انطباق پیشوندی
Preprocessing	پیش پردازش
Program Counter	شمارنده برنامه
Programmable	قابل برنامه ریزی
Range Match	انطباق بازه‌ای
Regular Expression	عبارت منظم
Replication Ratio	نسبت تکرار
Rule Programming	برنامه ریزی قانون
Scalability	مقیاس پذیری
Scheduler	زمان بند
Scheduling	زمان بندی
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 ۹: واژه‌نامه فارسی به انگلیسی

---

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

Pre-filtering	پیش تصفیه
Default	پیش فرض
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	کارایی
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	مقیاس پذیری
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	یکدست سازی
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: Software Defined Networking, Packet Classification, OpenFlow 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**