

گزارش پروژه درس مبانی فناوری اطلاعات
استفاده از گراف دانش در سیستم‌های پیشنهاددهنده

گروه ۶

فاطمه رحمانی

مرتضی زارعی

زهرا عبدی

مهرداد میرمحمدصادقی

تابستان ۹۹

فهرست

۳	۱	مقدمه
۳	۱.۱	چالش‌ها و مشکلات سیستم‌های پیشنهاددهنده
۴	۲.۱	روش‌های جمع‌آوری اطلاعات و ابزار آن
۴	۳.۱	گراف دانش
۷	۲	پیش‌زمینه
۷	۱.۲	تعبیه گراف دانش
۹	۲.۲	معیارهای مقایسه
۱۰	۳	انتشار فعالیت
۱۰	۱.۳	مقدمه
۱۰	۲.۳	گراف
۱۱	۳.۳	پیشنهاد دهی
۱۳	۴.۳	ارزیابی و نتیجه‌گیری
۱۶	۴	RKGE
۱۶	۱.۴	مقدمه
۱۷	۲.۴	تعبیه گراف دانش تکرار شونده
۱۹	۳.۴	استخراج معنای مسیر
۱۹	۴.۴	شبکه ای از شبکه‌های تکرار شونده
۲۲	۵.۴	آزمایش‌ها و تحلیل و بررسی
۲۶	۶.۴	نتیجه
۲۷	۵	RippleNet
۲۷	۱.۵	مقدمه
۲۹	۲.۵	RippleNet
۳۴	۳.۵	آزمایش‌ها

۳۷	KTUP ۶
۳۷	۱.۶ مقدمه
۳۸	۲.۶ کارهای انجام شده
۳۹	۳.۶ مدل
۳۹	۴.۶ TUP برای توصیه آیتم
۴۰	۵.۶ یادگیری مشترک از طریق KTUP برای دو وظیفه
۴۱	۶.۶ نتایج آزمایش‌ها
۴۲	۷.۶ نتیجه
۴۳	KGAT ۷
۴۳	۱.۷ مقدمه
۴۴	۲.۷ فرمول بندی کار
۴۵	۳.۷ ارتباط مرتبه‌بالا
۴۵	۴.۷ روش شناسی
۴۹	۵.۷ بهینه سازی
۴۹	۶.۷ نتیجه گیری
۵۱	IGNN ۸
۵۱	۱.۸ مقدمه
۵۱	۲.۸ معرفی مدل
۵۵	۳.۸ آزمایش‌ها
۵۸	۴.۸ نتیجه گیری
۵۹	۹ نتیجه‌گیری
۵۹	۱.۹ بررسی نهایی روش‌ها
۶۱	۲.۹ بررسی نتایج پیشنهاددهی حاصل از روش‌ها
۶۱	۳.۹ خلاصه ای از ویژگی‌های متدها
۶۳	References ۱۰

مقدمه

حجم بالای اطلاعات در اینترنت و افزایش روز افزون آن، کاربران را با مشکل یافتن اطلاعات مناسب در حداقل زمان مواجه کرده است و یافتن اطلاعات نهایی و داده‌ها براساس نیاز کاربران تبدیل به پروسه‌ای پیچیده و زمانبر شده است. درواقع همزمان با افزایش انتخاب‌ها، حجم اطلاعاتی که جهت رسیدن به هدف نهایی باید پردازش شوند و میزان زمان و انرژی مصرفی برای رسیدن به اطلاعات و داده‌های نهایی زیاد می‌شود. در چنین محیط‌هایی یافتن سیستمی با قابلیت شناخت، بروزرسانی علایق و سلیقه‌ها و همچنین اولویت‌های کاربران از یک طرف و توانایی شاخص‌گذاری و ذخیره اطلاعات به روشی قابل جستجو با سربار کم، برای پیش‌بینی و یافتن نیازهای کاربران و هدایت آن‌ها در جهت رفع نیازهایشان به شدت احساس می‌شود. سیستم‌های توصیه‌گر به عنوان تکنیک‌های هوشمند مبتنی بر کامپیوتر برای تسهیل تراکنش‌هایی با سربار زیاد اطلاعات و مشکلات اطلاعاتی تعریف می‌شوند. سیستم‌های پیشنهاددهنده براساس مدل سازی کاربر، می‌توانند کاربران را در جهت یافتن اطلاعات مورد نیاز وکالاهای مورد علاقه هدایت نمایند. این سیستم‌ها به طور موثر می‌توانند به هر دو طرف تراکنش سود برسانند. سیستم پیشنهاددهنده به مشتری از طریق هدایت وی به سمت گزینه‌های مورد علاقه‌اش و رفع نیاز وی کمک می‌کند. همچنین با افزایش مراجعه مشتریان به این سیستم‌ها و رضایتمندی آن‌ها، سازمان‌های ارائه‌دهنده نیز به هدف خود نائل می‌آیند.

۱.۱ چالش‌ها و مشکلات سیستم‌های پیشنهاددهنده

نمونه‌هایی از چالش‌های موجود در سیستم‌های توصیه‌گر عبارتند از:

- شروع سرد^۱: این اصطلاح به کاربرانی گفته می‌شود که جدید هستند و پروفایل آن‌ها تقریباً خالی می‌باشد و علایق و سلیقه‌های آن‌ها برای سیستم ناشناخته است که در بعضی سیستم‌های پیشنهاددهنده این مشکل با نظرخواهی هنگام ایجاد یک طرح حل می‌شود یا با کمک روش ترکیب نیز می‌توان این مشکل را حل نمود.
- اعتماد^۲: اعتبار افرادی که تاریخچه کوتاه‌تری در سیستم دارند با افرادی که مدت زمان طولانی‌تری عضو سیستم می‌باشند نباید برابر و یکسان باشد که این مشکل را نیز می‌توان با اولویت دادن به کاربران حل نمود.
- مقایسه‌پذیری^۳: با رشد تعداد کاربران و بخش‌ها، سیستم به منابع بیشتری برای پردازش اطلاعات و تشکیل توصیه‌ها نیاز دارد که این مشکل را می‌توان با ترکیب نوع‌های مختلف فیلترها و توسعه فیزیکی سیستم‌ها حل نمود.

^۱cold start

^۲Trust

^۳Scalability

- پراکندگی^۴: در فروشگاه‌های آنلاین، کاربران و اقلام زیادی وجود دارند که کاربران بخش کمی از اقلام را امتیازدهی می‌کنند. از سیستم‌های توصیه‌گر جمعی و روش‌های دیگر برای ایجاد همسایگی بین کاربران استفاده می‌گردد. درواقع پراکندگی مشکل فقدان اطلاعات است.
- حریم خصوصی^۵: مهم‌ترین مشکل، حفظ حریم خصوصی می‌باشد. سیستم‌های پیشنهاددهنده نیاز به اطلاعات دقیقی شامل داده‌های شخصی و داده‌هایی در مورد موقعیت کاربران دارند. بسیاری از فروشگاه‌های آنلاین حریم خصوصی کاربران را با استفاده از الگوریتم‌ها و برنامه‌های تخصصی حفظ می‌نمایند.

۲.۱ روش‌های جمع‌آوری اطلاعات و ابزار آن

سیستم‌های توصیه‌گر، سیستم‌هایی هستند که در پیدا کردن و انتخاب نمودن آیتم‌های موردنظر کاربران، به آن‌ها کمک می‌کنند. طبیعی است که این سیستم‌ها بدون در اختیار داشتن اطلاعات کافی و صحیح در مورد کاربران و آیتم‌های موردنظر آن‌ها (به عنوان مثال فیلم، موزیک، کتاب و ...) قادر به پیشنهاد دادن نمی‌باشند. بنابراین یکی از اساسی‌ترین اهداف آن‌ها، جمع‌آوری اطلاعات گوناگون در رابطه با سلیق کاربران و آیتم‌های موجود در سیستم است. منابع و روش‌های گوناگونی برای جمع‌آوری چنین اطلاعاتی وجود دارد. یک روش جمع‌آوری اطلاعات بصورت صریح است که در آن کاربر صراحتاً اعلام می‌کند که به چه چیزهایی علاقه دارد (به عنوان نمونه با امتیاز دادن به یک موزیک). روش دیگر، روش ضمنی است که کمی دشوارتر است و در آن سیستم باید سلیق کاربر را با کنترل و دنبال کردن رفتارها و فعالیت‌های او بیابد (به عنوان مثال باید ببیند کاربر به چه موزیک‌هایی بیشتر گوش می‌دهد، چه صفحاتی را بازدید می‌کند، با چه کسانی در ارتباط است و ...). علاوه بر اطلاعات ضمنی و صریح، برخی از سیستم‌ها نیز هستند که از اطلاعات شخصی کاربران استفاده می‌کنند. این سیستم‌ها، این اطلاعات را از طریق شبکه‌های اجتماعی بدست می‌آورند (به عنوان مثال سن، جنسیت و ملیت کاربران می‌تواند منبع خوبی برای شناخت کاربر و ارائه پیشنهاد به وی باشد. به این گونه از اطلاعات، اطلاعات جمعیت‌شناسی^۶ گفته می‌شود).

۳.۱ گراف دانش^۷

استنتاج یکی از مهارت‌های اولیه انسان‌ها برای کسب دانش بوده و هست ولی با گسترش داده‌های امروزی استنتاج به شکل قدیمی آن غیرممکن به نظر می‌رسد، ولی مهندسان برای دسته‌بندی و نشان دادن ارتباط بین داده‌ها شروع به استفاده از روش‌های جدیدی کردند که یکی از دستاوردهای آن گراف دانش بود که موجودیت‌ها و ارتباط بین آن‌ها را در یک گراف به نمایش می‌گذارد.

گراف دانش حاوی حجم زیادی از دانش است اما همچنین می‌تواند داده‌ها را به طور موثر سازماندهی کنند و به همین دلیل از آن به طور گسترده‌ای برای سیستم‌های پاسخگویی به سوالات، موتورهای جستجو و سیستم‌های پیشنهاد دهنده استفاده می‌شود. گراف دانش قادر به استخراج، سازماندهی و مدیریت موثر دانش از داده‌های در مقیاس بزرگ برای بهبود کیفیت خدمات اطلاعات و ارائه خدمات به کاربران است.

لیست ۱.۱ تعدادی از اولین گراف‌های دانش به وجود آمده است که هر کدام به دلایل خاص و با ویژگی‌های منحصر به فرد خود به وجود آمده‌اند. موفقیت آمیز بودن استفاده از گراف دانش در طیف گسترده‌ای از کارها موجب شد که محققان سعی کنند از گراف دانش برای بهبود عملکرد سیستم‌های پیشنهاددهنده استفاده کنند. گراف دانش از سه جهت می‌تواند برای پیشنهاددهی موثر باشد:

- گراف دانش ارتباط معنایی را بین آیتم‌ها ارائه می‌دهد، که می‌تواند به پیدا کردن ارتباطات پنهان آن‌ها و افزایش تنوع بخش‌های پیشنهادی کمک کند.

⁴Sparsity

⁵Privacy

⁶Demographic information

⁷Knowledge Graph

Knowledge graphs	#Entities	#Relations	#Facts
WordNet	0.15M	200,000	4.5M
Freebase	50M	38,000	3B
YAGO	17M	76	150M
DBpedia (En)	4.8M	2800	176M
Wikidata	16M	1673	66M
NELL	2M	425	120M

تصویر ۱۰.۱: اولین گراف‌های دانش به وجود آمده

- گراف دانش از ارتباطات با بخش‌های مختلفی تشکیل شده، که برای گسترش متعادل علایق کاربر و افزایش تنوع بخش‌های پیشنهادی مفید هستند.
- گراف دانش نتایج و سوابق تاریخی کاربر و موارد پیشنهادی را به هم متصل میکند و بدین ترتیب قابلیت توجه را برای سیستم پیشنهادی به وجود می‌آورد.

۱.۳.۱ دسته بندی

مندهای متفاوتی برای استفاده و استخراج اطلاعات از گراف دانش در سیستم‌های پیشنهاد دهنده تا به حال استفاده شده است که به چهار دسته کلی زیر تقسیم می‌شوند:

روش‌های جستجوی گراف^۸

روش‌های مبتنی به گراف برای پیشنهاد دادن به ساختار گراف و اتصالات موجودیت‌ها توجه می‌کند و مدل‌های مختلفی در این دسته جای گرفته‌اند که نتایج بسیار خوبی هم گرفته‌اند. با این وجود همه این متدها تنها از روش پیاده‌روی تصادفی^۹ روی گراف استفاده کرده‌اند که به راحتی روی داده‌های مرکزی و مهم بایاس می‌شوند و مهم‌تر از آن این روش‌ها فقط به توپولوژی و ساختار گراف اهمیت می‌دهند و به معنای مسیرها یا ارتباط بین موجودیت‌ها دقت نمی‌کنند که این باعث می‌شود از همه توانایی‌های گراف دانش برای پیشنهاد دادن استفاده نکنند.

روش‌های مبتنی بر مسیر ارجح^{۱۰}

مجموعه‌ی دیگری از متدهای استخراج اطلاعات، استفاده از مسیرهای ارجح روی گراف دانش هستند؛ به این صورت که از قبل فرمت‌هایی از طول و نوع مسیر در گراف را مشخص می‌کنند و از این مسیرها معنی‌های متفاوتی که در گراف دانش وجود داشته را برداشت می‌کنند. موفقیت این مدل‌ها به شدت به تعداد و کیفیت این مسیرهای از پیش تعریف شده وابسته است و تعریف این مسیرها به دانش زمینه‌ای نیازمند است که این متدها را برای پیشنهاددهی باکیفیت محدود می‌کند. تعریف این مسیرهای ارجح هم می‌تواند به صورت دستی و هم به وسیله مدل‌هایی که برای این منظور آموزش می‌بینند انجام شود.

روش‌های مبتنی بر تعبیه سازی گراف دانش^{۱۱}

دسته دیگری که نسبت به دو دسته قبلی جدیدتر است استفاده از تعبیه گراف دانش برای پیشنهاددهی است؛ تعبیه گراف صرفاً برای پیشنهاددهی معرفی

⁸Graph search Methods

⁹Random walk

¹⁰Meta Path based Methods

¹¹KG Embedding based Methods

نشده و برای کاربردهای متفاوتی از این مدل استفاده می‌شود ولی اکثر مدل‌های پیشنهاددهی جدید برای تشخیص روابط نهفته بین عناصر از این روش استفاده می‌کنند. این مفهوم در ادامه به تفصیل توضیح داده خواهد شد.

روش‌های مبتنی بر نظم^{۱۲}

می‌دانیم که گراف‌های دانشی که از آن‌ها برای پیشنهاددهی استفاده می‌کنیم گراف‌های کاملی نیستند و پراکندگی داده در آن‌ها وجود دارد. برای اینکه مشکل پراکندگی در گراف را حل شود در برخی متدها تکمیل گراف دانش در کنار عملیات پیشنهاددهی انجام می‌شود و روابطی که به نظر باید در گراف اصلی وجود داشته باشند به صورت ضمنی به گراف اضافه می‌شوند. در نهایت برای پیشنهاددهی از مجموعه اطلاعات به دست آمده استفاده می‌شود.

مدل‌هایی که در ادامه به بررسی دقیق آن‌ها می‌پردازیم در یک دسته یا ترکیبی از دسته‌هایی که مطرح کردیم جای می‌گیرند.

¹²Regularization based Methods

پیش زمینه

در ادامه به بررسی مقاله‌های تخصصی در این زمینه می‌پردازیم که فهم آن منوط بر دانستن پیش‌زمینه‌های موجود خواهد بود؛ پس قبل از شروع، به بررسی مفاهیم و اصطلاحات پرستفاده می‌پردازیم تا هم به فهم بهتر کمک کرده هم از تکرار این مفاهیم در هر قسمت جلوگیری کنیم.

۱.۲ تعبیه گراف دانش

فهم موجودیت‌های مختلف برای یک سیستم کامپیوتری کار ساده‌ای نیست مخصوصاً اگر بخواهیم براساس این فهم در مورد انتخاب این موجودیت‌ها تصمیم بگیریم؛ تعبیه یا درون‌سازی عملیاتی است که یک موجودیت را براساس یک وکتور در یک فضای پیوسته توصیف می‌کند. یکی از تعبیه‌هایی که در هوش مصنوعی و پردازش زبان طبیعی استفاده می‌شود تعبیه کردن کلمات است که کلمات موجود را به صورت وکتورهای معرفی می‌کند تا با کنار هم قرار دادن این وکتورها معنای جملات یا موارد دیگر را بررسی کنند.

تعبیه‌سازی گراف [۱] نیز از مواردی است که برای استخراج اطلاعات از گراف‌های دانش به وفور استفاده می‌شود، در واقع سعی می‌کند که گره‌ها و رابطه‌های موجود در گراف را براساس ارتباطی که در گراف دارند به صورت وکتورهایی در فضای پیوسته توصیف کند. این عملیات نوع‌های مختلفی دارد که می‌تواند براساس فاصله‌های تعریف شده در گراف^۱ یا براساس معنی این روابط^۲ باشد. مدل‌های TransE، TransH و TransR از نوع اول یعنی مبتنی بر فاصله و مدل‌هایی مثل Analogy، ComplEx و Dismult مدل‌های تطابق معنایی هستند. در ادامه به بررسی روش‌های تعبیه گراف مبتنی بر فاصله و مدل‌های آن می‌پردازیم.

هرگراف را می‌توانیم براساس گره‌های آن و اتصال آن‌ها از طریق یال‌ها به صورت کامل معرفی کنیم، گراف دانش نیز از این قاعده مستثنی نیست و می‌توانیم گراف را به صورت مجموعه‌ای از سه‌تایی‌های مرتب به صورت $\langle \text{head}, \text{relation}, \text{tail} \rangle$ نمایش دهیم که موجودیت‌های ابتدا و انتهای یک رابطه و یال بین آن‌ها یعنی رابطه بین آن‌ها را نشان می‌دهد.

برای ایجاد وکتورهایی که معرف موجودیت‌ها و روابط در گراف هستند باید سه مرحله زیر را پیش ببریم:

۱. ایجاد وکتورهای رندوم برای هرکدام

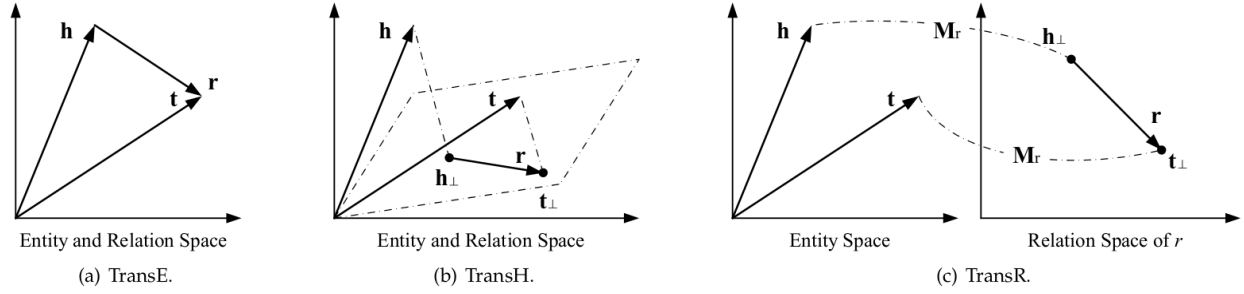
۲. تعریف تابع هزینه

۳. مینیمم کردن تابع هزینه با تغییر وکتورهای تولید شده

این مراحل در همه روش‌های مبتنی بر فاصله یکسان است ولی تعریف تابع هزینه در هر کدام به صورت منحصر به فرد و با توجه به نیازها و شرایط تعبیه گراف انجام می‌شود.

¹Translational Distance Models

²Semantic Matching Models



تصویر ۱.۲: مدل‌های مبتنی بر فاصله

۱.۱.۲ TransE

یکی از مدل‌های ساده و پرکاربرد در این حوزه transE است که سعی می‌کند در هر رابطه سه‌تایی $\langle h, r, t \rangle$ فاصله $h + r$ از وکتور t کمینه باشد. تابع هزینه در این مدل به صورت زیر تعریف می‌شود:

$$f_r(h, t) = -||h + r - t||_{1/2}$$

این نوع از تعبیه به دلیل استفاده از تعریف ساده فاصله بسیار پرکاربرد است ولی یکی از مشکلات این مدل ناتوانی در نمایش روابط یک-به-چند، چند-به-یک و چند-به-چند است. برای مشهود شدن این مشکل را با یک مثال ساده بررسی می‌کنیم، مثلاً در صورتی که یک کارگردان چند فیلم را ساخته باشد این فیلم‌ها با رابطه "ساخته شدن توسط" به این کارگردان مرتبط هستند و به دلیل یکسان بودن رابطه و ابتدای رابطه (کارگردان) وکتور حاصل از این فیلم‌ها بسیار مشابه هم خواهد بود، در صورتی که هر کدام از آن‌ها موجودیت‌های متفاوتی هستند.

۲.۱.۲ TransH

برای حل مشکل TransE در نمایش روابطی که به چند موجودیت مربوط می‌شود مدل جدید TransH ارائه شد که برای هر رابطه یک ابرصفحه در فضای $n + 1$ بعدی که n بعد وکتورهای مورد نظر است در نظر می‌گیرد و سپس برای هر رابطه ابتدا بردار را روی ابر-صفحه مورد نظر تصویر کرده سپس تابع هزینه‌ای که همانند تابع TransE است را برای آن محاسبه می‌کند. مثلاً در صورتی که بردارها در فضای دو بعدی نمایش داده شوند تصویر ۱.۲ نحوه تصویر کردن دو بردار را بر صفحه دوبعدی رابطه مورد نظر نشان می‌دهد (ابرفصله برای فضای سه بعدی محاسبه می‌شود که حاصل آن صفحه‌های دو بعدی رابطه است، مشابه این که دو بردار در فضای دو بعدی را در چه جهات مختلفی می‌توانیم مشاهده کنیم)

$$h_{\perp} = h - w_r^T h w_r, \quad t_{\perp} = t - w_r^T t w_r$$

$$f_r(h, t) = -||h_{\perp} + r - t_{\perp}||_2^2$$

۳.۱.۲ TransR

در این روش ایده اصلی بسیار مشابه TransH است ولی به جای تعریف ابر-صفحه برای هر رابطه کل فضای رابطه را تغییر می‌دهد. برای تغییر فضای موجودیت‌ها بر اساس هر رابطه برای هر رابطه یک ماتریس در نظر گرفته می‌شود که با ضرب وکتورهای هر موجودیت در این ماتریس‌ها وکتورهای مورد نظر در فضای جدید محاسبه می‌شوند و سپس تابع هزینه بر اساس وکتورهای جدید محاسبه شده مشابه تابع TransE اعمال می‌شود.

$$h_{\perp} = M_r^1 h, \quad t_{\perp} = M_r^2 t$$

یکی از خواص این روش این است که روابط و موجودیت‌ها می‌توانند هم بعد نباشند و مثلاً اگر موجودیت‌ها d بعدی باشند با ضرب ماتریس k در d در وکتورهای موجودیت‌ها می‌توانیم آن‌ها را به فضای k بعدی رابطه‌ها انتقال دهیم.

۲.۲ معیارهای مقایسه

برای مقایسه متدهای معرفی شده برای سیستم‌های پیشنهاد دهنده نیازمند معیارهایی هستیم که پیشنهادهاى داده شده را با هم مقایسه کنیم و کارایی مدل را بسنجیم. در برخی مقالات معیارهای ساده‌ای مثل MAE استفاده شده است ولی در اکثر مقالات جدید از معیارهای زیر برای بررسی کارایی و مقایسه مدل‌ها استفاده شده است. این معیارها برای هر مرتبه پیشنهاددهی که در آن N آیت به کاربر پیشنهاد داده‌ایم به صورت جداگانه محاسبه می‌شود، برای محاسبه کارایی کل سیستم نیز می‌توان از میانگین این مقادیر استفاده کرد.

معیار	توضیح
Precision	در یک مرتبه پیشنهاد دهی چه کسری از آیتم‌هایی که برای کاربر مناسب بوده است را به او پیشنهاد داده‌ایم.
Recall	چه کسری از آیتم‌های پیشنهاد داده شده به کاربر، برای او مناسب بوده است.
F_1	نسبت Precision به Recall.
Hit Rate	در صورتی که در N پیشنهاد داده شده به کاربر آیتم طلایی مورد نظر کاربر موجود باشد مقدار یک و در غیر اینصورت صفر در نظر گرفته می‌شود.
$NDCG^3$	یک اندازه‌گیری استاندارد از کیفیت رتبه بندی است که با توجه به میزان مرتبط بودن موارد مثبت و منفی N آیت بالای لیست رتبه بندی محاسبه می‌شود.

³Normalized Discounted Cumulative Gain

انتشار فعالیت

۱.۳ مقدمه

در اکثر سیستم‌های پیشنهاد دهنده متدهای مورد استفاده علاقه مندی کاربران را فقط از روی منابعی که از قبل مشخص هستند به دست می‌آورند، مثلاً در فیلتر مشارکتی فقط از امتیازدهی مستقیم کاربران به آیتم‌ها استفاده می‌شود. از آنجایی که استفاده از منابع از پیش تعیین شده باعث ایجاد محدودیت روی کیفیت نتیجه پیشنهاد دهی در سیستم‌ها می‌شود در این مقاله متد جدیدی ارائه شده که قادر است با یکی کردن منابع ناهمگون^۱ اطلاعات، کیفیت پیشنهاد دهی را در سیستم‌ها افزایش دهد.

برای بالا بردن کیفیت پیشنهاد دهی باید همه منابع اطلاعاتی موجود را در آن دخیل کرد، مثلاً برای اشاره به برخی منابع مهم در سیستم‌ها می‌توان به علایق کاربران روی آیتم‌ها، ویژگی‌های آیتم‌ها، ویژگی‌های کاربران، شبکه‌های اجتماعی، تعاملات کاربر با سیستم و ... اشاره کرد. در این مقاله سعی شده است که یک راهکار کلی برای حل این مدل مسائل داده شود و این راهکار باید به اندازه‌ای قابل تعمیم باشد که بتواند اطلاعات گوناگون از منابع اطلاعاتی مختلف را ادغام کرده و برای اهداف خود استفاده کند. پس هدف ارائه روشی برای ارائه اطلاعات است که می‌تواند به عنوان پایه‌ای پایدار برای تدوین سیستم‌های پیشنهاد دهنده عمومی‌تر عمل کند.

در این مقاله از تکنیکی به نام انتشار فعالیت روی گراف ساخته شده توسط اطلاعات استفاده شده است، در این مقاله از تکنیکی به نام انتشار فعالیت^[۲] روی گراف ساخته شده توسط اطلاعات استفاده شده است، گرچه این تکنیک ایده جدیدی نیست ولی تا قبل از آن در موضوع مشابه دیگری استفاده نشده است.

۲.۳ گراف

یکی از مهمترین نکات در این تحقیقات این است که تلاش شده گراف استفاده شده تا حد ممکن کلی باشد؛ در اینصورت این گراف شانس این را دارد که در ادامه حتی با مدل‌های محاسباتی دیگر نیز قابل استفاده باشد. در این مدل تلاش شده است که وزن‌های تطبیقی^۲ برای گراف استفاده شود تا بتوان این گراف را با ورود داده‌های جدید تطابق داد، از طرفی سعی شده که همه اطلاعات ممکن را به این گراف اضافه کرد تا هیچ محدودیتی روی پیشنهاددهی اعمال نکند.

¹Heterogeneous

²adaptive

۱.۲.۳ نماد گذاری

گراف استفاده شده در این سیستم که به صورت یک ابرگراف وزن دار است به این صورت معرفی می شود:

$$K = (N, E, T_N, T_E, t_N, t_E, w_E, A, a_n, a_e) \quad (۱.۳)$$

در این نمادگذاری N مجموعه گره های موجود در گراف را نشان می دهد، مجموعه یال های بین گره ها نیز با $E \subseteq \{u, v\} | u \in N \wedge v \in N$ نشان داده شده است، T_N نشان دهنده مجموعه انواع گره ها و T_E مجموعه انواع یال ها در گراف است. تابع t_N تابعی است که هر گره را به نوع آن متصل می کند، همچنین t_E تابعی است که نوع هر یال را در گراف مشخص می کند و w_E وزن هر یال در گراف را نشان می دهد. در نهایت A مجموعه صفاتی است که می توان به هر گره یا یال نسبت داد و a_n تابعی است که به هر گره صفات مورد نظر را تخصیص می دهد و به همین صورت a_e تابعی است که این صفات را به یال های گراف نسبت می دهد.

انواع مختلف گره موجودیت های مختلف در گراف را مدل می کند و انواع مختلف یال ها روابط مختلف بین موجودیت ها را مدل می کند. این پایگاه دانش طوری طراحی شده است که به اندازه کافی انعطاف پذیر باشد که بتواند اطلاعات موجود در هر برنامه ای را به نمایش بگذارد و هدف از طراحی آن معرفی یک متد برای تعریف یک چارچوب نمایش اطلاعات و همینطور فراهم کردن متدهای محاسباتی روی این چارچوب برای استفاده هر برنامه ای بوده است.

۳.۳ پیشنهاد دهی

یکی از مشکلات مهم فیلتر مشارکتی شروع سرد است زیرا در این متد تنها از تعاملات کاربر برای پیشنهاددهی استفاده می شود، حتی مدل های محتوا محور نیز برای توصیه کردن نیاز به اطلاعات دارند تا بتوانند سلیقه ی کاربر را مدل کنند. هدف در این مدل ارائه پیشنهاددهای مناسب در اولین فرصت ممکن به کاربر است. برای رسیدن به این هدف، تا جای ممکن داده های موجود را استفاده کرده و مدل محاسباتی ای ارائه شده است که از داده های ناهمگون اطلاعات مفید استخراج می کند و همینطور این استراتژی پوشش^۳ مسئله را نیز تضمین می کند. منظور از پوشش مسئله این است که متد توصیه گر تا مرحله مربوطه در چند درصد حالت ها توانسته است تخمین مناسبی برای کاربران بزند.

در ادامه به معرفی و توضیح تکنیک انتشار و استفاده از آن روی گراف معرفی شده می پردازیم. با استفاده از این تکنیک، مدل می تواند از ترکیب مسیرهای مختلف بین گره مبدا و مقصد برای نتیجه گیری استفاده کند. این متد به طول مسیر بین مبدا و مقصد وابسته است و این موضوع باعث می شود که از گره هایی که از مبدا و مقصد دور هستند در نتیجه نهایی بی تاثیر باشند.

۱.۳.۳ انتشار

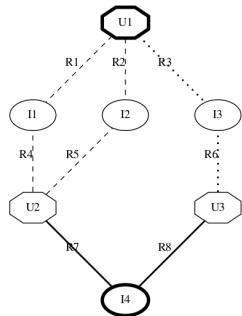
پخش یا انتشار یک فرایند مرحله به مرحله است. در مرحله اول مقدار فعال سازی در گره های مبدا برابر یک مقدار ثابت خواهد شد، (معمولاً مقدار یک) گره های مبدا گره هایی هستند که می خواهیم برای آن ها پیش بینی انجام دهیم و در ساده ترین حالت فقط شامل یک گره است مثلاً گره مربوط به یک کاربر که قصد توصیه کردن به او را داریم.

در هر مرحله مقداری از فعال سازی مربوط به هر گره بر روی گره های همسایه اش پخش می شود و قسمتی از آن نیز در گره باقی می ماند. پس از اتمام پخش در تمام گره ها، مقدار فعال سازی در هر گره برابر است با مجموع مقادیری که از گره های اطراف دریافت کرده است. دو پارامتر تنظیم کننده در این فرایند وجود دارند که یکی از آن ها مقدار فعال سازی پخش شده برای هر گره و دیگری مقدار فعال سازی باقیمانده در هر گره را مشخص می کند. مقدار فعال سازی روی همه یال ها به نسبت وزن آن یال پخش می شود. در صورتی که بخواهیم در شبکه مقدار فعال سازی همیشه ثابت باشد باید مجموع دو

^۳Coverage

متغیر معرفی شده برا هر گره برابر یک باشد.

در این متد از یک متغیر استفاده شده است که با استفاده از آن می‌توان آستانه‌ای برای مقدار فعال‌سازی هر گره تعیین کرد که در صورتی که مقدار فعال‌سازی از این آستانه کمتر باشد مقدار صفر برای آن در نظر گرفته خواهد شد.

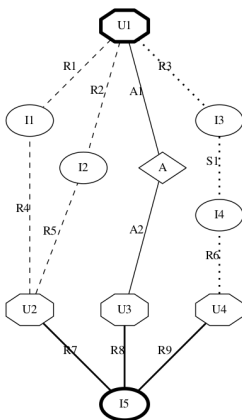


معیارهای متفاوتی را می‌توان برای پایان پروسه این الگوریتم در نظر گرفت به طور مثال می‌توان تعداد مراحل طی شده را در نظر گرفت و بعد از رسیدن به آن مرحله پخش فعال‌سازی به پایان برسد. راهکار دیگر برای پایان دادن استفاده از روش دلتا است به معنی که آنقدر پخش انجام گیرد که در مرحله بعد تغییر چندانی در مقدار فعال‌سازی در هر گره احساس نشود. در ادامه از روش اول برای پایان دادن به عملیات استفاده شده است زیرا برای پیشنهاددهی بهتر از از گره‌هایی که به گره مبدا و مقصد نزدیکتر هستند استفاده کرد و باید به این نکته هم توجه کنیم که استفاده از روش دلتا منابع زیادی از سیستم را استفاده خواهد کرد که برای این هدف مطلوب نیست.

۲.۳.۳ مقایسه با فیلتر مشارکتی

در این قسمت کارکرد فیلتر مشارکتی روی گراف ساده کاربر-آیتم و کارکرد متد انتشار فعالیت روی گراف توصیف شده در قسمت‌های قبل را بررسی و مقایسه می‌کنیم.

تصویر ۱.۳ یک سناریوی ساده از فیلتر مشارکتی را نمایش می‌دهد که در آن گره‌ها با برجسب U کاربران و گره‌ها با برجسب I آیتم‌ها را در گراف نمایش می‌دهند. یال‌های مشخص شده در گراف، امتیاز دهی یک کاربر روی یک آیتم خاص را نشان می‌دهد. در این مثال می‌خواهیم علاقه‌مندی کاربر U1 را روی آیتم I1 اندازه‌گیری کنیم که برابر مجموع وزن‌دار یال‌های R4 و R5 خواهد بود. برای تعیین وزن هر یال باید میزان مشابهت علاقه‌مندی کاربر U1 و کاربر مورد نظر را بسنجیم مثلاً با توجه به اشتراکات بین U1 و U2، وزن یال R4 و تحت تاثیر وزن یال‌های R1، R2، R3 و R5 خواهد بود.



در مقایسه در تصویر ۲.۳ گراف معرفی شده در مراحل قبل را می‌بینیم و می‌خواهیم نحوه استفاده از اطلاعات ناهمگون در این گراف را بررسی کنیم. در این گراف گره‌ها با برجسب U کاربران، گره‌ها با برجسب I آیتم‌ها و گره‌ها با برجسب A رده‌های زمانی را نمایش می‌دهد. رده زمانی یک مثال برای آن است که بفهمیم چطور از ویژگی‌های معرفی شده برای کاربر یا آیتم می‌توان در گراف استفاده کرد. در این مثال پیشنهاد دهی به کاربر U1 از روی مشابهت نسبت به کاربران U2، U3 و U4 و امتیاز دهی آن‌ها انجام می‌شود. به طور مثال کاربر U1 و U2 روی یک آیتم یکسان در گذشته نظر داده‌اند که این موضوع روی وزن R4 در تخمین نهایی محاسبه می‌شود، کاربر U1 و U3 در یک رده سنی هستند که با اتصال به گره از نوع A مشخص شده است و این موضوع در وزن R8 تاثیر خواهد داشت و در نهایت کاربران U1 و U3 در گذشته به آیتم‌هایی که مشابه بودند نظر داده‌اند که این موضوع نیز در نتیجه نهایی لحاظ می‌شود. در هنگام پخش فعال‌سازی روی گراف در نهایت مجموع فعال‌سازی پخش شده برای هر یال محاسبه می‌شود که این مقدار را با a نمایش می‌دهیم. در نهایت برای تخمین امتیاز از یک میانگین وزن‌دار استفاده می‌کنیم که مقدار a برای هر یال وزن امتیاز متناظر با آن یال را مشخص می‌کند. این متد از فرمول زیر برای محاسبه مقدار امتیاز تخمینی استفاده می‌کند:

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{e \in E_{itemRating} \wedge i \in e \wedge v \in e \wedge i \neq v} (r_{v,i} - \bar{r}_v) a_e}{\sum_{e \in E_{itemRating} \wedge i \in e \wedge v \in e \wedge i \neq v} a_e} \quad (2.3)$$

u کاربری است که می‌خواهیم امتیاز را برای او تخمین بزنیم و $\hat{r}_{u,i}$ مقدار این تخمین است، \bar{r}_u و \bar{r}_v میانگین کل امتیازهای کاربر u و v است

و a_e مقدار فعال‌سازی پخش شده روی یال e است، در واقع هر یالی که از نوع امتیاز دهی باشد ($e \in ItemRating$) و یک سر آن به آیتم مورد نظر متصل باشد و سر دیگر آن به کاربری که امتیازدهی آن مشخص است براساس وزن محاسبه شده در مرحله قبل روی نتیجه نهایی تاثیر خواهد داشت.

می‌توان این عملیات را اینطور تفسیر کرد که مجموع فعال‌سازی جمع شده روی یک یال بین کاربر v و آیتم i در واقع همان میزان مشابهت کاربر u به کاربر v است.

۴.۳ ارزیابی و نتیجه‌گیری

ارزیابی این سیستم به روش ارزیابی با سری زمانی انجام شده است. برای ارزیابی مدل با این متد از داده‌هایی که به ترتیب صعودی زمان مرتب شده و در چند بازه زمانی تقسیم شده‌اند استفاده می‌شود و در هر بار تکرار مراحل زیر اجرا می‌شود:

۱. قسمت بعدی داده برای تخمین برداشته می‌شود

۲. امتیازدهی این قسمت از داده‌ها از سیستم سوال می‌شود

۳. خطا با توجه به داده واقعی حساب شده

۴. در نهایت داده واقعی به سیستم اضافه می‌شود

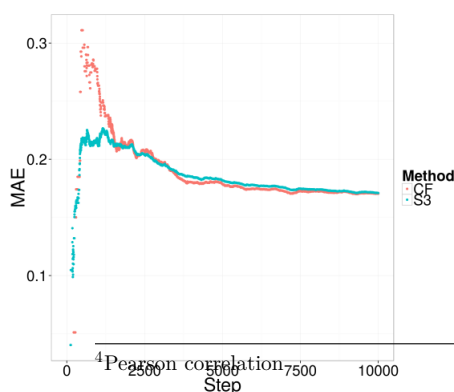
در ابتدای کار پایگاه دانش شامل تمام داده‌های موجود در مورد کاربران و آیتم‌ها است به جز داده‌های امتیاز دهی کاربران به آیتم‌ها که به تدریج در هر تکرار به گراف اضافه می‌شود و اینگونه می‌توانیم یک سناریوی در شرایط واقعی را شبیه‌سازی کنیم و متدهای مختلف را در ابتدای ارزیابی که داده‌ها کم است و اصطلاحاً به آن شروع سرد می‌گوییم مقایسه کنیم.

برای مقایسه بهتر متد معرفی شده با فیلتر مشارکتی متد انتشار فعالیت را با محدودیت گام‌های مختلف آزمایش شده است که نام هرکدام را در تصویر ۳.۳ مشاهده می‌کنید. این ارزیابی روی دیتاست MovieLens با بیش از یک میلیون داده صورت گرفته است.

متد فیلتر مشارکتی با تابع مشابهت پیرسن^۴ در این ارزیابی استفاده شده است.

۱.۴.۳ خطای تخمین

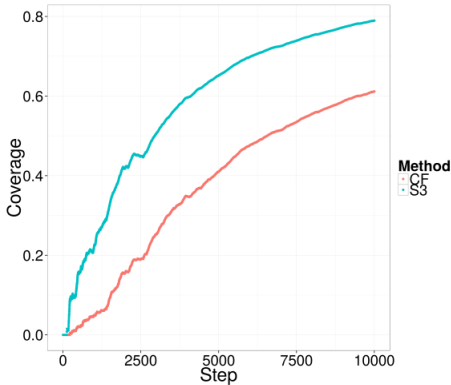
برای مقایسه میزان خطا در ابتدا دو متد CF و S۳ انتخاب شده است، S۳ محدودترین نوع از انتشار فعالیت است زیرا هر چه محدودیت گام کمتر باشد متد برای کشف روابط محدودتر خواهد بود و در نتیجه کارایی کمتری خواهد داشت. در نمودار ۴.۳ دو متد CF و S۳ با هم مقایسه شده‌اند، محور افقی تعداد گام و محور عمودی MAE تا مرحله مشخص شده است. به طور خلاصه با توجه به نمودار مشاهده می‌شود که متد S۳ سریع‌تر همگرا شده و در ابتدای کار یعنی در شرایط شروع سرد پوشش بهتری می‌دهد. همانطور که قبلاً گفته



تصویر ۴.۳: مقایسه MAE در S۳ و CF

شد پوشش درصد حالاتی که متد توانسته است تا مرحله مشخص شده تخمین مناسبی بزند تعریف می‌شود.

۲.۴.۳ پوشش



تصویر ۵.۳: مقایسه پوشش در S3 و CF

نمودار ۵.۳ مقایسه دو متد CF و S3 از نظر پوشش است و نشان می‌دهد که انتشار فعالیت پوشش بهتری نسبت به فیلتر مشارکتی فراهم می‌کند و این تفاوت از همان ابتدای کار قابل مشاهده است چون وقتی اطلاعات واقعی در گراف کمتر است متد معرفی شده خیلی بهتر از فیلتر مشارکتی عمل می‌کند. این مسئله را می‌توان اینگونه توجیه کرد که می‌توانیم پوشش مسئله را مشابه یافتن مسیر بین دو گره در گراف در نظر بگیریم، چون فیلتر مشارکتی محدود به استفاده از یال‌های امتیازدهی است در ابتدا که داده‌ای وجود ندارد نمی‌تواند مسیری پیدا کند در صورتی که متد معرفی شده برای یافتن مسیر از هر نوع رابطه بین کاربران و آیتم‌ها در گراف استفاده می‌کند.

در جدول ۶.۳ می‌توانیم پوشش و MAE را برای مدل‌های مختلف مشاهده کنیم. قابل مشاهده است که MAE در این مدل‌ها تفاوت چندانی ندارد و می‌توان اینطور گفت که این متد با استفاده از داده‌های ناهمگون در پیشنهاددهی توانسته است خطایی در حد فیلتر مشارکتی داشته باشد در صورتی که پوشش آن بالاتر است. البته همانطور که مشاهده می‌شود مقدار بیشتر محدودیت گام باعث افزایش پوشش نبوده است این پارامتر باید به درستی و با توجه به کاربرد آن در مدل تنظیم گردد و باید به این موضوع توجه کرد که پارامترهای انتخاب شده برای مدل‌های دیگر باید به صورت جداگانه تنظیم شوند.

۳.۴.۳ نتیجه

Engine name	Coverage	MAE
CF	6118	0,1704
S3	7897	0,1710
S4	7897	0,1710
S5	7910	0,1711
S6	7910	0,1708
S7	7910	0,1707
S8	7910	0,1705

همانطور که در قسمت‌های قبل مشاهده کردیم خطای تخمین این متد تقریباً برابر فیلتر مشارکتی بود ولی چون این متد از داده‌های مختلفی برای پیشنهاددهی استفاده می‌کند پوشش بهتری نسبت به متد معیار خواهد داشت. با مقایسه همین متد با حدود گام‌های متفاوت مشخص شد که با افزایش گام‌ها با اینکه پوشش افزایش می‌یابد ولی خطای تخمین تقریباً ثابت است از این مطلب می‌توان اینگونه برداشت کرد که در صورتی که به دنبال تخمین امتیاز برای یک آیتم خاص هستیم، زمانی که در یک مرحله به آن آیتم رسیدیم می‌توان در همان مرحله جستجو را پایان داد چون با افزایش تعداد گام خطای تخمین را نمی‌توانیم کاهش دهیم.

ویژگی دیگر این متد همگرایی سریعتر آن است، می‌توان این ویژگی را اینگونه توجیه کرد که چون این متد از تعداد بیشتری امتیازدهی برای تخمین استفاده می‌کند در ابتدای کار یعنی در شرایط شروع سرد این تعداد بالاتر باعث همگرایی کل سیستم خواهد شد. این متد در مراحل نهایی خطایی تقریباً برابر فیلتر مشارکتی دارد پس می‌توان گفت این متد به گونه‌ای طراحی شده است که از داده‌های ناهمگون استفاده کرده و در عین حال خطای پیشنهاددهی در آن افزایش نیافته است.

علاوه بر این پایگاه دانش طراحی شده در این مقاله قابلیت پوشش اطلاعات بیشتری را نیز دارد به طور مثال می‌توان استحکام یک رابطه را با دادن وزن به یال‌های مربوط به آن را مشخص کرد حال در روند استفاده از این گراف دانش می‌توان علاوه بر پیشنهاد دادن براساس داده‌های موجود از داده نهایی صحیح به دست آمده بازخورد گرفت و گراف را در این حین آموزش داد. برای این کار می‌توان از شبکه‌های عصبی استفاده کرد تا به طور مثال بر

اساس بازخورد اهمیت یال‌ها را تغییر داد یا نحوه نمایش گره‌ها را تغییر داد تا نتایج بهتری در ادامه از این گراف دانش قابل استخراج باشد.

اثبات شده است که گراف دانش برای بهبود سیستم‌های توصیه‌گر موثر است. روش‌های موجود عمدتاً به ویژگی‌های دست ساز از گراف‌های دانش (به عنوان مثال ، مسیر ارجح) وابسته است ، که به دانش زمینه نیاز دارد. این مقاله درباره یک رویکرد تعبیه گراف دانش است که به طور خودکار بازنمایی معنایی (یا نمایش معنایی) موجودیت‌ها و مسیرهای بین موجودیت‌ها را برای توصیف (یا تشریح) ترجیحات کاربران، می‌آموزد.

RKGE [۳] از یک معماری جدید شبکه‌ی تکرار شونده^۱ استفاده می‌کند که شامل مجموعه‌ای از شبکه‌های تکرار شونده می‌باشد تا از معناشناسی مسیرهای پیوند دهنده یک جفت موجودیت مشابه، که به طور یکپارچه در پیشنهاددهی استفاده می‌شوند، استفاده کند. علاوه بر این، از یک اپراتور جمع‌آوری برای ایجاد تمیز در احتمال مسیرهای مختلف در توصیف ترجیحات کاربر نسبت به موارد استفاده می‌کند. اعتبار سنجی گسترده در مجموعه داده‌های دنیای واقعی، برتری RKGE را نسبت به روشهای پیشرفته نشان می‌دهد.

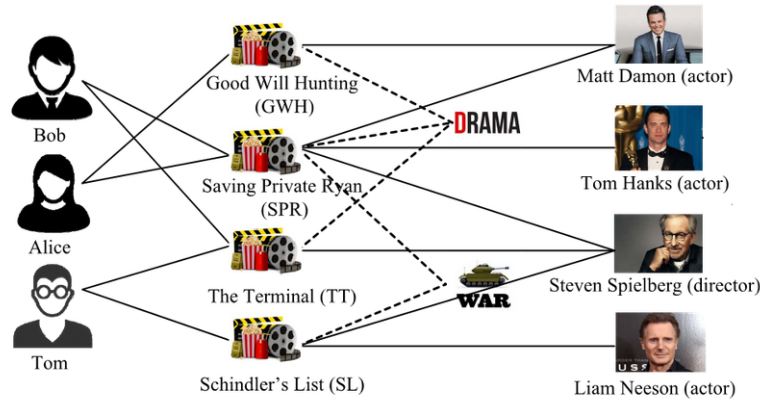
گراف‌های دانش انواع مختلفی از اطلاعات مربوط به آیتم‌ها (به عنوان مثال ، ژانر ، کارگردان ، بازیگر یک فیلم) را در یک فضای سراسری یکپارچه متصل می‌کنند ، این به آسان‌تر شدن بینش در مورد مشکلات پیشنهاددهی که کشف آن‌ها فقط با داده‌های حوجود درباره تعامل کاربر با آیتم‌ها مشکل است ، کمک می‌کند. روش‌های مبتنی بر مسیر ارجح ، برای نشان دادن معانی مسیری بیشتر به دانش زمینه‌ای و ویژگی‌های دست ساز وابسته هستند. مهمتر از همه ، ویژگی‌های دست ساز اغلب برای پوشش تمام روابط ممکن موجودیت ناقص هستند ، بنابراین مانع از بهبود کیفیت پیشنهاددهی می‌شود. برخلاف روش‌های مبتنی بر مسیر ارجح که به ویژگی‌های دست ساز متکی هستند ، روشهای مبتنی بر تعبیه‌ی گراف دانش بطور خودکار تعبیه موجودیت‌ها را در گراف‌های دانش با استفاده از خود شبکه‌ی یک سطحی از موجودیت‌ها با خصوصیاتشان یاد می‌گیرند. در نتیجه ، آن‌ها عملکرد بهتری نسبت به روش‌های مبتنی بر مسیر ارجح کسب کرده‌اند. یک محدودیت عمده در این روش‌ها ، بی‌اعتنایی به روابط معنایی بین موجودیت‌ها است که با مسیرها به هم وصل می‌شوند ، که بطور گسترده در روش‌های مبتنی بر مسیر ارجح مورد مطالعه قرار گرفته است.

بنابراین این مقاله به دنبال یک روش جدید داده محور است که به ویژگی‌های دستی (به عنوان مثال ، مسیر ارجح) وابسته نباشد ، اما بتواند موجودیت و مسیر بین آن‌ها را در نظر بگیرد. با توجه به مثال میتوانیم فیلم ” لیست شیندلر “ که کارگردان آن با ” نجات سرباز رایان “ یکی است یا ” ویل هانتینگ نابغه “ که ژانرش درام است را پیشنهاد کنیم. این مثال نشان می‌دهد که مسیرهای مختلفی که به یک جفت موجودیت مشابه وصل می‌شوند اغلب روابط معنایی مختلف را با هم دارند.

بهره برداری کامل از مسیرها در گراف دانش برای توصیه ، مستلزم آن است که نه تنها معانی مسیرهای مختلف را به دست آوریم بلکه از ویژگی‌های متمایز آن‌ها در توصیف ترجیحات کاربر نسبت به آیتم‌ها استفاده کنیم.

شبکه‌های تکرار شونده قادر به مدل سازی توالی‌هایی با طول های مختلف هستند ، و این امر را برای مدل سازی مسیرها - یعنی توالی تعداد افراد

¹ recurrent network



تصویر ۱.۴: گراف دانش

مختلف - در گراف‌های دانش مناسب می‌کند.

. مهمتر از همه ، شبکه‌های مکرر نه تنها می‌توانند از ویژگی‌های معنادار موجودیت‌ها استفاده کنند (از طریق یک لایه تعبیه کننده) بلکه از روابط موجودیت‌ها نیز با رمزگذاری کل مسیر استفاده می‌کنند ، بنابراین یک رویکرد یکپارچه برای یادگیری بازنمایی‌های موجودیت‌ها و روابط میان موجودیت‌ها ارائه می‌دهند. با توجه به قدرت توصیفی مختلف مسیرها در توصیف ترجیحات کاربر ، مدل سازی کلیه مسیرهای مرتبط در گراف دانش توسط معماری شبکه‌ی تکرار شونده‌ی استاندارد غیرقابل استفاده است.

برای بهره برداری از گراف‌های دانش برای پیشنهاددهی و همچنین برای پرداختن به چالش فوق ، یک چارچوب یکپارچه‌ی تعبیه‌ی گراف دانش تکرار شونده^۲ را پیشنهاد شده است.

آزمایش‌های گسترده در مورد داده‌های دنیای واقعی نشان می‌دهد که RKGE به طور مداوم از پیشرفته‌ترین روش‌ها ۸۴٪.۱۷ در دقت و ۸/۱۱٪ در MRR به طور متوسط بهتر عمل می‌کند.

۲.۴ تعبیه‌ی گراف دانش تکرار شونده

با توجه به داده‌های تعامل کاربر-آیتم ، هدف این مقاله بهره برداری از اطلاعات ناهمگن رمزگذاری شده در گراف دانش برای کمک به یادگیری نمایش‌های با کیفیت بالا از کاربر و آیتم است که از آن‌ها برای تولید پیشنهاددهی‌های بهتر استفاده می‌شود. انتظار می‌رود گراف دانش معانی روابط موجودیت‌ها و موجودیت‌های رمزگذاری شده را به طور کامل ضبط کند. برای دستیابی به این هدف ، این مقاله رویکرد تعبیه‌ی گراف دانش تکرار شونده (RKGE) را پیشنهاد می‌کند. چارچوب کلی در شکل ۲ نشان داده شده است. RKGE ابتدا به صورت خودکار مسیرهای معنایی بین جفت‌های موجود را استخراج می‌کند ، سپس از یک معماری جدید شبکه تکرار شونده استفاده می‌کند تا مسیرهای مختلف را از طریق دسته‌ای از شبکه‌های مکرر رمزگذاری کند. این امر بیشتر از طریق یک عملیات جمع کردن میزان قابل قبول بودن مسیر را تعیین می‌کند.

جدول ۱ نمادهای مورد استفاده در طول این مقاله را خلاصه می‌کند. از کاربر به عنوان $U = [u_1, u_2, \dots, u_m]$ و آیتم به عنوان $V = [v_1, v_2, \dots, v_n]$ و از $R \in Rm \times n$ برای مشخص کردن تعاملات تاریخی کاربر با آیتم استفاده می‌شود و $r_{ij} = 1$ نشان می‌دهد کاربر u_i آیتم v_j را ترجیح میدهد و در غیراینصورت $r_{ij} = 0$ برقرار است. از موجودیت (موجودیت) به عنوان یک اصطلاح عمومی برای اشاره به کلیه اشیاء مرتبط (به عنوان مثال ، کاربر ، مورد ، ژانر ، بازیگر) استفاده شده است که می‌توانند در یک گراف دانش منطبق شوند.

تعریف ۱. نمودار دانش. $E = \{e_1, e_2, \dots, e_k\}$ و $R = \{r_1, r_2, \dots, r\}$ به ترتیب مجموعه‌های موجودیت‌ها و روابط موجودیت‌ها را در

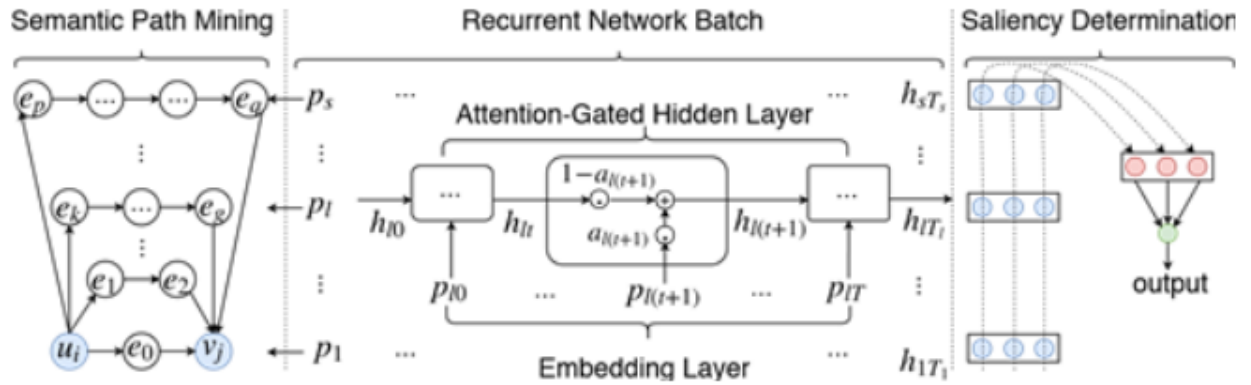
²unified recurrent knowledge graph embedding framework

Notations	Descriptions
$\mathcal{U} = \{u_1, u_2, \dots, u_m\}$	User set
$\mathcal{V} = \{v_1, v_2, \dots, v_n\}$	Item set
$R \in \mathbb{R}^{m \times n}$	User-item interaction matrix
r_{ij}, \tilde{r}_{ij}	Observed and estimated ratings
$\mathcal{E} = \{e_1, e_2, \dots, e_k\}$	Entity set
$\mathcal{R} = \{r_1, r_2, \dots, r_g\}$	Entity relation set
$\mathcal{G} = (\mathcal{E}, \mathcal{L})$	Knowledge graphs
$\mathcal{P}(e_i, e_j) = \{p_1, p_2, \dots, p_s\}$	Paths between entity pair (e_i, e_j)
$p_l = e_0 \rightarrow e_1 \dots \rightarrow e_T$	Path p_l between entity pair
$p_l = \{p_{l0}, p_{l1}, \dots, p_{lT}\}$	Embedding of path p_l
a_{lt}	Attention gate at current step
h_{lt}	Current hidden state
h'_{lt}	Current candidate hidden state
h	Aggregated hidden state
W, H	Linear transformation parameters
σ	Sigmoid activation function
\mathcal{J}	Objective function

تصویر ۲.۴: جدول ۱ (Notations)

نظر نشان می دهند. گراف دانش به عنوان یک نمودار مستقیم $G = (E, L)$ با یک تابع نگاشت از نوع موجودیت تعریف می شود $\phi: E \rightarrow A$ و یک تابع نگاشت از نوع پیوند $\Psi: L \rightarrow R$. هر موجودیت $e \in E$ به یک نوع موجودیت $\phi(e) \in A$ و هر پیوند $l \in L$ به یک نوع پیوند (رابطه) $\Psi(l) \in R$ تعلق دارد و در آخر، از $P(e_i e_j) = \{p_1, p_2, \dots, p_s\}$ برای نشان دادن مسیرهای مرتبط بین موجودیت های e_i و e_j استفاده شده است.

گراف دانش مورد بررسی در این مطالعه می تواند به عنوان یک شبکه اطلاعات ناهمگن در نظر گرفته شود، زیرا بیش از یک نوع موجودیت و رابط موجودیت دارد، یعنی، $|A| > 1$ و یا $|R| > 1$. شکل ۱ یک نمونه شبیه سازی شده از گراف دانش را در حوزه فیلم ارائه می دهد، جایی که موجودیت شامل کاربر، فیلم و ویژگی های مربوطه (مانند ژانر، بازیگر و کارگردان) می شوند، و پیوندها روابط بین موجودات را نشان می دهند (به عنوان مثال، "رتبه بندی" رفتار و رفتار "عمل").



تصویر ۳.۴: چارچوب کلی RKGE، که شرح یک جفت آیت-کاربر است.

۳.۴ استخراج معنای مسیر

برای بهره برداری کامل از روابط موجودیت در گراف دانش، ابتدا مسیرهای با معانی مختلف بین موجودیت‌ها را استخراج می‌کنیم. با توجه به حجم زیاد و پیچیدگی گراف دانش، تعداد زیادی مسیر وجود دارد که جفت موجودیت را به هم متصل می‌کند که ممکن است شامل انواع مختلف موجودیت و انواع رابطه در مرتبه‌های مختلف با طول‌های مختلف باشد. بنابراین برای کمک به انتخاب مسیرهای برجسته استراتژی‌های زیر را در نظر گرفته شده‌اند:

استراتژی ۱: فقط مسیرهای کاربر را به موارد $P(u_i, v_j)$ در نظر گرفته شود که u_i را با همه آیت‌های دارای رتبه بندی وی مرتبط می‌کند، یعنی، $\{v_j | r_{ij} > 0\}$ این مسیرها با توجه به اینکه هدف این است که آیت‌ها به کاربران پیشنهاد شوند برای توصیه کردن مفیدتر هستند. علاوه بر این، آن‌ها مسیرهای مربوط به آیت-آیت و کاربر-کاربر را به عنوان پیامدهای مسیر کاربر-آیت شامل می‌شوند. استراتژی ۲: مسیری را با محدودیت طول مشخص شود، یعنی فقط مسیرهایی با طول کمتر از آستانه استفاده می‌شوند.

با این دو استراتژی، RKGE مسیرهای واجد شرایط را با معانی مختلفی که جفت‌های موجودیت (به عنوان مثال کاربر-مورد) را به صورت خودکار متصل می‌کنند، به جای ویژگی‌های دستی متصل می‌کند. این مسیرها بیشتر توسط مجموعه شبکه تکرار شونده پردازش می‌شوند تا بطور خودکار بازنمایی‌های معنایی آن‌ها را برای توصیه یاد بگیرند.

۴.۴ شبکه ای از شبکه‌های تکرار شونده

با توجه به اینکه مسیر هدایت شده بین جفت کاربر-آیت به عنوان یک دنباله در نظر گرفته می‌شوند شبکه‌هایی راجع به رمزگذاری مسیر در نظر می‌گیریم. این به طور عمده به توانایی آن‌ها در مدل سازی توالی‌هایی با طول‌های مختلف و در گرفتن معنای موجودیت و کل مسیر بین جفت موجود نسبت داده می‌شود. با توجه به اینکه چندین مسیر با طول‌های مختلف ممکن است یک جفت موجودیت را به هم متصل کند، یک معماری شبکه جدید را برای ثبت همه روابط ممکن، که شامل مجموعه ای از شبکه های تکرار شونده است، ابداع شده است. از آنجا که تعداد مسیرهای بین جفت موجودیت‌های مختلف پویا است، تعداد شبکه های تکرار شونده در گروه مطابق با مسیرهای متصل بین جفت موجودیت‌ها متفاوت است. علاوه بر این، تمام شبکه‌های تکرار شونده در این گروه پارامترهای یکسانی دارند تا از تطبیق بیش از حد جلوگیری شود. فرض کنید مسیری با طول‌های مختلف، یک جفت موجودیت را به هم متصل می‌کند (u_i, v_j) ، یعنی، $P(u_i, v_j) = \{p_1, p_2, \dots, p_s\}$ توجه داشته باشید که s پویا است، زیرا ممکن است جفت‌های مختلف موجود با تعداد متفاوتی از مسیرها مرتبط باشند. برای هر مسیر pl با طول T در قالب

$$p_l = e_0 \xrightarrow{1^r} e_1 \xrightarrow{2^r} e_2 \dots \xrightarrow{T^r} e_T \quad (1.4)$$

با

$$e_0 = u_i, e_T = v_j \quad (2.4)$$

شبکه تکرار شونده با یادگیری یک بازنمایی معنایی برای هر موجودیت و یک بازنمایی واحد برای کل مسیر را رمزگذاری می‌کند. در RKGE، این اهداف توسط دو لایه شبکه، یعنی لایه تعبیه سازی و لایه پنهان مورد توجه، انجام می‌شود، همانطور که در شکل ۲ نشان داده شده است. لایه تعبیه سازی. برای هر موجودیت و به صورت pl یک pl_t بازنمایی توزیع شده را یاد می‌گیرد که نگاشت می‌شود و یک بردار با ابعاد کم را نشان می‌دهد، که هر عنصر بردار نشان دهنده میل این نهاد به یک موضوع نهفته است، بنابراین معنای موجودیت را ضبط می‌کند. این منجر به بازنمایی مسیر pl می‌شود $pl = \{pl_0, pl_1, pl_2, \dots, pl_T\}$ که در آن هر عنصر بیانگر نمایش (embedding) نهاد مربوطه در pl است. این نمایه جدید سپس به

عنوان ورودی به لایه پنهان تغذیه می‌شود تا یک نمایه واحد را که کل مسیر را رمزگذاری می‌کند یاد بگیرد.

لایه پنهان با توجه به دروازه برای یادگیری بازنمایی مسیر تعبیه موجودات در مسیر و نظم این موجودات را در نظر می‌گیرد. برای رمزگذاری دنباله از ابتدا مسیر e_0 به انتها e_T یک روش مبتنی بر جریان نیاز دارد: در هر مرحله $t - 1$ ، حالت پنهان $h_{l(t-1)}$ را یاد می‌گیرد که پیامد را از e_0 به $e_t - 1$ کدگذاری می‌کند، سپس به عنوان ورودی همراه با تعبیه e_t (یعنی pl_t) برای یادگیری حالت پنهان مرحله بعدی، یعنی h_{l_t} استفاده می‌شود. حالت نهایی h_{l_t} کل مسیر را رمزگذاری می‌کند، بنابراین به عنوان بازنمایی کل مسیر در نظر گرفته می‌شود. در این مقاله پیشنهاد شده است برای کنترل بهتر جریان اطلاعات از طریق مسیر pl ، از دروازه توجه استفاده شود، که اثبات شده از شبکه‌های عصبی عود مکرر مؤثرتر است در مرحله t با a_{l_t} ، دروازه توجه را نشان داده شده است، که یک مقیاس بین $[0, 1]$ است. حالت پنهان در زمان t به صورت زیر مدل می‌شود:

$$h_{l_t} = (1 - a_{l_t}) \cdot h_{l(t-1)} + a_{l_t} \cdot h'_{l_t} \quad (3.4)$$

جایی که دروازه توجه a_{l_t} سهم ورودی حالت پنهان قبلی $h_{l(t-1)}$ و نامزد فعلی حالت پنهان h'_{l_t} را متعادل می‌کند. وضعیت مخفی نامزد فعلی با وارد کردن کامل ورودی در مرحله زمان فعلی بیشتر ارائه می‌شود:

$$h'_{l_t} = \sigma(W \cdot h_{l(t-1)} + H \cdot p_{l_t} + b) \quad (4.4)$$

که در آن W ، H به ترتیب پارامترهای دگرگونی خطی برای مراحل قبلی و فعلی هستند. b اصطلاح bias است. σ تابع فعال سازی سیگموئید است.

سرانجام، با استفاده از شبکه‌های مکرر دو جهته دروازه توجه به حداکثر رساندن کاوش در دنباله ورودی استنباط می‌شود. از دروازه توجه بر اساس هر دو مشاهده ورودی در مرحله زمانی فعلی و اطلاعات مربوط به مشاهده همسایه در هر دو جهت، فرموله شده توسط

$$a_{l_t} = \sigma M^T(h \rightarrow lt; \leftarrow h_{l_t}) + b \quad (5.4)$$

جایی که σ تابع فعال سازی سیگموئید برای کنترل دامنه دروازه توجه به دامنه $[0, 1]$ است. M بردار وزن است و b' term اصطلاح bias لایه توجه است. (\rightarrow ; \leftarrow) نشان دهنده همبستگی بین بردارها است. h_{l_t} و h'_{l_t} نمایشهای پنهان یک مدل شبکه عود کننده دو جهته هستند، که به عنوان

$$\vec{h}_{l_t} = \sigma(\vec{W} \cdot \vec{p}_{l_t} + \vec{H} \vec{h}_{l(t-1)} + \vec{b}) \quad (6.4)$$

خلاصه اطلاعات متن در طی مرحله زمان t انجام می‌شود، داده شده توسط،

$$h_{l_t} = \sigma(\overleftarrow{W} \cdot p_{l_t} + \overleftarrow{H} \overleftarrow{h}_{l(t-1)} + \overleftarrow{b}) \quad (7.4)$$

بنابراین، h_{l_t} مسیر را از ابتدا تا مرحله t خلاصه می‌کند، در حالیکه $h_{l_t} \leftarrow$ مسیر را از انتهای تا مرحله t خلاصه می‌کند. به طور کلی، با درج هر مسیر واجد شرایط (با تعداد کل S) بین u_i و v_j در شبکه مکرر با توجه به درج مربوط به طور همزمان، نتیجه یک دسته شبکه مکرر است که هر یک از شبکه‌های مکرر مورد توجه، یک مسیر واحد را رمزگذاری می‌کنند. برای جلوگیری از جا به جایی بیش از حد، تمام شبکه‌های مکرر مورد توجه، در این گروه پارامترهای یکسانی دارند. سرانجام، نمایه‌های پنهان همه راه‌ها، یعنی بازنمودهای روابط موجودیت u_i و v_j را بدست می‌آید. اهمیت تفاوت این حالت‌های پنهان در مدل‌سازی روابط موجودیت‌ها با یک عمل جمع مشخص می‌شود.

۱.۴.۴ تعیین برتری

از آنجا که مسیرهای ارتباطی u_i و v_j وجود دارد، ممکن است مسیرهای مختلف در مدل سازی روابط بین آن‌ها نقش‌های مختلفی داشته باشد. به عنوان مثال، کارهای قبلی نشان داده‌اند که مسیرهای کوتاه‌تر ممکن است تأثیر بیشتری نسبت به مسیری طولانی‌تر داشته باشند، زیرا مسیرهای کوتاه‌تر اغلب نشانگر اتصال قوی‌تر با معانی واضح‌تر است. از این رو، این مقاله یک روش داده محور را از طریق عملیات جمع کردن طراحی نموده است تا به تشخیص اهمیت مسیر کمک کند. مکانیسم‌های توجه همچنین به نظر می‌رسد یک راه حل ممکن برای حل این مسئله است. با این حال، به طور کلی هدف این است که اهمیت هر عنصر را در یک دنباله واحد مشخص شود، در حالی که هدف این مقاله تصمیم‌گیری در مورد شایستگی هر مسیر (یعنی دنباله) بین یک جفت موجود است. برای مسیرهای S در $P(u_i, v_j)$ آخرین حالت‌های پنهان آن‌ها که توسط شبکه تکراری یاد می‌گیرند عبارتند از $h_{1T_1} h_{2T_2} \dots h_{sT_s}$ ، جایی که T_s آخرین مرحله p_s و همچنین طول p_s است. براین اساس، یک لایه استخر حداکثر اضافه می‌کنیم تا برجسته‌ترین ویژگی را در تمام مسیرها بدست آوریم. این نتیجه در حالت پنهان جمع ساعت:

$$h[j] = \max(h_i, T_i) \quad 1 \leq i \leq s \quad (۸.۴)$$

که در آن $h[j]$ مقدار بعد از h است. علاوه بر این، برای جلوگیری از تسلط یک h_{iT_i} خاص، به عنوان مثال، یک مسیر واحد در $P(u_i, v_j)$ ، یک عمل جمع کردن متوسط را به سمت آخرین حالت‌های پنهان در تمام مسیرها انجام می‌دهیم. از طریق عملیات جمع کردن، یک حالت پنهان نهایی در تمام مسیرهای بین u_i و v_j ، یعنی اثرات کل مسیرها در رابطه u_i و v_j دریافت می‌کنیم. سپس یک لایه کاملاً متصل را پس از لایه استحکام اتخاذ می‌کنیم تا رابطه (نزدیکی) u_i و v_j ، یعنی r_{ij} ، را تعیین کنیم،

$$\tilde{r}_{ij} = f(h) = \sigma(W_r h + b_r) \quad (۹.۴)$$

که در آن ضریب رگرسیون W_r است و b_r اصطلاح bias است. ما یک تابع σ sigmoid () را برای کنترل دامنه $f(h)$ به $[0, 1]$ اتخاذ می‌کنیم. پس از اتمام آموزش مدل، با رمزگذاری مسیرهای متصل بین آن‌ها از طریق RKGE، نمایش‌های بهتری از u_i و v_j حاصل می‌شود. به دنبال، در طی فرایند آزمون، میزان نزدیکی کاربر u_i و موارد v_k را از طریق محصول داخلی تعبیه‌های مربوطه آن‌ها محاسبه می‌کنیم، یعنی $s(u_i, v_k) = u_i v_k$. از محصول داخلی استفاده می‌شود زیرا از پیش بینی رتبه بندی کاربر از طریق شبکه، که به یک منبع تغذیه وقت گیر در کل شبکه نیاز دارد، کارآمدتر است. در آخر، موارد را بر اساس نمره نزدیکی رتبه بندی می‌کنیم، و موارد برتر K را با بالاترین امتیاز به u_i توصیه می‌کنیم.

۲.۴.۴ بهینه سازی مدل

یادگیری مدل. با توجه به داده‌های آموزش D_{train} ، که شامل مواردی به شکل (u_i, v_j, r_{ij}) است، RKGE پارامترهای درگیر را با به حداقل رساندن عملکرد می‌آموزد:

$$J = \frac{1}{(|D_{train}|)} \sum_{r_{ij} \in D_{train}} BCELoss(\tilde{r}_{ij}, r_{ij}) \quad (۱۰.۴)$$

که در آن BCELoss (۰) آنتروپی صلیبی دودویی بین رتبه بندی‌های مشاهده شده و تخمین زده شده است. بنابراین، مسئله توصیه را به عنوان یک مسئله طبقه بندی باینری مطرح می‌کنیم. از آنجا که معادله ۷ و تمام مازول‌های RKGE از نظر تحلیلی متفاوت هستند، می‌توان آن را به راحتی با روشی انتهایی تا انتها آموزش داد. پارامترها توسط الگوریتم انتشار پشت در طول زمان (BPTT) در لایه‌های مکرر و انتشار عادی برگشت در

قسمت‌های دیگر به روز می‌شوند. به طور تصادفی موارد غیرمجاز را برای هر کاربر به عنوان نمونه های منفی نمونه می‌گیریم، که تعداد آنها با موارد دارای امتیاز وی یکسان است. مسیرهای ارتباط کاربران و موارد منفی آن‌ها نیز برای کمک به تعادل در یادگیری مدل مورد استفاده قرار می‌گیرد. مقیاس پذیری مدل. برای بهبود مقیاس پذیری مدل، یک مکانیزم آموزش موازی دو سطح برای RKGE طراحی شده است. در هر تکرار آموزش مدل، مسیرهای اتصال همه جفت‌های کاربر به موارد برای کاربران جداگانه به RKGE تغذیه می‌شود. در همین حال، پارامترهای مربوط به چندین مسیر بین هر جفت کاربر به موارد به طور همزمان بروزرسانی می‌شوند. در نتیجه، زمان آموزش RKGE در دو مجموعه داده ارزیابی، یعنی IM-MovieLens و Yelp به ترتیب حدود سه ساعت و یک ساعت است. به طور خلاصه، RKGE قابل مقایسه با مجموعه داده‌های بزرگ و KG است.

۵.۴ آزمایش‌ها و تحلیل و بررسی

۱.۵.۴ راه اندازی آزمایشی

برای نشان دادن اثر بخشی توصیه، دو مجموعه داده از دنیای واقعی انتخاب شده است. اولین مجموعه داده IM-MovieLens و دومین مجموعه داده IM-MovieLens IMDB۳. یک مجموعه داده رتبه بندی فیلم شخصی است. دیتابیس دوم Yelp است که Yelp Dataset Challenge است که توسط Yelp منتشر شده است و هم اکنون در Kaggle در دسترس است. Yelp بسیار پراکنده تر از IM-MovieLens است، بنابراین انتظار می‌رود عملکرد بر روی Yelp کاهش یابد.

در راستای ادبیات [۲، ۴۲، ۴۳]، دقت در $N = \{1, 5, 10\}$ یعنی $\text{Prec} @ N$ و top-N به عنوان رتبه متقابل MMR برای معیارهای ارزیابی در نظر گرفته شدند.

$$MRR_N = \frac{1}{m} \sum_{i=1}^m \left(\sum_{v_j \in \text{test}(u_i)} \frac{1}{\text{rank}(u_i, v_j)} \right) \quad (11.4)$$

جایی که $N = 10$ را تنظیم می‌کنیم. v_j مورد صحیح توصیه شده در لیست بالا-N است. $\text{test}(u_i)$ مجموعه موارد موجود در داده های آزمون برای u_i است. مرتبه (u_i, v_j) موقعیت v_j در لیست توصیه های u_i است.

۲.۵.۴ نتایج RKGE

اثرات طول مسیر. مسیرهای با طول‌های مختلف معنای متفاوتی را ضبط می‌کند، که به استنباط ترجیحات کاربر از زوایای مختلف کمک می‌کند و امکان می‌دهد توصیه‌های مختلفی ایجاد شود. فرضیه این است که مسیرهایی با طول‌های نسبتاً کوتاه برای مدل‌سازی رابطه موجودیت سودمندتر هستند زیرا معانی واضح تر و قابل تفسیر دارند. این در روش مبتنی بر مسیر فرا متاثر تأیید شده است. در اینجا بررسی می‌کنیم که آیا همان نتیجه با رویکرد مبتنی بر تعبیه KG وجود دارد یا خیر. برای بررسی تجربی تأثیر طول مسیر بر صحت توصیه، مسیرهایی با طول‌های مختلف، یعنی $L = \{3, 5, 7\}$ را در مدل RKGE پیشنهادی گنجانیده شده است. از آنجا که RKGE در مدل سازی مسیرهای جفت کاربر به موارد قرار دارد، و این مسیرها دارای طول‌های عجیب و غریب هستند زیرا موارد فقط به صورت غیرمستقیم از طریق ویژگی‌های آن‌ها قابل پیوند هستند. شکل ۳ نتایج را در دو مجموعه داده نشان می‌دهد. از نتایج، مشاهده می‌کنیم که با افزایش طول مسیر، عملکرد به تدریج در هر دو مجموعه داده کاهش می‌یابد. این شهود را تأیید می‌کند و یافته‌های قبلی را در زمینه جدید رویکرد مبتنی بر تعبیه KG تأیید می‌کند.

Impacts of Pooling Operations. To determine the saliency of

connected with rated movies by different types of paths, instead

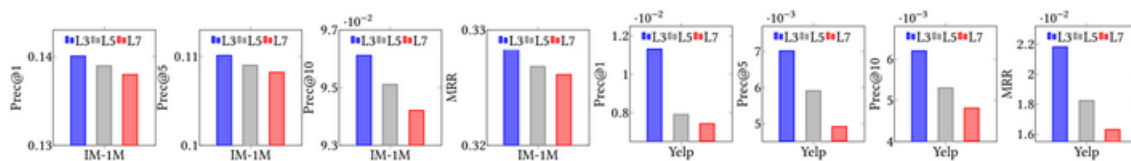


Figure 3: The impacts of different path lengths for RKGE on IM-1M and Yelp datasets.

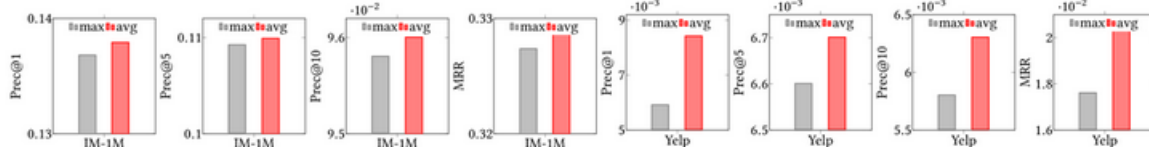


Figure 4: The impacts of different pooling strategies for RKGE on IM-1M and Yelp datasets.

برای درک تأثیر مسیرهای مختلف در توصیف ترجیحات کاربر، دو استراتژی جمع کردن را با هم مقایسه می‌کنیم: اول حداکثر جمع کردن بر مهمترین مسیرها و سپس جمع کردن متوسط است که هدف آن جمع کردن تأثیر همه مسیرها است. از این رو، استحکام متوسط از نتیجه تسلط بر مسیری معین جلوگیری می‌کند. شکل ۴ عملکرد آن‌ها را بر روی دو مجموعه داده نشان می‌دهد، از این رو می‌توانیم توجه داشته باشیم که عملکرد استخر متوسط به طور کلی از عملکرد حداکثر جمع کردن بهتر است. این از شهود پشتیبانی می‌کند که ترجیحات کاربر نسبت به موارد با ترکیبی از عوامل ناهمگن تعیین می‌شود و اهمیت روشی را نشان می‌دهد که می‌تواند از این عوامل استفاده کامل کند. تفسیر RKGE با ضبط کامل معانی موجودیت‌ها و روابط موجودی که در ها KG رمزگذاری شده‌اند، RKGE نه تنها توصیه بهتری را ارائه می‌دهد، بلکه تعامل بهتری نیز برای تعامل مورد با کاربر دارد. برای نشان دادن این موضوع، ابتدا برای هر کاربر موارد رتبه بندی شده خود را (یعنی موارد موجود در داده‌های آموزش) و موارد صحیح توصیه شده (یعنی تقاطع بین موارد موجود در لیست توصیه های برتر او و داده‌های تست) را در KG قرار می‌دهیم و سپس بررسی می‌کنیم که آیا مسیرهای معنایی وجود دارد که آن موارد را پیوند دهد یا خیر. برای نتیجه گیری، در اینجا فقط نتایج یک کاربر نمونه گیری تصادفی را نشان می‌دهیم که به آن باب گفته می‌شود، در شکل ۵. مشاهدات مشابهی را می‌توان برای سایر کاربران در دو مجموعه داده بدست آورد. در شکل ۵، موارد در بالا، موارد رتبه بندی شده باب در داده‌های آموزش است، در حالی که موارد موجود در پایین، موارد صحیح در داده‌های آزمون توصیه می‌شود. برای سادگی، فقط چهار نوع موجودیت، یعنی فیلم، ژانر، بازیگر و کارگردان را در KG نگه می‌داریم. چندین یافته جالب به دست آمده است. اول، فیلم‌های صحیح توصیه شده، همه به فیلم‌های رتبه بندی شده باب یا به صورت ژانر، یا بازیگران، یا کارگردانان مرتبط اند این نشان می‌دهد که RKGE به خوبی می‌تواند اولویت خاص باب را از زوایای مختلف با KG استنباط کند و بر اساس آن توصیه‌های صحیحی را ایجاد می‌کند. دوم، اکثر فیلم‌های صحیح توصیه شده با انواع مختلف مسیرها با فیلم‌های دارای امتیاز مرتبط هستند.

این بدان معنی است که تعامل کاربر-آیتم در مسیرهای مختلف، احتمالاً با درجات مختلف، تحت تأثیر قرار می‌گیرد و تأثیرات مشترک آن‌ها را می‌توان به طور مؤثر توسط RKGE ضبط کرد، و به آن اجازه می‌دهد تا توصیاتی با تفسیر بالا ارائه دهد.

۳.۵.۴ مقایسه نتایج

جدول ۳ عملکرد کلیه روش‌های مقایسه را در دو مجموعه داده در دنیای واقعی خلاصه می‌کند، جایی که برای هر مجموعه داده دو نمایش ایجاد می‌شود: "همه کاربران" نشان می‌دهد که همه کاربران در داده‌های آزمون در نظر گرفته شده‌اند. در حالی که "شروع سرد" بدان معنی است که فقط کاربرانی که کمتر از ۵ رتبه بندی در داده‌های آزمون دارند درگیر هستند. از نتایج این دو دیدگاه می‌توان به تعدادی از مشاهدات جالب توجه اشاره کرد.



تصویر ۴.۴: شکل ۵ یک مثال از RKGE روی IM-۱M

عملکرد در همه کاربران. با توجه به توصیه اصلی، بدون توجه به اطلاعات کمکی، MostPop و BPRMF عملکرد بدتری نسبت به سایر روش‌ها دارند. این به تأیید سودمندی KG برای توصیه کمک می‌کند.

با درج ویژگی‌های مورد در KG به عنوان ویژگی‌های خام، LIBFM عملکرد بهتری نسبت به BPRMF، گاهی اوقات حتی بهتر از روش‌های مبتنی بر KG موجود (به عنوان مثال، HeteRS، HeteRec در Yelp)، نشان می‌دهد برتری آن در استفاده از اطلاعات کمکی برای توصیه‌های موثر است. با وجود این، LIBFM تعامل موجودیت را به روشی خطی مدل می‌کند، بنابراین با قدرت بیانگر آن برای گرفتن الگوهای پیچیده ذاتاً محدود است. روش‌های مبتنی بر شبکه عصبی که به خوبی طراحی شده‌اند، قادر به مدل سازی مناسبات پیچیده موجود هستند، همانطور که توسط عملکرد فیلتر N مشارکتی نشان داده شده است. اگرچه صرفاً با توجه به داده‌های تعامل با مورد کاربر، فیلتر N مشارکتی حتی بهتر از LIBFM عمل می‌کند، که اثربخشی معماری عصبی را نشان می‌دهد.

از نظر روش‌هایی که مخصوص KG طراحی شده است، HeteRec از HeteRS بهتر عمل می‌کند. دلیل احتمالی این است که HeteRS یک روش مبتنی بر نمودار است که بر اساس پیاده روی تصادفی ساخته شده است، بنابراین در گرفتن معنای اشخاص و روابط موجود رمزگذاری شده در KG ناتوان است، در حالی که HeteRec یک رویکرد مبتنی بر مدل فاکتور نهفته است که از ناهمگونی رابطه در KG استفاده می‌کند.

معرفی مسیرهای متا. این تأیید می‌کند که مسیرهای معنایی در KG در واقع تسهیل تولید توصیه‌های مؤثر است. GraphLF همچنین مبتنی بر پیاده روی تصادفی است، اما در عین حال قدرت فاکتورسازی نهفته و استدلال منطقی را در هم می‌آمیزد، بنابراین عملکرد بهتری نسبت به HeteRS و HeteRec به دست می‌آورد. CKE با آموختن بازنمایی‌های معنایی از KGs، بهترین عملکرد را در بین چهار روش مبتنی بر KG موجود (یعنی HeteRS، HeteRec، GraphLF و CKE) انجام می‌دهد و این حاکی از اثربخشی تعبیه شبکه برای توصیه بهتر است. با این حال، روابط موجودات زوج مرتبط با مسیرها را نادیده می‌گیرد، بنابراین در بدست آوردن معنای کامل رمزگذاری شده توسط KG ها موفق نیست.

به طور کلی، هنگامی که با تمام روش‌های مقایسه دیگر در بین دو مجموعه داده مقایسه می‌شود، RKGE پیشنهادی به طور مداوم به بهترین عملکرد دست می‌یابد. پیشرفت‌های w.r.t. دقت و MRR به ترتیب به طور متوسط ۴۲.۲۶ درصد، ۳۰.۲۰ درصد است (آزمون P-value، زوجی t (آزمون

Table 3: Performance of all comparison approaches on IM-1M and Yelp across all the evaluation metrics. The best performance is boldfaced; the runner up is labeled with “*”; the column ‘Improve’ indicates the relative improvements that RKGE achieves w.r.t. the best performance of methods proposed by others.

View	Datasets	Metrics	MostPop	BPRMF	LIBFM	NCF	HeteRS	HeteRec	GraphLF	CKE	RKGE	Improve
AllUsers	IM-1M	Prec@1	0.0118	0.0409	0.0459	0.0450	0.0689	0.0764	0.1069*	0.0954	0.1396	30.58%
		Prec@5	0.0064	0.0438	0.0525	0.0482	0.0528	0.0579	0.0360	0.0781*	0.1092	39.82%
		Prec@10	0.0081	0.0441	0.0456	0.0485	0.0475	0.0488	0.0581	0.0682*	0.0861	26.25%
		MRR	0.0245	0.1234	0.1412	0.1360	0.1600	0.1737	0.1524	0.2440*	0.3056	25.25%
	Yelp	Prec@1	0.0003	0.0051	0.0054	0.0056	0.0047	0.0072	0.0083	0.0084*	0.0113	34.52%
		Prec@5	0.0007	0.0058	0.0059	0.0060*	0.0052	0.0050	0.0054	0.0057	0.0070	16.67%
		Prec@10	0.0004	0.0052	0.0054	0.0055	0.0031	0.0039	0.0056*	0.0053	0.0062	10.71%
		MRR	0.0010	0.0162	0.0167	0.0178	0.0116	0.0151	0.0189*	0.0178	0.0218	15.34%
ColdStart	IM-1M	Prec@1	0.0028	0.0171	0.0330	0.0188	0.0405	0.0573	0.0677	0.0687*	0.0809	17.76%
		Prec@5	0.0017	0.0191	0.0203	0.0210	0.0428	0.0428	0.0267	0.0432*	0.0481	11.34%
		Prec@10	0.0013	0.0205	0.0273	0.0225	0.0370	0.0402	0.0422*	0.0372	0.0467	10.66%
		MRR	0.0050	0.0438	0.0457	0.0481	0.1239	0.1355	0.1188	0.1408*	0.1521	8.03%
	Yelp	Prec@1	0.0004	0.0028	0.0042	0.0031	0.0037	0.0053	0.0061	0.0067	0.0061*	-8.96%
		Prec@5	0.0003	0.0037	0.0040	0.0041	0.0035	0.0035	0.0038	0.0052*	0.0056	7.69%
		Prec@10	0.0003	0.0031	0.0039	0.0034	0.0031	0.0032	0.0047*	0.0043	0.0055	17.02%
		MRR	0.0006	0.0098	0.0104	0.0108	0.0097	0.0113	0.0141	0.0151	0.0149*	-1.32%

تصویر ۵.۴: جدول ۳ عملکرد همه رویکردهای مقایسه

(۱۰.۰) >. این بدان معنی است که با ترکیب مناسب نقاط قوت جاسازی شبکه و استخراج مسیر معنایی در KG می‌توان عملکرد توصیه را بیشتر کرد. عملکرد در شروع سرد. مشاهدات مشابه با ”همه کاربران“ را می‌توان در ”شروع سرد“ مشاهده کرد. همانطور که در تنظیمات قبلی، RKGE به طور قابل توجهی بهتر (p-value < ۰.۱۰) از بهترین روش موجود با ۲۵٪.۹ و ۳۶٪.۳ برای Precision و MRR در IM-۱M بهتر است. در حالی که در Yelp، قضیه کمی متفاوت است. عملکرد RKGE در برخی از معیارها از CKE بدتر است (مثلاً Prec @ ۱، MRR). این احتمالاً به دلیل تراکم نمودار بسیار پایین Yelp در مقایسه با IM-۱M است که منجر به استفاده از مسیر معنایی کافی برای RKGE نمی‌شود. بیشتر از استحکام روش‌های مقایسه شده برای توصیه شروع سرد استفاده می‌کنیم. مشاهده می‌کنیم که اکثر روش‌ها در مقابل کاربران شروع سرد آسیب پذیر هستند. دلیل این امر این است که این روش‌ها ترجیح کاربر را بر اساس تعامل‌های تاریخی می‌آموزند که حاوی اطلاعات بسیار محدودی برای کاربران شروع سرد است. جالب اینجاست که CKE و RKGE به طور مداوم از سایر روش‌ها پیشی می‌گیرند و دلالت بر استحکام تعبیه KG برای کاربران شروع سرد می‌کند. RKGE بیشتر از CKE پیشی می‌گیرد، نشان می‌دهد که استفاده از معانی مسیر توسط RKGE می‌تواند ترجیحات کاربران را حتی از سوابق محدود تعامل تاریخی آن‌ها ضبط کند. علاوه بر این، همچنین متوجه می‌شویم که نسبت‌های بهبود RKGE در ”همه کاربران“ نسبت به ”شروع سرد“ بزرگتر است. این امر می‌تواند با این واقعیت توضیح داد که متفاوت از سایر موارد مربوط به شروع سرد، در حالی که کاربران شروع سرد دارای مقدار زیادی از اطلاعات خارجی به عنوان کاربران شروع به کار هستند، به توصیه‌های KG، کاربران شروع سرد تعداد بسیار محدودی از مسیرهای پیوند دهنده اشخاص را دارند. بنابراین، توصیه با KG برای کاربران شروع سرد مشکل تر است.

۶.۴ نتیجه

نمودار دانش به دلیل اثربخشی آن در تقویت عملکرد توصیه ، مورد توجه جامعه توصیه قرار گرفته است. در این مقاله یک چارچوب تعبیه دانش دانش - RKGE - با یک معماری جدید شبکه تکراری برای توصیه با کیفیت بالا ارائه شده است. RKGE نه تنها بازنمایی معنایی انواع مختلف موجودات را می آموزد بلکه به طور خودکار روابط موجودات رمزگذاری شده در KG را ضبط می کند. اعتبار سنجی گسترده در دو مجموعه داده در دنیای واقعی ، برتری RKGE را نسبت به آخرین روش های پیشرفته نشان می دهد. برای کار در آینده ، با در نظر گرفتن طبقه بندی انواع موجودیت ها در KG ، RKGE را گسترش می دهیم.

RippleNet

۱.۵ مقدمه

رشد بسیار سریع محتوا و خدمات آنلاین از جمله اخبار، فیلم، موسیقی، رستوران‌ها و کتاب‌ها انتخاب‌های سختی را در اختیار کاربران قرار می‌دهد. سیستم‌های توصیه‌گر قصد دارند با پیدا کردن مجموعه‌ای کوچک از آیتم‌ها برای کاربران به منظور برآورده کردن علایق شخصی‌شان، به رشد سریع اطلاعات بپردازند. از میان استراتژی‌های پیشنهادی، فیلتر مشارکتی به موفقیت بزرگی دست یافته است. با وجود این، روش‌های مبتنی بر فیلتر مشارکتی معمولاً از پراکندگی تعاملات کاربر-آیتم و مسئله شروع سرد رنج می‌برند. محققان به منظور بررسی و رفع این محدودیت‌ها و بهبود عملکرد توصیه، تلفیق اطلاعات جانبی با فیلتر مشارکتی از جمله شبکه‌های اجتماعی، خصوصیات کاربر-آیتم، تصاویر و زمینه‌ها را پیشنهاد کرده‌اند. این مقاله در میان انواع مختلف اطلاعات جانبی، گراف دانش را به دلیل اینکه معمولاً شامل حقایق و ارتباطات بسیار سودبخش دربارهٔ آیتم‌ها است، به عنوان منبع اطلاعات جانبی در نظر می‌گیرد.

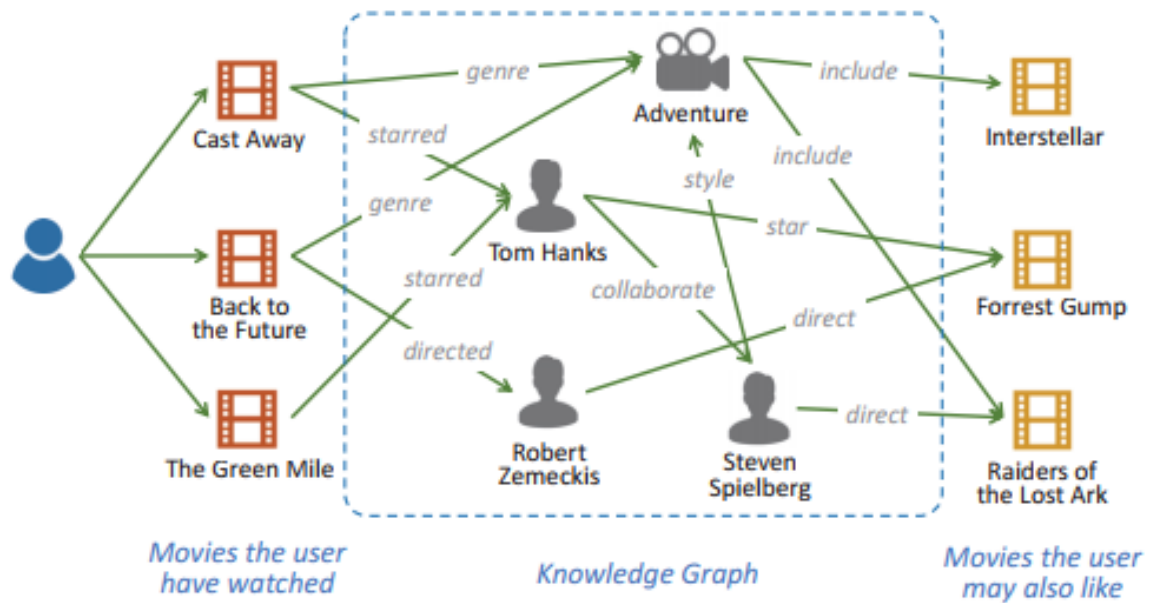
این مقاله برای پرداختن به محدودیت‌های روش‌های موجود، RippleNet [۴] را پیشنهاد می‌دهد که یک چارچوب انتها به انتها برای توصیه آگاهانه گراف دانش است. RippleNet برای پیش‌بینی نرخ کلیک^۱ (CTR) طراحی شده است، که یک جفت کاربر-آیتم را به عنوان ورودی می‌گیرد و احتمال جذب کاربر به آیتم (مثلاً کلیک کردن یا مرور) را به عنوان خروجی می‌دهد. ایده اصلی RippleNet، انتشار اولویت است: بدین صورت که RippleNet علایق گذشته هر کاربر را به عنوان موجودیت‌هایی در گراف دانش در نظر می‌گیرد که در واقع هسته اصلی کار را تشکیل می‌دهند، سپس علایق کاربر را به طور تکراری در امتداد پیوندهای گراف دانش گسترش می‌دهد تا علایق بالقوه سلسله مراتبی کاربر را با توجه به آیتم مورد نظر کشف کند. ما انتشار اولویت را به امواج واقعی که با انتشار قطرات باران بر روی آب ایجاد شده‌اند، تشبیه می‌کنیم که در آن امواج چندگانه برای تشکیل توزیع اولویت حاصله از کاربر بر روی گراف دانش قرار می‌گیرند. تفاوت عمده RippleNet با مقالات موجود در این است که RippleNet مزایای دو نوع روش مبتنی بر تعبیه سازی و مبتنی بر مسیر ارجح را با هم ترکیب می‌کند:

۱. RippleNet روش‌های KGE را به وسیله انتشار اولویت با توصیه یکی می‌کند.

۲. RippleNet می‌تواند به طور خودکار و بدون هیچ نوع طراحی دستی، مسیرهای احتمالی از آیتم موجود در تاریخچهٔ یک کاربر را به آیتم مورد نظر پیدا کند.

به لحاظ تجربی، RippleNet را در سه سناریوی واقعی از پیشنهادات فیلم، کتاب و اخبار بکار گرفته‌اند. نتایج آزمایش نشان می‌دهد که RippleNet در پیشنهادات فیلم، کتاب و اخبار، افزایش AUC را کسب کرده است که با آخرین معیارها برای پیشنهادات مقایسه شده است. همچنین دریافتیم که RippleNet دیدگاه جدیدی از توجیه‌پذیری را برای نتایج توصیه شده در مورد گراف دانش ارائه می‌دهد.

¹Click-through Rate



تصویر ۱۰۵: نمایی از گراف دانش در سیستم‌های توصیه‌گر فیلم

به طور خلاصه، کارهای انجام شده در این مقاله به شرح زیر است:

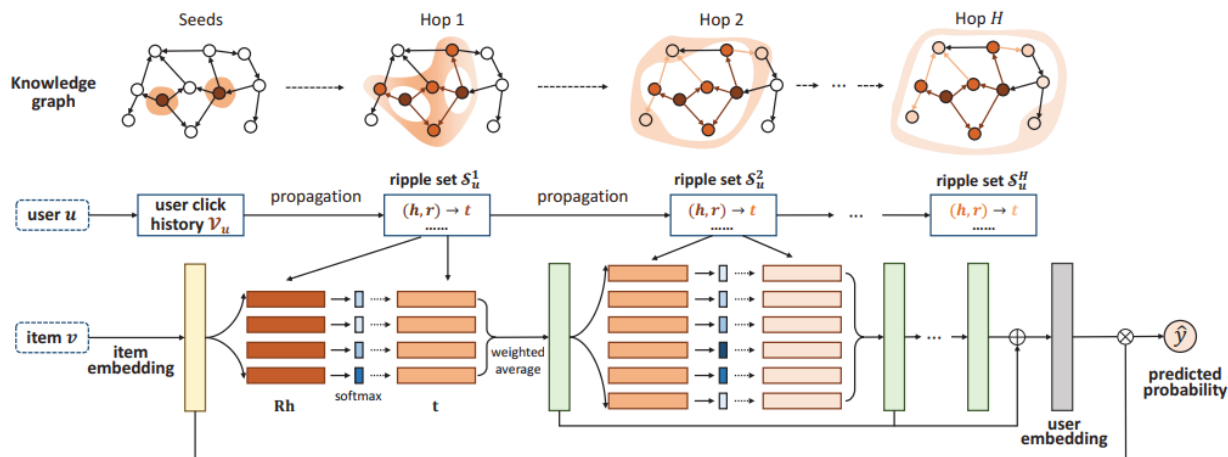
- براساس دانسته‌های ما، این اولین کار برای ترکیب روشهای مبتنی بر تعبیه سازی و مبتنی بر مسیر ارجح در توصیه آگاهانه گراف دانش است.
- RippleNet را پیشنهاد داده که چارچوبی انتها به انتها است که از گراف دانش برای کمک به سیستم‌های توصیه‌گر استفاده می‌کند. RippleNet
- علائق بالقوه سلسله مراتبی کاربران را بطور خودکار و با تکرار انتشار ترجیحات کاربران در گراف دانش کشف می‌کند.
- آزمایش‌ها را در سه سناریو پیشنهادات واقعی انجام داده و نتایج، اثربخشی RippleNet را طی چندین معیار پیشرفته ثابت می‌کند.

مسئله پیشنهاد آگاهانه گراف دانش به شرح زیر طرح می‌شود. در یک سیستم توصیه‌گر معمولی، $U = \{u_1, u_2, \dots\}$ و $V = \{v_1, v_2, \dots\}$ است که به ترتیب مجموعه ای از کاربران و آیتم‌ها را بیان می‌کنند. ماتریس تعامل کاربر-آیتم $Y = \{y_{uv} | u \in U, v \in V\}$ است که با توجه به بازخورد ضمنی کاربران تعریف می‌شود که:

$$y_{uv} = \begin{cases} 1, & \text{if interaction } (u, v) \text{ is observed;} \\ 0, & \text{otherwise.} \end{cases} \quad (1.5)$$

مقدار ۱ برای y_{uv} نشان می‌دهد که تعاملی ضمنی بین کاربر u و آیتم v وجود دارد، ازجمله رفتارهایی چون کلیک، تماشا، مرور، و غیره. علاوه بر ماتریس تعامل Y ، گراف دانش G نیز داریم که شامل سه گانه (h, r, t) می‌باشد.

باتوجه به ماتریس تعامل Y و نیز گراف دانش G ، قصد داریم پیش‌بینی کنیم که آیا کاربر u علاقه بالقوه ای به آیتم v که قبل از آن هیچ تعاملی با آن نداشته است، دارد یا خیر. هدف ما یادگیری تابع پیش‌بینی $\hat{y}_{u,v} = F(u, v; \Theta)$ است، که $\hat{y}_{u,v}$ این احتمال را نشان می‌دهد که کاربر u بر روی آیتم v کلیک خواهد کرد و Θ پارامترهای مدل تابع F را نشان می‌دهد.



تصویر ۲.۵: چارچوب کلی RippleNet. گراف‌های دانش بخش فوق، مجموعه موج‌های متناظر را نشان می‌دهند که با سابقه کلیک کاربر فعال شده اند.

۲.۵ RippleNet

در این بخش، به طور مفصل درباره RippleNet پیشنهادی بحث می‌شود. همچنین در مورد مدل بحث هایی ارائه داده می‌شود.

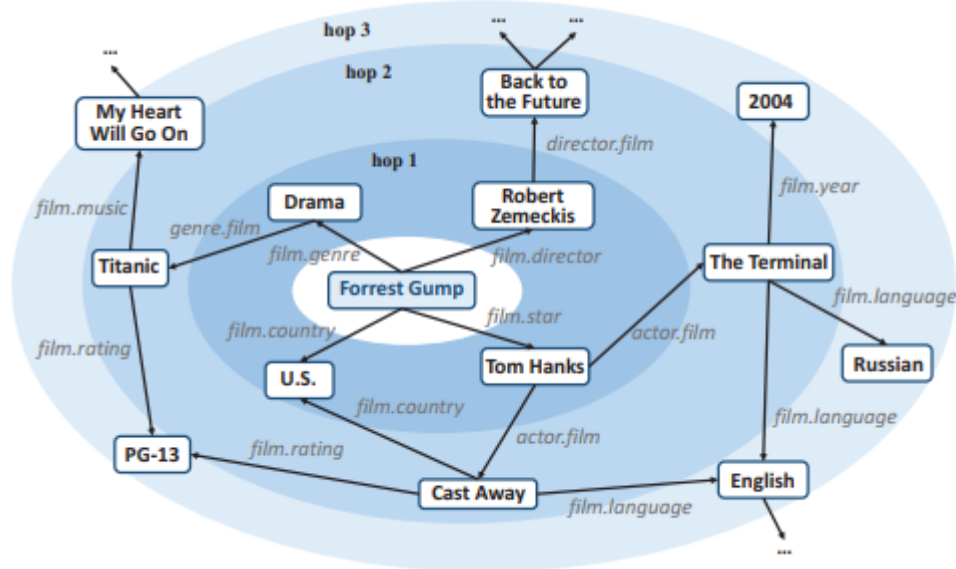
۱.۲.۵ چارچوب RippleNet

چارچوب RippleNet در تصویر ۲.۵ نشان داده شده است. RippleNet کاربر u و آیت v را به عنوان ورودی در نظر می‌گیرد و احتمال پیش‌بینی شده را که کاربر u بر روی آیت v کلیک خواهد کرد را به دست می‌آورد. برای ورودی کاربر u ، مجموعه علائق تاریخی وی به عنوان مجموعه مرکزی در گراف دانش در نظر گرفته می‌شوند، سپس در امتداد پیوندها گسترش می‌یابد تا چندین مجموعه موج $S_k^u (k = 1, 2, \dots, H)$ تشکیل شود. مجموعه موج S_k^u مجموعه ای از دانش‌های سه گانه است که k لایه به دور از مجموعه هسته V_u هستند. این مجموعه‌های موج برای تعامل با آیت تعبیه شده (بلوک زرد) به طور تکراری برای بدست آوردن پاسخ‌های کاربر u با توجه به آیت v (بلوک‌های سبز) مورد استفاده قرار می‌گیرند که سپس برای تشکیل نهایی تعبیه کاربر، ترکیب می‌شوند (بلوک خاکستری). در نهایت، از تعبیه کاربر u و آیت v با هم استفاده می‌کنیم تا احتمال پیش‌بینی شده $\hat{y}_{u,v}$ را محاسبه کنیم.

۲.۲.۵ مجموعه موج^۲

گراف دانش معمولاً حاوی حقایق و ارتباطات مثرتر در میان موجودیت ها است. به عنوان مثال، همانطور که در تصویر ۳.۵ نشان داده شده است، فیلم «Forrest Gump» با «Robert Zemeckis» (کارگردان)، «Tom Hanks» (ستاره)، «U.S.» (کشور) و «Drama» (ژانر) مرتبط است، در حالی که «Tom Hanks» بیشتر با فیلم‌های «The Terminal» و «Cast Away» که در آن بازی کرده است، پیوند دارد. این ارتباطات پیچیده در گراف دانش، چشم‌انداز عمیق و نهفته‌ای را برای کشف ترجیحات کاربر ترسیم می‌کند. به عنوان مثال، اگر کاربر تا به حال «Forrest Gump» را تماشا کرده است، احتمالاً هوادار Tom Hanks شده و علاقه‌مند به «The Terminal» و «Cast Away» می‌شود.

²Ripple Set



تصویر ۳.۵: نمایی از مجموعه موج‌های "Forrest Gump" در گراف دانش فیلم. دایره‌های متحدالمرکز مجموعه موج‌ها را با لایه‌های مختلف نشان می‌دهد. توجه داشته باشید که مجموعه موج‌ها از لایه‌های مختلف لزوماً در عمل از هم جدا نیستند.

در RippleNet، برای توصیف ترجیحات گسترش‌یافته و سلسله مراتبی کاربران برحسب گراف دانش، به صورت بازگشتی مجموعه‌ای از موجودیت‌های مرتبط با لایه k ام برای کاربر u را به شرح زیر تعریف می‌کنیم:

تعریف ۱ (موجودیت وابسته) با توجه به ماتریس تعامل Y و گراف دانش G ، مجموعه‌ای از موجودیت‌های مربوط به لایه k ام برای کاربر u به صورت زیر تعریف می‌شود:

$$\varepsilon_u^k = \{t | (h, r, t) \in G \text{ and } h \in \varepsilon_u^{k-1}\}, \quad k = 1, 2, \dots, H \quad (2.5)$$

که در آن $\varepsilon_u^k = V_u = \{v | y_{uv} = 1\}$ مجموعه‌ای از آیتم‌های کلیک شده کاربر در گذشته است که می‌تواند به عنوان مجموعه مرکزی کاربر u در گراف دانش در نظر گرفته شود.

موجودیت‌های وابسته را می‌توان به عنوان بسط‌هایی طبیعی از علایق گذشته کاربر با توجه به گراف دانش در نظر گرفت. با توجه به تعریف موجودیت‌های وابسته، بنابراین مجموعه موج لایه k ام کاربر u را به شرح زیر تعریف می‌کنیم:

تعریف ۲ (مجموعه موج). مجموعه موج لایه k ام از کاربر u به عنوان مجموعه سه‌گانه دانش تعریف می‌شود که از ε_u^{k-1} شروع می‌شود:

$$S_u^k = \{(h, r, t) | (h, r, t) \in G \text{ and } h \in \varepsilon_u^{k-1}\}, \quad k = 1, 2, \dots, H \quad (3.5)$$

واژه "موج" دو معنی دارد:

۱. مشابه با امواج واقعی چندگانه ایجاد شده توسط قطرات باران، علاقه بالقوه کاربر به موجودیت‌ها توسط ترجیحات گذشته وی فعال می‌شود، سپس توسط پیوندهای گراف دانش به صورت لایه به لایه، از نزدیک تا دور انتشار می‌یابد. این تشابه را با دایره‌های متحدالمرکز که در تصویر ۳.۵ نشان داده شده است، به تصویر می‌کشیم.

۲. قدرت ترجیحات بالقوه کاربر در مجموعه امواج، با افزایش تعداد لایه k ام کاهش می‌یابد که مشابه دامنه کاهش تدریجی امواج واقعی است. رنگ آبی درحال محو شدن در تصویر ۳.۵ نشان‌دهنده کاهش ارتباط بین مرکز و موجودیت‌های مجاور است.

یکی از نگرانی‌های مربوط به مجموعه امواج این است که با افزایش تعداد لایه (k) ممکن است اندازه شان خیلی بزرگ شود. برای پرداختن به این نگرانی، توجه داشته باشید که:

۱. تعداد زیادی از موجودیت‌ها در گراف دانش واقعی «موجودیت‌های غرق شده» هستند، بدین معنی که فقط لینک‌های ورودی دارند اما هیچ

لینک خروجی‌ای ندارند مانند "۲۰۰۴" و "PG-۱۳" در تصویر ۳.۵.

۲. در سناریوهای پیشنهادی ویژه مانند پیشنهادات فیلم یا کتاب، روابط می‌توانند به دسته بندی‌های مرتبط با سناریو محدود شوند تا اندازه مجموعه امواج کاهش یابد و ارتباط بین موجودیت‌ها بهبود یابد. به عنوان مثال در تصویر ۳.۵ همه روابط مربوط به فیلم هستند و کلمه "فیلم" را در نام خود دارند.

۳. تعداد حداکثر لایه (H) معمولاً در عمل خیلی زیاد نیست، زیرا موجودیت‌هایی که از تاریخچه کاربر بسیار دور هستند، ممکن است اختلال بیشتری را نسبت به سیگنال‌های مثبت ایجاد کنند. در قسمت آزمایش‌ها درباره انتخاب H بحث خواهیم کرد.

۴. در RippleNet، می‌توانیم مجموعه‌ای با اندازه ثابت از همسایه‌ها را به جای استفاده از مجموعه موج کامل برای کاهش بیشتر سربار محاسبه، نمونه برداری کنیم. طراحی چنین نمونه‌گیری‌هایی جهت مهم برای کارهای آینده است، به خصوص نمونه برداری غیر یکنواخت برای به دست آوردن بهتر علایق بالقوه سلسله مراتبی کاربر.

۳.۲.۵ انتشار اولویت

روش‌های سنتی مبتنی بر فیلتر مشارکتی و انواع آن‌ها، بازنمایی‌های نهفته کاربران و آیتم‌ها را می‌آموزند، سپس رتبه بندی‌های نامعلوم را با اعمال مستقیم توابع خاص، از جمله ضرب داخلی، در بازنمایی‌شان پیش‌بینی می‌کنند. RippleNet برای مدل‌سازی تعاملات بین کاربران و آیتم‌ها با روشی دقیق‌تر، تکنیک انتشار اولویت را برای کشف علایق بالقوه کاربران در مجموعه امواج‌شان، بکار می‌گیرد.

همانطور که در تصویر ۲.۵ نشان داده شده است، هر آیتم v با تعبیه‌ی آیتم $v \in R^d$ مرتبط است که d بُعد تعبیه شده است. تعبیه آیتم می‌تواند شناسه‌ای با روش one-hot، خصوصیات، دسته‌ای از کلمات یا اطلاعات زمینه‌ای از یک آیتم را براساس سناریوی بکارگیری، یکی کند. با توجه به تعبیه آیتم v و مجموعه امواج لایه اول کاربر u ، به هر سه گانه (h_i, r_i, t_i) در S_u^1 ، یک احتمال ارتباط با مقایسه آیتم‌های v با رأس h_i و رابطه r_i در این سه گانه، اختصاص داده می‌شود:

$$P_i = \text{softmax}(v^T R_i h_i) = \frac{\exp(v^T R_i h_i)}{\sum_{(h_i, r_i, t_i) \in S_u^1} \exp(v^T R h)} \quad (۴.۵)$$

که به ترتیب $h_i \in R^d$ و $R_i \in R^{d \times d}$ تعبیه‌های مربوط به رابطه r_i و رأس h_i هستند. احتمال ارتباط p_i را می‌توان به عنوان شباهت آیتم v و موجودیت h_i در نظر گرفت که در فضای رابطه R_i اندازه‌گیری می‌شود. توجه داشته باشید که ضروریست به هنگام محاسبه ارتباط آیتم v و موجودیت h_i ، ماتریس تعبیه R_i را در نظر بگیریم، زیرا ممکن است جفت آیتم - موجودیت به هنگام بررسی شدن با روابط مختلف، دارای شباهت‌های مختلف باشند. برای مثال، "Forrest Gump" و "Cast Away" به هنگام در نظر گرفتن کارگردان‌ها یا ستاره‌هایشان بسیار شبیه بهم هستند، اما اگر با ژانر یا نویسنده سنجیده شوند، اشتراک کمتری دارند.

پس از به دست آوردن احتمالات ارتباط، مجموع موجودیت‌های انتهایی در S_u^1 را در نظر می‌گیریم که با احتمالات ارتباط اندازه‌گیری شده اند و بردار برگشت داده می‌شود:

$$o_u^1 = \sum_{(h_i, r_i, t_i) \in S_u^1} p_i t_i \quad (۵.۵)$$

که $t_i \in R^d$ تعبیه موجودیت انتهایی t_i است. بردار O_u^1 را می‌توان به‌عنوان پاسخ مرتبه ۱ تاریخچه کلیک کاربر V_u با توجه به آیت v در نظر گرفت. این شبیه به روش‌های فیلتر مشارکتی مبتنی بر آیت است که در آن به منظور کاهش اندازه پارامترها، کاربر به جای نشان داده شدن با یک بردار ویژگی مستقل، با آیت‌های مرتبط با خودش نشان داده می‌شود. از طریق عملیات‌ها در معادله‌های (۴-۵) و (۵-۵)، علایق کاربر از مجموعه سوابقش (V_u) به مجموعه موجودیت‌های مربوط به لایه اول خود ε_u^1 با لینک‌ها در S_u^1 منتقل می‌شوند که در RippleNet انتشار اولویت نامیده می‌شود. توجه داشته باشید که با جایگزین کردن v با O_u^1 در معادله (۴-۵)، می‌توانیم پروسه انتشار اولویت را برای به دست آوردن پاسخ مرتبه ۲ کاربر O_u^2 تکرار کنیم و این روش را می‌توان به صورت تکراری روی مجموعه امواج کاربر S_u^i به صورت $i = 1, 2, \dots, H$ انجام داد. بنابراین، اولویت کاربر تا H لایه دور از تاریخچه کلیک وی پخش می‌شود و پاسخ‌های متعدد کاربر u را برای مرتبه‌های مختلف مشاهده می‌کنیم: $O_u^1, O_u^2, \dots, O_u^H$. تعبیه کاربر u با توجه به آیت v با ترکیب پاسخ تمام مرتبه‌ها محاسبه می‌شود:

$$u = O_u^1 + O_u^2 + \dots + O_u^H \quad (۶.۵)$$

توجه داشته باشید که اگرچه پاسخ کاربر در آخرین لایه O_u^H حاوی کلیه اطلاعات مربوط به لایه‌های قبلی به صورت تئوری است، اما بازهم لازم است O_u^k برای k های کوچک را در محاسبه تعبیه کاربر گنجانید زیرا ممکن است در O_u^H رقیق شوند. سرانجام، تعبیه کاربر و تعبیه آیت برای محاسبه احتمال پیش بینی کلیک ترکیب می‌شوند:

$$\hat{y}_{uv} = \sigma(u^T v) \quad (۷.۵)$$

که در آن $\sigma(x)$ تابع سیگموئید است.

۴.۲.۵ الگوریتم یادگیری

در RippleNet، تصمیم بر آن است که بعد از مشاهده گراف دانش G و ماتریس بازخورد ضمنی Y ، احتمال زیر پارامترهای مدل به حداکثر رسانده شوند که شامل تعبیه همه موجودیت‌ها، روابط و آیت‌ها است. $p(\Theta)$ به عنوان توزیع گوسی^۳ با میانگین صفر و ماتریس کواریانس مورب^۴ تنظیم شده است.

در RippleNet، از یک روش فاکتورگیری تانسور سه طرفه^۵ برای تعریف تابع احتمال برای KGE استفاده شده است. همچنین توابع امتیازدهی جفت موجودیت-موجودیت در KGE و جفت آیت-موجودیت در انتشار اولویت می‌توانند در همان مدل محاسبه متحد شوند. تابع هزینه زیر برای RippleNet بدست آمده است:

$$\begin{aligned} \min L &= -\log(p(Y|\Theta, G) \cdot p(G|\Theta) \cdot p(\Theta)) \\ &= \sum_{(u,v) \in Y} -(y_{uv} \log \sigma(u^T v) + (1 - y_{uv}) \log(1 - \sigma(u^T v))) \\ &\quad + \frac{\lambda_2}{2} \sum_{r \in R} \|I_r - E^T R E\|_2^2 + \frac{\lambda_2}{2} (\|V\|_2^2 + \|E\|_2^2 + \sum_{r \in R} \|R\|_2^2) \end{aligned} \quad (۸.۵)$$

که V و E به ترتیب ماتریس تعبیه شده برای همه آیت‌ها و موجودیت‌ها هستند، l_r قسمتی از شاخص I در گراف دانش برای رابطه r است و R ماتریس تعبیه رابطه r است. در معادله (۸-۵)، در عبارت اول میزان اتلاف آنتروپی متقاطع بین شواهد تجربی تعاملات Y و مقدار پیش بینی

^۳Gaussian Distribution

^۴Diagonal Covariance Matrix

^۵Three-way Tensor Factorization

شده توسط RippleNet، اندازه گیری می شود، در عبارت دوم، توان دوم خطای بین شواهد تجربی گراف دانش l_r و ماتریس نشانگر بازسازی شده^۶ $E^T RE$ اندازه گیری می شود و عبارت سوم، تنظیم کننده ای برای جلوگیری از اتصالات بیش از حد تطبیق شده^۷ است. مسئله فوق بصورت مستقیم غیرقابل حل است، بنابراین از یک الگوریتم گرادیان نزولی تصادفی^۸ (SGD) استفاده می شود تا بصورت تکراری تابع هزینه را بهینه کنیم. پس از استراتژی نمونه گیری منفی در الگوریتم یادگیری RippleNet، برای محاسبات کارآمدتر در هر تعامل آموزشی، به طور تصادفی گروه کوچکی از تعاملات مثبت-منفی از Y و سه گانه واقعی-کاذب از G به عنوان نمونه قرار داده می شوند. سپس شیب تلفات L با توجه به پارامترهای مدل Θ محاسبه شده و کلیه پارامترها با انتشار مجدد^۹ بر اساس دسته کوچک نمونه به روز می شوند.

۵.۲.۵ بحث

شرح دادن

سیستم های توصیه گر قابل شرح قصد دارند علت اینکه چرا کاربر ممکن است آیتم خاصی را دوست داشته باشد، را روشن کنند که این امر به بهبود پذیرش یا رضایتشان از پیشنهادات و افزایش اعتماد در سیستم های توصیه گر کمک میکند. توضیحات معمولاً بر اساس برچسب های اجتماعی، شبکه های اجتماعی، جنبه و بیان احساسات است چرا که RippleNet علایق کاربران را بر اساس گراف دانش کاوش می کند که دیدگاه جدیدی از توجیه پذیری را با ردیابی مسیرها از سابقه کاربر به یک آیتم در گراف دانش با احتمال ارتباط بالا (معادله ۴-۵)، در اختیار قرار می دهد. به عنوان مثال، علاقه کاربر به فیلم “Back to the Future” ممکن است با مسیر، کاربر «تماشا کرد» “Forrest Gump” «به کارگردانی» “Robert Zemeckis” «کارگردانی می کند» “Back to the Future” توضیح داده شود در صورتی که آیتم “Back to the Future” احتمال ارتباط بالایی را به ترتیب با “Forrest Gump” و “Robert Zemeckis” در مجموعه موج لایه اول و لایه دوم داشته باشد. توجه داشته باشید که جدای از روش های مبتنی بر مسیر که الگوهای مسیر بصورت دستی طراحی می شوند، RippleNet بطور خودکار مسیرهای توضیح احتمالی را با توجه به احتمال ارتباط کشف می کند. در ادامه نمونه های تجربی در بخش آزمایش ارائه داده می شود تا بطور شهودی قابلیت توضیح RippleNet نشان داده شود.

برهم نهی موج^{۱۰}

پدیده ای رایج در RippleNet این است که مجموعه امواج یک کاربر ممکن است اندازه بزرگی داشته باشد که علایق بالقوه اش را به طور اجتناب ناپذیری در انتشار الویت کم می کند. با این حال، مشاهده می کنیم که موجودیت های مربوط به آیتم های مختلف در تاریخچه کلیک کاربر، اغلب همپوشانی بسیاری باهم دارند. به عبارت دیگر، یک موجودیت را می توان در گراف دانش با چندین مسیر که از روی تاریخچه کلیک کاربر شروع میشوند، به دست آورد. به عنوان مثال، “Saving Private Ryan” به کاربری متصل است که به ترتیب “The Terminal”، “Jurassic Park”، و “Braveheart” را از طریق بازیگری “Tom Hanks”، کارگردانی “Steven Spielberg” و ژانر “War” تماشا کرده است. این مسیرهای موازی، علایق کاربر را در موجودیت های همپوشانی شده افزایش می دهند. به حالت «برهم نهی موج» اشاره شده است، زیرا مشابه با پدیده تداخل در فیزیک است که در آن دو موج منجر به ایجاد موج منتج از دامنه بیشتر در نواحی خاص می شود. پدیده برهم نهی موج در گراف دانش دوم در تصویر ۲.۵ نشان داده شده است که رنگ قرمز تیره در اطراف دو موجودیت میانی پایین تر، حاکی از استحکام بیشتر علایق احتمالی کاربر است. در بخش آزمایشات درباره برهم نهی موج بحث خواهد شد.

^۶Reconstructed Indicator Matrix

^۷overfitting

^۸Stochastic Gradient Descent

^۹back-propagation

^{۱۰}Ripple Superposition

۶.۲.۵ پیوندها به کار موجود

در اینجا بحث را درمورد کارهای مرتبط ادامه داده و به کمک تکنیک های موجود در حوزه بزرگتری مقایسه شده اند.

مکانیسم توجه

مکانیسم توجه در ابتدا در دسته بندی تصویر^{۱۱} و ترجمه ماشینی^{۱۲} ارائه شده است که هدف آن یادگیری این است که در هنگام انجام وظیفه، به طور خودکار مرتبطترین قسمت از ورودی را پیدا کند. این ایده خیلی زود به سیستم های توصیه گر منتقل شد.

شبکه های حافظه دار^{۱۳}

شبکه های حافظه دار یک مدل توجه مکرر است که از یک ماژول حافظه خارجی برای پاسخگویی به سؤال و مدل سازی زبان استفاده می کند. عملیات خواندن تکراری بر روی حافظه خارجی، شبکه های حافظه را قادر می سازد تا وابستگی به مسافت طولانی را در متون استخراج کنند. محققان همچنین استفاده از شبکه های حافظه دار را در کارهای دیگری مانند طبقه بندی احساسات و پیشنهادات توصیه کرده اند. توجه داشته باشید که این کارها معمولاً بر حافظه های سطح ورودی یا سطح جمله متمرکز می شوند درحالی که کار ما به اتصالات سطح موجودیت در گراف دانش می پردازد، که هنگام انجام تکرارهای چندلایه ای، دقیق تر و ملموس تر است. علاوه بر این، کار ما یک رابطه KGE را به عنوان تنظیم کننده ای برای یادگیری با ثبات تر و مؤثرتر در برمی گیرد.

تعبیه گراف دانش

محققان پیشنهاد می کنند اطلاعات کمکی مانند انواع موجودات ها، قوانین منطق و توضیحات متنی برای کمک به KGE درج شوند. با این حال، این روش ها با توجه به اهداف یادگیری خود برای برنامه هایی که درون آن ها نمودار است مانند پیش بینی لینک یا طبقه بندی سه گانه، مناسب تر هستند. از این دیدگاه، RippleNet را می توان به عنوان یک روش KGE طراحی شده خاص در نظر گرفت که به طور مستقیم به ارائه توصیه می پردازد.

۳.۵ آزمایش ها

در این بخش، RippleNet در سه سناریو در دنیای واقعی ارزیابی شده است: پیشنهادات فیلم، کتاب و اخبار. نخست مجموعه داده ها، مدل های پایه و تنظیمات آزمایش ها معرفی شده و سپس نتایج آزمایش ارائه داده می شود.

۱.۳.۵ مجموعه داده

از سه مجموعه داده MovieLens-۱M Book-Crossing، و Bing-News در آزمایش ها به ترتیب برای توصیه های فیلم، کتاب و اخبار استفاده شده است. از آنجا که MovieLens-۱M و Book-Crossing داده های بازخورد صریح هستند، آن ها به بازخورد ضمنی تبدیل شده اند. از شناسه تعبیه شده کاربران و آیتم ها به عنوان ورودی خام برای MovieLens-۱M و Book-Crossing استفاده شده است در حالی که برای Bing-News، شناسه تعبیه شده قطعه ای از اخبار و متوسط کلمه تعبیه شده از عنوان آن به عنوان ورودی خام برای آیتم در نظر گرفته شده است چراکه عناوین خبری معمولاً بسیار طولانی تر از اسامی فیلم ها یا کتاب ها هستند، از این رو اطلاعات مفیدی را برای پیشنهادات ارائه می دهند.

¹¹image Classification

¹²Machine Translation

¹³Memory Networks

۲.۳.۵ تنظیمات آزمایش‌ها

در RippleNet تعداد لایه‌ها را برای MovieLens-۱M و Book-Crossing برابر $H = 2$ و برای Bing-News برابر با $H = 3$ قرار می‌دهیم.

روش RippleNet در دو سناریو آزمایشی مورد ارزیابی قرار گرفته است:

۱. در پیش بینی نرخ کلیک (CTR)، مدل آموزش دیده‌ای را برای هر قطعه از برهم کنش‌های موجود در مجموعه آزمون اعمال کرده و احتمال کلیک پیش بینی شده به دست می‌آید. ما از صحت و AUC برای ارزیابی عملکرد پیش بینی CTR استفاده می‌کنیم.
۲. در K توصیه برتر از مدل آموزش دیده‌ای استفاده شده است تا آیت‌های K را با بالاترین احتمال کلیک پیش بینی شده برای هر کاربر در مجموعه آزمون انتخاب کرده و برای ارزیابی مجموعه‌های توصیه شده Precision، Recall و F_1 را انتخاب کرده است.

۳.۳.۵ نتایج

نتایج کلیه روش‌ها در پیش بینی CTR و K توصیه برتر به ترتیب در تصویر ۴.۵، ۵.۵، ۶.۵ و ۷.۵ ارائه شده است. چندین مشاهده برجسته است:

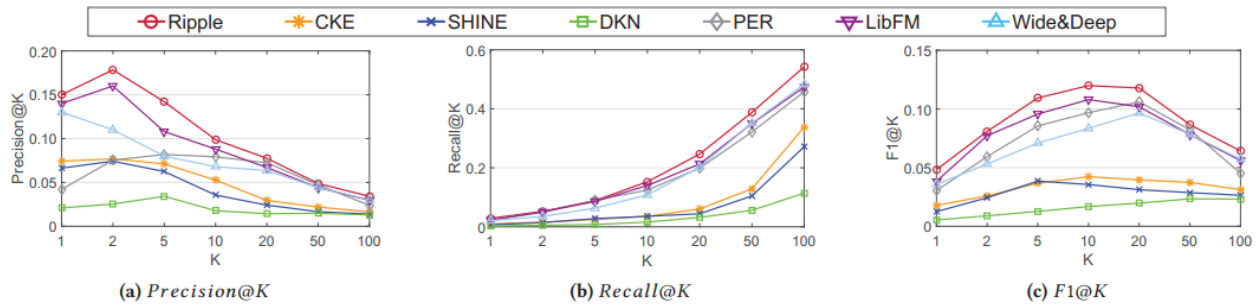
Model	MovieLens-1M		Book-Crossing		Bing-News	
	AUC	ACC	AUC	ACC	AUC	ACC
RippleNet*	0.921	0.844	0.729	0.662	0.678	0.632
CKE	0.796	0.739	0.674	0.635	0.560	0.517
SHINE	0.778	0.732	0.668	0.631	0.554	0.537
DKN	0.655	0.589	0.621	0.598	0.661	0.604
PER	0.712	0.667	0.623	0.588	-	-
LibFM	0.892	0.812	0.685	0.639	0.644	0.588
Wide&Deep	0.903	0.822	0.711	0.623	0.654	0.595

* Statistically significant improvements by unpaired two-sample t -test with $p = 0.1$.

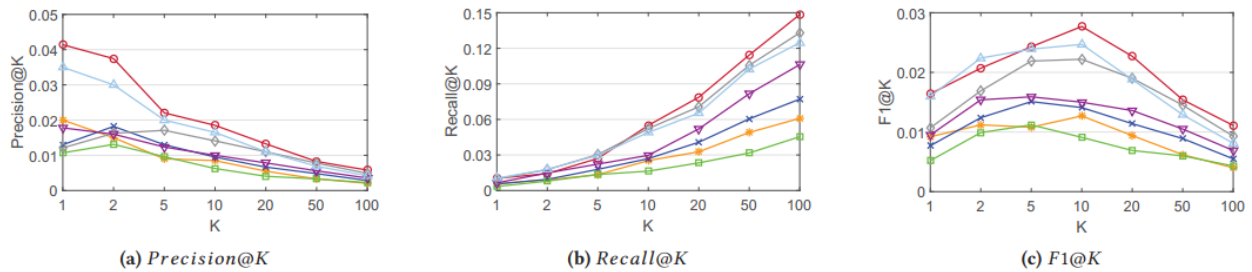
تصویر ۴.۵: نتایج AUC و Accuracy در پیش بینی CTR

- CKE عملکرد نسبتاً ضعیفی را نسبت به سایر مدل‌های پایه دارد که احتمالاً به این دلیل است که بدون ورودی بصری و متنی فقط دانش ساختاری را در دسترس داریم.
- SHINE در پیشنهادات فیلم و کتاب عملکرد بهتری را در مقایسه با اخبار دارد. این امر به این دلیل است که وقتی سه گانه‌های لایه اول به عنوان ورودی پروفایل در نظر گرفته می‌شوند، برای اخبار بسیار پیچیده هستند.

- DKN در مقایسه با سایر مدل‌های پایه بهترین عملکرد را در توصیه‌های خبری دارد اما در توصیه فیلم و کتاب بدترین عملکرد را دارد. دلیل این است که اسامی فیلم و کتاب برای ارائه اطلاعات مفید، بسیار کوتاه و مبهم هستند.
- PER در پیشنهادات فیلم و کتاب عملکرد رضایت‌بخشی ندارد زیرا مسیرهای ارجح تعریف شده توسط کاربر به سختی می‌تواند مطلوب باشد. علاوه بر این، نمی‌توان از آن در پیشنهاد اخبار استفاده کرد زیرا انواع موجودیت‌ها و روابط درگیر در اخبار برای پیش تعریف مسیرهای ارجح بسیار پیچیده هستند.
- از آنجاییکه دو ابزار توصیه عمومی، LibFM و Deep Wide عملکرد رضایت‌بخشی دارند که نشان می‌دهد که آن دو می‌توانند به خوبی از دانش KG در الگوریتم‌هایشان استفاده کنند.
- RippleNet در بین همه روش‌ها در سه مجموعه داده بهترین عملکرد را دارد. به خصوص، RippleNet نسبت به مدل‌های پایه عملکرد بهتری روی AUC در پیشنهادات فیلم، کتاب و اخبار دارد.



تصویر ۵.۵: نتایج ارزیابی‌ها برای k توصیه برتر روی مجموعه داده MovieLens-1M

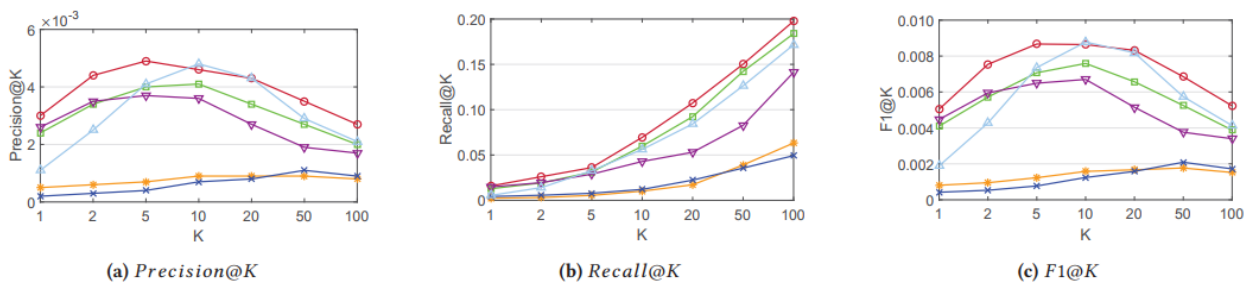


تصویر ۶.۵: نتایج ارزیابی‌ها برای k توصیه برتر روی مجموعه داده Book-Crossing

RippleNet در k توصیه برتر نیز عملکرد برجسته‌ای را دارد همانطور که در تصویر ۵.۵، ۶.۵ و ۷.۵ نشان داده شده است. توجه داشته باشید که عملکرد k توصیه برتر برای Bing-News بسیار پایین است زیرا تعداد اخبار به طور قابل توجهی از تعداد فیلم و کتاب بیشتر است. ۲ نتیجه مهم بدست آمده عبارتند از:

۱. اندازه مجموعه موج در هر لایه: با افزایش اندازه مجموعه موج، عملکرد RippleNet در ابتدا بهبود می‌یابد زیرا یک مجموعه موج بزرگتر می‌تواند دانش بیشتری را از گراف دانش رمزگذاری کند. اما توجه داشته باشید که وقتی اندازه خیلی بزرگ است، عملکرد کاهش می‌یابد. به طور کلی، مطابق با نتایج آزمایش اندازه ۱۶ یا ۳۲ برای اکثر مجموعه داده‌ها کافی است.

۲. شماره لایه (H): بهترین عملکرد زمانی حاصل می‌شود که H برابر با ۲ یا ۳ باشد زیرا اگر H خیلی کم باشد نمی‌تواند ارتباطات مفید از فاصله‌های دور را کشف کند و اگر زیاد باشد ممکن است نویزهای بیشتری را نسبت به سیگنال‌های مثبت دریافت کند.



تصویر ۷.۵: نتایج ارزیابی‌ها برای k توصیه برتر روی مجموعه داده Bing-News

۱.۶ مقدمه

قراردادن گراف دانش در سیستم‌های توصیه‌گر، به طور امیدوارکننده‌ای دقت پیشنهاد و توضیح آن را بهبود داده است. با این حال، روش‌های موجود تا حد زیادی بر این باورند که گراف دانش کامل است و "دانش" را به سادگی در گراف دانش در مرحله سطحی از داده‌های خام یا تعبیه‌های موجودیت انتقال می‌دهند. این امر احتمالاً به عملکرد نامطلوبی منجر می‌شود، زیرا یک گراف دانش عملیاتی به ندرت می‌تواند کامل باشد و یک گراف دانش معمولاً حقایق، روابط و موجودیت‌هایی را از دست داده است. بنابراین، در این باره بحث می‌شود که مدنظر قراردادن ماهیت ناکامل گراف دانش هنگام وارد کردن آن در سیستم توصیه‌گر، حائز اهمیت است.

در تمایز با روش‌های توصیه مبتنی بر گراف دانش پیشین، اطلاعات رابطه‌ای گراف دانش را منتقل کرده است، تا دلایلی را درک کند که چرا یک کاربر یک آیتم را دوست دارد. به عنوان نمونه، اگر کاربر چندین فیلم به کارگردانی (رابطه) یک شخص (موجودیت) تماشا کرده باشد، می‌توانیم استنباط کنیم که رابطه کارگردان هنگام تصمیم‌گیری کاربر نقش اساسی را ایفا می‌کند، بنابراین به درک اولویت کاربر در یک دسته بندی دقیق‌تر کمک می‌کند.

در این مقاله، پیشنهاد شده است تا مدل ارجحیت کاربر مبتنی بر ترجمه را با تکمیل گراف دانش در یک مدل مشترک برای پیشرفت‌های متقابل، متحد سازیم. [۵]

ایده اصلی به دو صورت است:

۱. استفاده از حقایق در گراف دانش به عنوان داده‌های کمکی برای تقویت مدل تعامل کاربر با آیتم.
۲. تکمیل حقایق مفقودشده در گراف دانش بر اساس مدل سازی پیشرفته کاربر-آیتم. به عنوان مثال، می‌توانیم اولویت کاربر درباره کارگردان را از طریق موجودیت‌ها و روابط مرتبط درک کنیم.

در این مقاله یک مدل اولویت کاربر مبتنی بر ترجمه^۱ (TUP) پیشنهاد شده است تا با یادگیری گراف دانش به صورت یکپارچه ادغام شود. ایده اصلی این است که بین کاربران و آیتم‌ها چندین رابطه ضمنی وجود دارد که نشان دهنده دلایل استفاده کاربران از آیتم‌ها است. در حالیکه می‌توانیم تعداد اولویت‌ها را از قبل تعریف کنیم و TUP را از داده‌های تعامل کاربر-آیتم آموزش دهیم، اولویت‌ها به صورت بردارهای پنهان شرح داده می‌شوند. برای اینکه اولویت‌ها را با معانی ضمنی بهم مرتبط کنیم، آن‌ها را با روابط موجود در گراف دانش تطبیق می‌دهیم. در واقع به صورت تخصصی، تعبیه‌های روابط و نیز تعبیه‌های موجودیت از گراف دانش به TUP منتقل شده و همزمان وظایف پیشنهاد و تکمیل گراف دانش آموزش داده می‌شود. پژوهش فعلی در صدد انجام سه وظیفه اصلی بوده است که عبارتند از:

¹Translation-based User Preference model

۱. پیشنهاد یک مدل مبتنی بر ترجمه که از بازنمایی‌های اولویت‌های ضمنی، برای فهمیدن روابط بین کاربران و آیتم‌ها استفاده می‌کند.
۲. بر اهمیت توصیه آیتم و تکمیل گراف دانش به طور مشترک، برای جفت کردن بازنمودهای اولویت با روابط آگاه به دانش تاکید شده است، در نتیجه مدل را با قابلیت توضیح می‌توان قدرتمند کرد.
۳. آزمایش‌هایی گسترده در دو مجموعه داده برای N توصیه برتر و وظایف تکمیل گراف دانش، انجام گرفته تا منطق یادگیری مشترک را تأیید کند. نتایج تجربی، اثربخشی و توجه‌پذیری مدل را نشان داده‌اند.

۲.۶ کارهای انجام شده

۱.۲.۶ توصیه آیتم

محققان در مرحله اولیه توصیه آیتم بر توصیه‌سازی کاربران یا آیتم‌های مشابه به یک کاربر هدف با بکارگیری صرف تاریخچه‌ی تعاملات مانند فیلترهای مشارکتی، دستگاه‌های فاکتورسازی، تکنیک‌های فاکتورسازی ماتریس، BPRMF و توسعه آن‌ها تمرکز دارند. چالش اصلی در استخراج ویژگی‌های کاربران و آیتم‌ها جهت محاسبه شباهت بین آن‌ها و به اصطلاح روش‌های مبتنی بر شباهت قرار دارد. با وجود تعدد مدل‌های شبکه عصبی، حجم وسیعی از روش‌ها، روش‌های مبتنی بر شباهت را با شبکه عصبی گسترش می‌دهند و مکانیزم مؤثرتری را در استخراج خودکار ویژگی‌های نهفته کاربران و آیتم‌ها جهت توصیه‌سازی ارائه می‌دهند. با این حال، هنوز هم از موضوع کمبود داده‌ها و مسئله شروع سرد رنج می‌برند. روش‌های مبتنی بر محتوا با معرفی اطلاعات جانبی مختلف همچون مرورهای متنی، داده‌های رابطه‌ای و گراف‌های دانش با این موضوعات مقابله می‌کنند.

در میان اطلاعات جانبی، گراف‌های دانش (به عنوان مثال DBPedia) به دلیل وجود ساختارهای کاملاً تعریف شده و منابع کافی، پتانسیل‌های زیادی را در توصیه نشان می‌دهند. این نوع از روش‌ها اساساً دانش ساختاری موجودیت‌ها را از گراف دانش به مدل‌سازی تعامل کاربر-آیتم برحسب نگاشت بین موجودیت‌ها و آیتم‌ها انتقال می‌دهند. آن‌ها به دو گروه طبقه بندی می‌شوند: روش‌های تجمع جفت داده‌های کاربر-آیتم با سه‌گانه‌های گراف دانش و روش‌های ترکیب تعبیه‌های آیتم و موجودیت آموزش یافته از منابع مختلف.

توصیه مبتنی بر ترجمه الهام گرفته از یادگیری بازنمایی^۲ گراف دانش است و بر این فرض استوار است که انتخاب آیتم‌ها، روابط ترجمه در فضای بردار پنهان را جبران می‌کند که در آن روابط هم به کاربران در توصیه‌های متوالی مرتبط هستند و نیز از طریق دقت مبتنی بر حافظه مدل‌سازی ضمنی شده‌اند. بنابراین این نوع روش را با در نظر گرفتن مسئله N به N در مدل‌سازی ارجحیات کاربر به عنوان روابط ترجمه، بهبود می‌بخشیم که باید بعداً به واسطه انتقال دانش موجودیت‌ها و روابطشان از گراف دانش ارتقا یابد.

۲.۲.۶ تکمیل گراف دانش

دانش خارجی در بسیاری از کارهای پردازش زبان طبیعی از جمله پاسخگویی به سؤال مؤثر است که باعث افزایش محبوبیت گراف‌های دانش شده است. اگرچه روش‌های بسیاری برای یافتن موجودیت‌ها و روابط آن‌ها از متون وجود دارد، گراف‌های دانش موجود خیلی از کامل بودن دور هستند. مطالعات اخیر برای تکمیل گراف دانش اشتیاق زیادی را برای یادگیری بازنمایی‌های با ابعاد کم موجودیت‌ها و روابط نشان می‌دهد در حالی که دانش ساختاری نمودار را حفظ می‌کند. روش‌های یادگیری بازنمایی را به دو گروه طبقه بندی می‌کنیم: مدل‌های مسافت ترجمانی و مدل‌های تطابق معنایی.

²representation learning

۳.۲.۶ رابطه بین دو کار

آیتم‌ها معمولاً در بسیاری از حوزه‌ها با موجودیت‌ها مرتبط هستند از قبیل کتاب‌ها، فیلم‌ها و موسیقی که امکان انتقال دانش بین این حوزه‌ها را میسر می‌سازد. این اطلاعات شامل دو وظیفه هستند که مکمل یکدیگرند و ارتباط بین آیتم‌ها یا کاربران و آیتم‌ها را نشان می‌دهند. با توجه به مدل‌ها، هر دو وظیفه قصد رتبه بندی کاندیداهای موجود در یک پرس‌وجو (مثلاً یک موجودیت یا کاربر) و نیز ارتباط عینی یا ضمنی آن‌ها را دارند.

۳.۶ مدل

قبل از معرفی روش‌های پیشنهادی، نخست بیابید به صورت رسمی دو وظیفه و نیز TransH را به عنوان مولفه‌ای برای تکمیل گراف دانش در مدل خود تعریف کنیم.

توصیه آیتم:

با توجه به لیستی از تعاملات آیتم - کاربر $y = (u, i)$ ، از بازخورد ضمنی به عنوان پروتکلی استفاده شده است بنابراین هر زوج (u, i) نشان می‌دهد که کاربر $u \in U$ ، آیتم $i \in I$ را مصرف می‌کند.

تکمیل گراف دانش:

گراف دانش یک گراف مستقیم متشکل از سه جزء موضوع-ویژگی-شی است. هر سه جز یا بخش بیانگر این است که رابطه‌ای از موجودیت رأس تا موجودیت انتهایی وجود دارد که رسماً توسط (e_h, e_t, r) تعریف می‌شود که این مجموعه $e_h, e_t \in E$ ، موجودیت‌ها هستند و $r \in R$ رابطه‌ها هستند.

TUP مدلی را برای توصیه آیتم خاطرنشان می‌سازد. این مدل لیستی از زوج‌های کاربر-آیتم y را به عنوان ورودی می‌گیرد و خروجی امتیاز مربوطه $g(u, i; p)$ است که نشان‌دهنده این احتمال هست که کاربر u آیتم i را دوست دارد، با توجه به اولویت $p \in P$ که تعداد مجموعه اولویت p از قبل تعریف شده است. برای هر جفت کاربر-آیتم، اولیوی را در نظر می‌گیریم که به عنوان نقش مشابه متناسب با دو موجودیت عمل می‌کند. برای سروکار داشتن با مسئله روابط $N \times N$ ، ابرصفحه‌هایی معرفی شده‌اند و هر اولویت را با دو بردار w برای انتقال به یک ابرصفحه تعیین می‌کنیم که p برای انتقال بین کاربران و آیتم‌ها است.

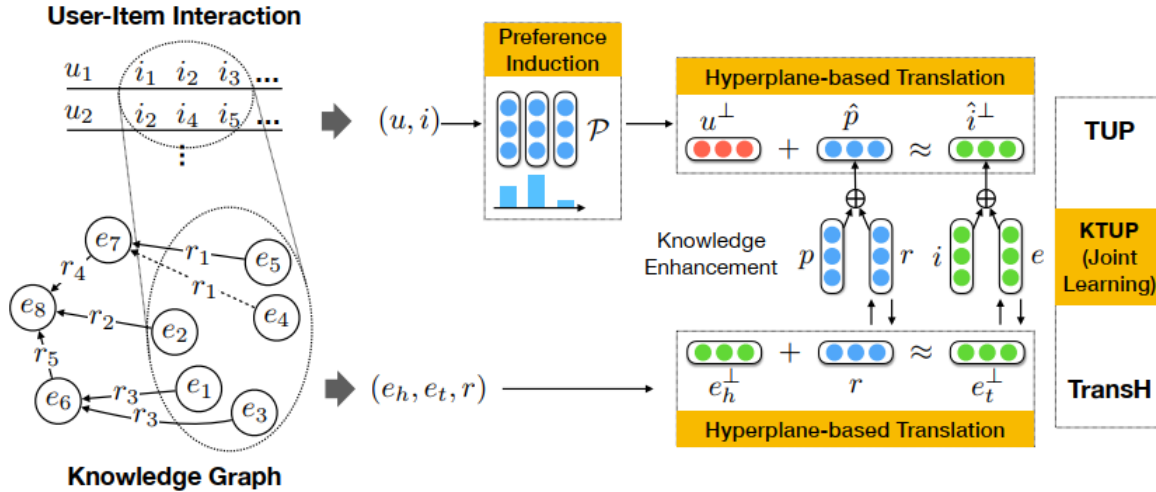
KTUP یک معماری چند وظیفه‌ای است. با توجه به گراف دانش و Y و مجموعه‌ای از ترازبندی‌های آیتم-موجودیت $A = \{(i, e) | i \in \text{موجودیت}\}$ که در آن هر زوج (i, e) به معنای آن است که در گراف دانش داده شده i می‌تواند به موجودیت e نگاشت شود.

۴.۶ TUP برای توصیه آیتم

با الهام از فرضیه ترجمه فوق بین دو موجودیت در گراف دانش، TUP را برای مدل‌های عینی ارجحیات کاربر پیشنهاد می‌دهیم و آن‌ها را به عنوان روابط ترجمه بین کاربران و آیتم‌ها در نظر می‌گیریم. با توجه به مجموعه‌ای از تعاملات کاربر-آیتم Y ، اولیوی برای هر زوج کاربر-آیتم به طور خودکار لحاظ می‌شود و تعبیه‌های اولویت p ، کاربر u و آیتم i یاد گرفته می‌شود که $i \approx u + p$ می‌شود.

۱.۴.۶ استقراء اولویت

این مؤلفه بناست که با توجه به جفت کاربر-آیتم (u, i) اولیوی را از مجموعه فاکتورهای نهفته P استقراء کند. این فاکتورها توسط همه کاربران به اشتراک گذاشته می‌شود و هر $p \in P$ یک اولویت متفاوت را نشان می‌دهد که هدف آن اخذ مشترکات در میان کاربران بصورت ویژگی‌های سراسری



تصویر ۱۰۶: چهارچوب KTUP

مکمل با تعبیه‌های کاربر است که بر یک کاربر واحد محلی تمرکز دارد.

دو استراتژی برای استقراء اولویت طراحی می‌کنیم: رویکردی سخت که یکی از اولویت‌های P را انتخاب می‌کند و رویکردی نرم که تمام اولویت‌ها را با دقت و توجه ترکیب می‌کند.

منطق موجود در پشت استراتژی سخت این است که هر اولویت یکتا، زمانی اثر می‌پذیرد که کاربر درباره آیتم تصمیم می‌گیرد.

برای نمونه برداری صحیح اولویت در زوج کاربر-آیتم از Straight-Through (ST) Gumbel SoftMax بهره می‌گیریم که از ترفند پارامترسازی مجدد برای انتشار بازگشتی استفاده می‌کند.

۵.۶ یادگیری مشترک از طریق KTUP برای دو وظیفه

KTUP مدل توصیه مبتنی بر ترجمه TUP را با قراردادن گراف دانش موجودیت‌ها و روابط در آن، گسترش می‌دهد. دانش کمکی بصورت ملموس‌تری اتصال بین آیتم‌ها را به عنوان محدودیت‌هایی برای زوج‌های مدل کاربر-آیتم تکمیل می‌کند. تصویر ۱۰۶ چهارچوب کلی KTUP را نشان می‌دهد. در سمت چپ، ورودی‌ها که شامل تعاملات کاربر - آیتم، گراف دانش و ترازبندی بین آیتم‌ها و موجودیت‌هاست، وجود دارند. در گوشه بالایی سمت راست، TUP برای توصیه آیتم قرار دارد، در حالی که TransH برای تکمیل گراف دانش در گوشه پایینی سمت راست قرار دارد. در مورد روابط و موجودیت‌ها، تعبیه آیتم پیشرفته شده حاوی دانش رابطه‌ای است که قابل اجرا در تعاملات کاربر-آیتم است و توصیه آیتم را بهبود می‌بخشد.

۱۰.۵.۶ آموزش

KTUP با استفاده از تابع هدف عمومی^۳ به شرح زیر آموزش داده می‌شود:

$$L = \lambda L_p + (1 - \lambda)L_k \quad (۱.۶)$$

^۳overall objective function

که λ یک ابرپارامتر برای تعادل بین دو وظیفه است.

۶.۶ نتایج آزمایش‌ها

در این بخش، آزمایش‌های کمی بر روی وظایف جداگانه‌ای از توصیه آیتم و تکمیل گراف دانش انجام شده است. برای هر وظیفه، از دو مجموعه داده در دامنه‌های مختلف استفاده شده است و ارزیابی‌های بیشتری در مورد تاثیر پراکندگی داده‌ها و نیز مساله $N \times N$ انجام گرفته است. به تبع از CoFM، از دو مجموعه داده در دسترس عموم در حوزه فیلم و کتاب استفاده شده است: MovieLens-1m و DBbook2014. هر دو این مجموعه داده‌ها شامل کاربران و امتیازدهی‌های آن‌ها در فیلم‌ها یا کتاب‌ها هستند که بعداً از طریق نگاشت آیتم‌ها به موجودیت‌های خارجی، برای LODRecSys پالایش می‌شوند. برای جمع‌آوری حقایق مرتبط از DBPedia، فقط آن سه گانه‌هایی را در نظر می‌گیریم که به طور مستقیم به موجودیت‌های نگاشت شده با آیتم مرتبطند، نقش آن‌ها (یعنی عینی بودن یا ذهنی بودن) برای موجودیت اهمیت ندارد. پس از پیش پردازش، ۶۰۴۰ کاربر و ۳۳۳۰ آیتم با ۹۹۸۵۳۹ رتبه بندی در MovieLens-1m وجود دارد. مسئله پراکندگی داده‌ها در DBbook2014 شدیدتر است.

۱.۶.۶ ارزیابی و تکمیل گراف دانش

در این بخش وظیفه تکمیل گراف دانش را مورد ارزیابی قرار داده می‌شود. همچنین قرار است موجودیت‌های از دست رفته و ناموجود e_n ، e_t ، r برای جزء سه تایی (e_n, e_t, r) را پیش بینی شود. برای هر موجودیت از دست رفته، تمام موجودیت‌ها به عنوان کاندید در نظر گرفته می‌شوند و آن‌ها با توجه به امتیازات محاسبه شده بر اساس تعبیه‌های موجودیت و رابطه، رتبه بندی می‌شوند. جدول ۲.۶ عملکرد کلی را نشان می‌دهد. می‌توان دید که KTUP تقریباً از سایر مدل‌های موجود در هر دو مجموعه داده به جز ارزش رتبه میانگین در MovieLens-1m عملکرد بهتری دارد. استدلال می‌کنیم این معیار از اهمیت کمتری برخوردار است زیرا به راحتی به وسیله سه‌گانه‌ای سرسخت با رتبه پایین کاهش می‌یابد. در مقایسه با TransH بهبود

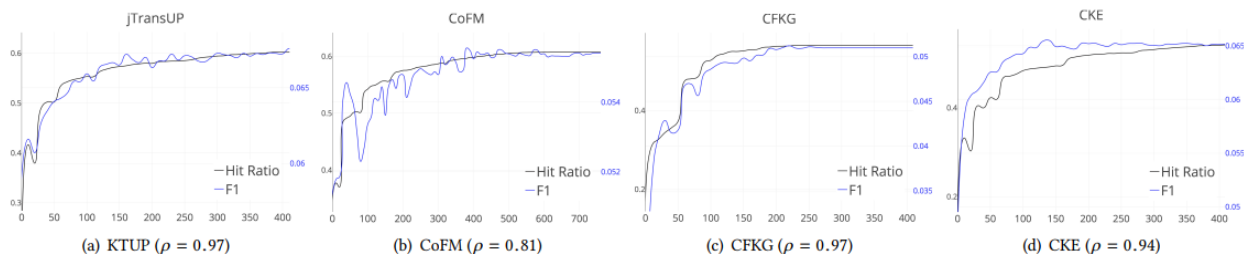
	MovieLens-1m		DBbook2014	
	Hit@10 (%)	Mean Rank	Hit@10 (%)	Mean Rank
TransE	46.95	537	60.71	531
TransH	47.63	537	60.06	556
TransR	38.93	609	56.33	563
CFKG	41.56	523	58.83	547
CKE	34.37	585	54.66	593
CoFM (share)	46.62	515	57.01	529
CoFM (reg)	46.51	506	60.81	521
KTUP (hard)	48.39	525	60.53	501
KTUP (soft)	48.90	527	60.75	499

تصویر ۲.۶: ارزیابی کلی تکمیل گراف دانش

بیشتری در نسبت ضربه برای MovieLens-1m به دست می‌آورد زمانی که همین برای DBbook2014 مقایسه می‌شود، زیرا MovieLens-1m دارای روابط بیشتری بین کاربران و آیتم‌هایی است که برای مدل‌سازی دانش ساختاری بین موجودیت‌ها مفید است.

اگرچه ارزیابی بر روی وظایف جداگانه‌ای انجام شده است، هنوز مشخص نیست که طرح‌های انتقالی مختلف چطور و چگونه اثربخشی دارند. بنابراین همبستگی‌های آموزشی بین منحنی‌های آموزش دو وظیفه را بررسی می‌کنیم. هر همبستگی قوی اشاره به انتقال یادگیری کامل‌تر و استفاده بهتر از اطلاعات تکمیلی از یکدیگر دارد. از آنجا که تکمیل گراف دانش هیچ مقادیری از F_1 ندارد، بنابراین نسبت ضربه آن در رابطه با محور چپ y ارائه می‌شود و پیشنهاد آیتم از طریق F_1 در محور y راست نشان داده می‌شود.

همانطور که در شکل ۲.۶ نشان داده شده است، می‌بینیم که KTUP و CFKG قوی‌ترین همبستگی بین منحنی‌های خود را نشان می‌دهند بدین



تصویر ۳.۶: همبستگی منحنی آموزش

معنی که افزایش و کاهش منحنی باید همزمان بر روی منحنی دیگر نیز منعکس گردد. این بدان معنی است که انتقال روابط نقش مهمی را در آموزش دو وظیفه با یکدیگر ایفا می‌کند. با این حال، CFKG عملکرد خوبی را در هر دو وظیفه ندارد (در جدول ۲.۶ نشان داده شده است). مورد مطالعه مقاله، نمونه‌ای از \mathcal{M} -Movielens است که مبین تصویری بصری از قابلیت توضیح پذیری پژوهش فعلی است. در سمت چپ کاربری قرار دارد که با ۷ فیلم تعامل داشته است. KTUP ابتدا اولویت‌های کاربر را به این فیلم‌ها القا می‌کند و درمی‌یابد که آنچه مورد نظر کاربر است، روابط کارگردان و بازیگر نقش اصلی است. بنابراین، نزدیک‌ترین آیتم‌ها بر اساس اولویت‌های حاصل از آن جستجو خواهد شد. چهار فیلم پیشنهادی را در سمت راست ارائه می‌دهیم که عبارتند از: بتمن برای همیشه، بتمن و رابین.

۷.۶ نتیجه

در این مقاله، مدل جدید توصیه‌گر مبتنی بر ترجمه TUP پیشنهاد شده است و آن را برای ادغام با تکامل گراف دانش که اصطلاحاً KTUP نامگذاری شده است، تعمیم دادیم. TUP قادر به مدل‌سازی روابط ضمنی مختلف بین کاربران و آیتم‌هایی است که ترجیحات کاربران را هنگام استفاده از آیتم‌ها آشکار می‌سازد.

KTUP قابلیت توجه‌پذیری مدل را از طریق روابط همسو و اولویت‌ها و عملکرد هر دو وظیفه را از طریق یادگیری مشترک بهبود می‌بخشد.

KGAT

۱.۷ مقدمه

با وجود اینکه فیلتر مشارکتی همه‌گیر است و کارایی دارد؛ اما توانایی لازم برای استفاده از اطلاعات اضافی کاربران و تعاملات بین کاربر و آیتم را ندارد و همچنین در شرایطی که تعامل بین کاربر و آیتم بسیار کم است بطور ضعیف عمل می‌کند. یک راه متداول برای جمع‌آوری و استفاده از این اطلاعات انتقال آنها به بردار ویژگی‌های عمومی^۱ است که در واقع تمامی صفات مربوط به یک موجودیت را در یک بردار نمایش می‌دهد و سپس آن را به یک مدل یادگیری با نظارت^۲ برای^۳ پیش‌بینی امتیاز لازم، می‌دهد. با وجود کارایی مناسبی که این روش دارد؛ اما این روش هر تعامل را به صورت یک نمونه داده‌ی مستقل مدلسازی می‌کند و روابط بین آن‌ها مورد توجه قرار نمی‌گیرد؛ این موضوع آن‌ها را برای بیرون کشیدن سیگنال مشارکتی مبتنی بر صفات از رفتار مشارکتی کاربران، ناکارآمد می‌کند.

اگر گراف ۱.۷ را مورد توجه قرار دهیم متوجه می‌شویم که فیلتر مشارکتی صرفاً براساس تاریخچه‌ی مشترک کاربران از استفاده از آیتم‌ها پیشنهاددهی انجام می‌دهد، اما در یادگیری با نظارت ما می‌توانیم بر اساس اطلاعاتی که در مورد صفات مشابه آیتم‌ها داریم نیز پیشنهاددهی داشته باشیم. بدیهی است این دو نوع اطلاعات برای توصیه، نه تنها مکمل هستند، بلکه بین کاربر هدف و آیتم رابطه عالی ایجاد می‌کنند.

برای پرداختن به محدودیت مدل‌های یادگیری با نظارت مبتنی بر ویژگی^۴، یک راه حل این است که برای ساختن مدل پیش‌بینی، گراف اطلاعات جانبی آیتم را در نظر بگیریم. که آن را گراف دانش می‌نامیم.

ما ساختار ترکیبی از گراف دانش و گراف کاربر-آیتم را به عنوان (CKG) تعریف می‌کنیم. همانطور که در گراف دانش مشارکتی شکل ۱ نشان داده شده است، کلید توصیه موفقیت‌آمیز بهره برداری کامل از روابط مرتبه بالا در CKG است. با این وجود، برای بهره‌برداری از چنین اطلاعات مرتبه‌ای، چالش‌های غیرقابل اغماضی وجود دارد:

- گره‌های مرتبط با کاربر هدف به طور چشمگیری با اندازه‌ی مرتبه گراف، افزایش پیدا می‌کنند که این اضافه بار محاسباتی چشمگیری به مدل اضافه می‌کند و در واقع گره‌هایی که لازم است بررسی شوند، دیگر مختصر نیستند.
- روابط مرتبه بالا، به طور نامساوی در پیش‌بینی موثر هستند که در اینجا لازم می‌شود که به طور دقیق به روابط در مدل وزن‌دهی کنیم یا آن‌ها را انتخاب کنیم.

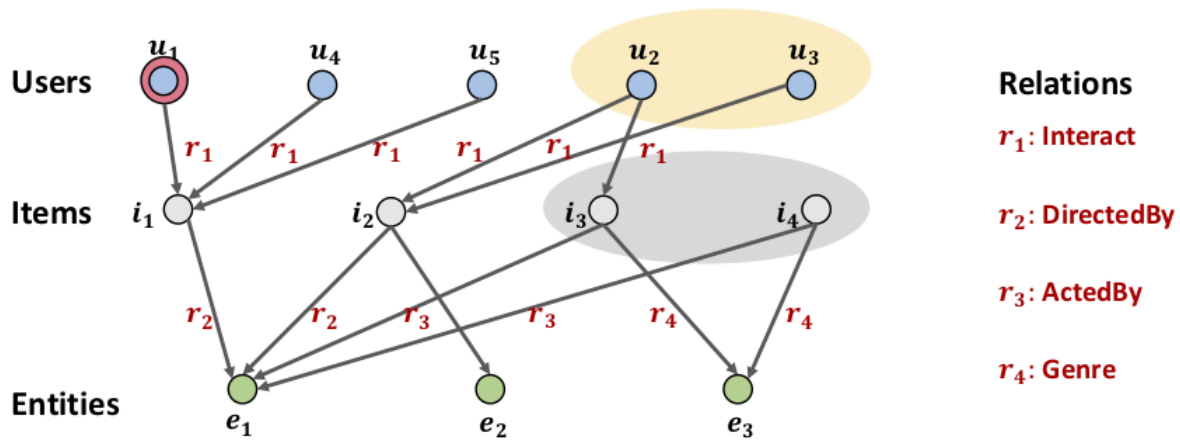
به‌طور خاص یک مدل جدید به نام شبکه‌ی توجه گراف دانش^۴ پیشنهاد می‌شود که دو طراحی برای مواجهه‌ی هماهنگ با چالش‌های مدل‌سازی رابطه مرتبه بالا دارد:

¹Generic Feature Vector

²supervised learning

³feature-based

⁴Knowledge Graph Attention Network



تصویر ۱۰۷: نمایی از گراف دانش مشارکتی؛ u_1 کاربر مورد نظر برای ارائه‌ی توصیه است. دایره‌ی زرد و دایره‌ی خاکستری بیانگر کاربران و آیتم‌های مهمی است که توسط روابط مرتبه بالا کشف شده‌اند اما در روش‌های سنتی از آن‌ها غافل می‌شوند.

۱. انتشار تعبیه‌ی بازگشتی، که تعبیه‌ی یک گره را بر اساس تعبیه‌ی همسایگانش به‌روز می‌کند و به صورت بازگشتی انجام چنین انتشار تعبیه‌ی را برای گرفتن اتصالات مرتبه بالا در یک پیچیدگی زمانی خطی انجام می‌دهد.

۲. جمع‌بندی مبتنی بر توجه، که از مکانیسم توجه عصبی برای یادگیری وزن هر همسایه در طول انتشار استفاده می‌کند، به گونه‌ای که وزن توجه انتشارات پی‌درپی می‌تواند اهمیت اتصال مرتبه بالا را نشان دهد.

مدل ارائه شده‌ی KGAT [۶] مزیت‌هایی نسبت به روش‌های موجود دارد:

۱. در مقایسه با روش‌های مبتنی بر مسیر، از فرایند شدید کار در تحقق مسیر جلوگیری می‌کند، این برای استفاده کارآمدتر و راحت‌تر است.
۲. در مقایسه با روش‌های مبتنی بر نظم، مستقیماً روابط مرتبه بالایی را در مدل پیش‌بینی‌کننده ایجاد می‌کند، بنابراین تمام پارامترهای مرتبط برای بهینه سازی هدف توصیه متناسب هستند.

۲.۷ فرمول بندی کار

۱.۲.۷ گراف دوطرفه‌ی کاربر-آیتم

در سناریوهای توصیه، معمولاً با تعامل کاربر و آیتم روبرو هستیم که می‌توان آن را به صورت یک گراف دوطرفه به صورت $\{(u, y_{ui}, i) \mid u \in U, i \in I\}$ تعریف کرد.

در این تعریف U و I در واقع به ترتیب مجموعه‌ی کاربران و آیتم‌ها هستند و y_{ui} نیز نشان می‌دهد که بین کاربر u و آیتم i تعامل وجود دارد یا خیر (اگر ۱ باشد وجود تعامل و ۰ عدم وجود تعامل را نشان می‌دهد).

۲.۲.۷ گراف دانش

علاوه بر اطلاعات تعاملی، اطلاعات جانبی نیز برای آیتم‌ها موجود است که از موجودیت‌های دنیای واقعی تشکیل شده است. این اطلاعات جانبی را در قالب گراف جهت‌داری که متشکل از سه‌گانه‌های نهاد-صفت-شیء^۵ به صورت $\{(h, r, t) \mid h, t \in E, r \in R\}$ است.

^۵subject-property-object

این سه‌گانه بیان می‌کند که رابطه‌ی r بین موجودیت راس h و موجودیت انتهایی t وجود دارد البته این رابطه در جهت برگشت نیز برقرار است (برای مثال اگر داشته باشیم ”هاگ جکمن، بازیگر در، لوگان“ می‌توان در جهت برگشت رابطه‌ی بین هاگ جکمن و لوگان را با، ”بازی شده توسط“ بیان کنیم.

۳.۲.۷ گراف دانش مشارکتی

این گراف به این صورت عمل می‌کند که گراف دوطرفه‌ی کاربر-آیتم را با گراف دانش ترکیب می‌کند و نتیجه به صورت یک سه‌تایی حاصل می‌شود و روابط از تعامل بین کاربر و آیتم و روابط گراف دانش نتیجه می‌شود و موجودیت‌های دو طرف روابط از موجودیت‌های گراف دانش و کاربران گراف دوطرفه‌ی کاربر-آیتم می‌آید.

$$G = \{(h, r, t) | h, t \in E', r \in R'\}, \text{ where } E' = E \cup U \text{ and } R' = R \cup \{Interact\}$$

شرح کار: حال کارپیشنهاددهی را که در این مقاله باید به آن پرداخته شود به شیوه‌ی زیر فرموله می‌شود:
ورودی: گراف دانش مشارکتی که شامل گراف دوطرفه‌ی کاربر-آیتم و گراف دانش است.
خروجی: یک تابع پیش‌بینی که احتمال استفاده‌ی کاربر u از آیتم i را پیش‌بینی می‌کند.

۳.۷ ارتباط مرتبه‌بالا

استفاده از اتصال به مرتبه‌بالا برای انجام توصیه‌های با کیفیت بالا از اهمیت ویژه‌ای برخوردار است. زنجیره‌ای که در این گراف جدید تشکیل می‌شود به این صورت است و L در اینجا طول دنباله است.

$$e_0 \xrightarrow{r_1} e_2 \xrightarrow{r_2} \dots \xrightarrow{r_L} e_L, \text{ where } e_i \in E' \text{ and } r_i \in R'$$

برخلاف، فیلتر مشارکتی که بر مبنای استفاده‌های کاربرانی است که علایق مشترک دارند، مدل‌های یادگیری با نظارت مانند FM^* و NFM^* ، براساس ویژگی‌ها و صفات آیتمی است که توسط کاربر مورد نظر استفاده شده است. برای مثال؛ رابطه‌ی زیر پیشنهاد می‌کند که u_1 ممکن است از i_2 استفاده کند بخاطر کارگردان مشترک آن e_1 با آیتم i_1 که پیش از این پسندیده است.

$$u_1 \xrightarrow{r_1} i_1 \xrightarrow{r_2} e_1 \xrightarrow{-r_2} i_2$$

البته از نواقص مدل ذکر شده این است که یکسری ارتباطات مرتبه بالا را پشتیبانی نمی‌کند. مثل این موضوع که جنس ارتباط صفت با دوتا آیتم متمایز، متفاوت است.

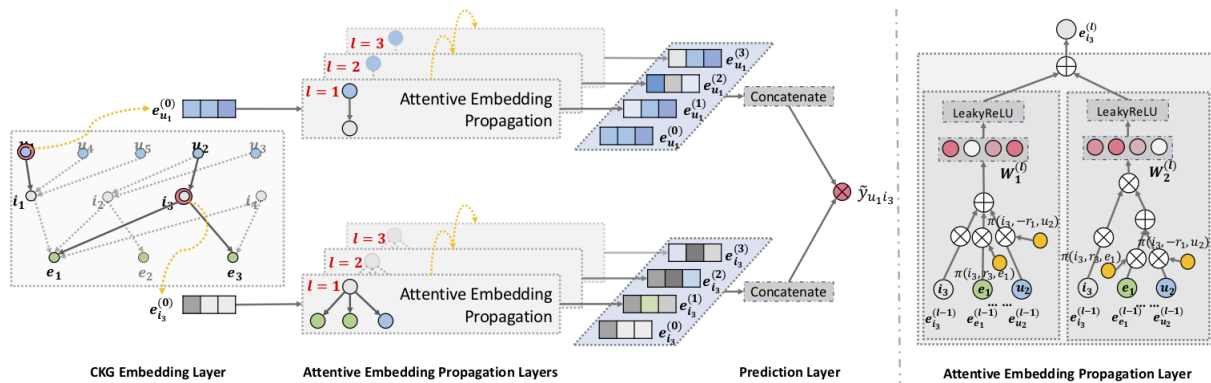
۴.۷ روش شناسی

براساس ۲.۷ که ساختار مدل را نشان می‌دهد؛ می‌توان بیان کرد که مدل از سه جزء اصلی تشکیل شده است:

۱. لایه‌ی نهفته‌سازی که با حفظ ساختار گراف دانش مشارکتی هر گره را به عنوان یک بردار پارامتر می‌کند؛
۲. لایه‌ی انتشار تعبیه‌های مورد توجه که به طور بازگشتی تعبیه‌ها را از همسایه‌های یک گره برای به‌روزرسانی بازنمایی آن، منتشر می‌کند و از مکانیسم توجه آگاه به دانش برای یادگیری وزن هر همسایه در طول یک انتشار استفاده می‌کند؛

⁶factorization machine

⁷neural factorization machine



تصویر ۲۰۷: نمایی از مدل KGAT

۳. لایه‌ی پیش‌بینی که بازنمایی‌های هر کاربر و آیت‌م را از تمام لایه‌های انتشار جمع‌آوری می‌کند و امتیاز تطابق پیش‌بینی شده را به عنوان خروجی می‌دهد.

۱.۴.۷ لایه‌ی تعبیه سازی

تعبیه‌سازی گراف دانش یک راه موثر برای پارامترکردن موجودیت‌ها و روابط به عنوان بازنمایی برداری است؛ در حالی که ساختار گراف حفظ می‌شود. در اینجا از متد TransR استفاده می‌شود. پس تابع امتیازدهی به صورت زیر خواهد بود:

$$g(h, r, t) = \|W_r e_h + e_r - W_r e_t\|_2^2 \quad (۱.۷)$$

آموزش TransR در واقع ترتیب نسبی بین سه‌تایی‌های معتبر و سه‌تایی‌های بدون اتصال را در نظر می‌گیرد و تفاوت آنها را با تابع هزینه‌ی رتبه بندی دوطرفه به صورت زیر نشان می‌دهد:

$$L_{KG} = \sum_{(h, r, t, t') \in T} -\ln \sigma(g(h, r, t') - g(h, r, t)) \quad (۲.۷)$$

که در رابطه‌ی بالا سه‌تایی (h, r, t) عضو گراف است و معتبر است ولی سه‌تایی دیگر داخل گراف موجود نیست. این لایه موجودیت‌ها و روابط موجود بر روی ریزدانگی سه‌گانه را مدل می‌کند، به عنوان یک تنظیم‌کننده کار می‌کند و اتصالات مستقیم را به بازنمایی‌ها تزریق می‌کند و بنابراین توانایی نمایش مدل را افزایش می‌دهد.

۲.۴.۷ لایه‌ی انتشار نهفته‌های مورد توجه

بعد، بر پایه معماری شبکه‌ی کانولوشن گراف بنا می‌شود تا به صورت بازگشتی، تعبیه‌ها در امتداد اتصال مرتبه بالا منتشر شود؛ علاوه بر این، با بهره‌برداری از ایده شبکه توجه گراف، وزن‌های مهم از انتشار پی‌درپی تولید می‌شود تا اهمیت چنین اتصالاتی نشان داده شود. انتشار اطلاعات: یک موجودیت در چند سه‌گانه می‌تواند حضور داشته باشد و نقش یک پل برای انتشار اطلاعات داشته باشد برای مثال در روابط زیر، i_2 اطلاعات e_1 و e_2 را به عنوان ورودی دریافت می‌کند و ویژگی‌ها و صفات خود را غنی می‌کند؛ در واقع نقش یک پل برای انتشار و انتقال اطلاعات از e_2 به u_2 را دارد. در واقع به این صورت ارتباط و انتشار اطلاعات از یک موجودیت به همسایگان خود اتفاق می‌افتد.

$$e_1 \xrightarrow{r_2} i_2 \xrightarrow{-r_1} u_2$$

$$e_2 \xrightarrow{r_3} i_2 \xrightarrow{-r_1} u_2$$

اگر موجودیت h را در نظر بگیریم، می‌توانیم بیان کنیم N_h که به صورت $N_h = \{(h, r, t) | (h, r, t) \in G\}$ تعریف می‌کنیم؛ شبکه‌ای است که در راس آن h قرار دارد و در واقع یک خود شبکه h است که به h مربوط است. برای توصیف ساختار اتصال مرتبه اول موجودیت h ، ترکیب خطی خود شبکه‌ی آن را محاسبه می‌کنیم:

$$e_{N_h} = \sum_{(h, r, t) \in N_h} \pi(h, r, t) e_t \quad (3.7)$$

که $\pi(h, r, t)$ عامل زوال در هر انتشار در یال (h, r, t) را کنترل می‌کند، و نشان می‌دهد که به شرط رابطه r چه مقدار اطلاعات از t تا h انتشار می‌یابد. توجه آگاه به دانش: حال $\pi(h, r, t)$ در فرمول بالا را به صورت زیر پیاده سازی می‌کنیم که در واقع در آن از تابع غیر خطی \tanh که ضرب توجه را به فاصله‌ی بین e_t و e_h در فضای r وابسته می‌کند. برای مثال؛ انتشار اطلاعات بیشتر از موجودیت‌های نزدیک‌تر. پس از آن به نرمال سازی ضرایب سه تایی های متصل به h با استفاده از فرمول زیر که در آن سه تایی های موجود در صورت و مخرج کسر هردو در گراف دانش هستند، می‌پردازیم:

$$\pi(h, r, t) = \frac{\exp(\pi(h, r, t))}{\sum_{(h, r', t') \in N_h} \exp(\pi(h, r', t'))} \quad (4.7)$$

به عنوان نتیجه متوجه می‌شویم که، امتیاز توجه نهایی می‌تواند نشان دهد که کدام یک از گره‌های همسایه باید بیشتر برای گرفتن سیگنال‌های مشارکتی مورد توجه قرار گیرد.

متمایز از انتشار اطلاعات در شبکه‌ی کانولوشن گراف و GraphSage که ضریب نزول را بین دو گره به عنوان $1/|N_t|$ or $1/\sqrt{|N_h||N_t|}$ تعیین می‌کند، مدل نه تنها از ساختار مجاورت گراف (ساختار همسایه بودن گره‌ها) بهره می‌برد، بلکه اهمیت تفاوت همسایگان را نیز مشخص می‌کند. علاوه براین، متمایز از شبکه توجه گراف که فقط گره‌ها را به عنوان ورودی معرفی می‌کند، رابطه بین e_t و e_h مدل می‌شود و اطلاعات بیشتری در حین انتشار رمز می‌شود.

تجمع اطلاعات: در آخر تعبیه‌ی گره h را که گره راس یک خود شبکه کوچک است را با تعبیه‌ی آن شبکه توسط تابع f ترکیب می‌کنیم و به عنوان نمایش جدید h ارائه می‌کنیم. برای بدست آوردن $f(0)$ می‌توان از سه نوع تجمع زیر استفاده کرد:

- تجمع کننده ی شبکه‌ی کانولوشن گراف که در واقع دو بازنمایی را جمع می‌کند و با استفاده از یک انتقال غیرخطی حاصل را به صورت زیر محاسبه می‌کند:

$$f_{CGN} = \text{LeakyReLU}(W(e_h + e_{N_h})) \quad (5.7)$$

در فرمول بالا W در واقع ماتریس وزن قابل یادگیری است که اطلاعات مفید برای انتشار را بیرون می‌کشد و نگهداری می‌کند همچنین تابع LeakyReLU به عنوان تابع فعال سازی عمل می‌کند.

- تجمع کننده‌ی GraphSage نیز مکانیزمی شبیه به تجمع کننده‌ی شبکه‌ی کانولوشن گراف دارد با این تفاوت که دو بازنمایی را بهم الحاق می‌کند:

$$f_{\text{GraphSage}} = \text{LeakyReLU}(W(e_h || e_{N_h})) \quad (6.7)$$

⁸ego network

• تجميع کننده‌ی تعامل دوگانه با دقت طراحی شده تا دونوع تعامل صفات بين eh و eNh به صورت زیر در نظر بگیرد:

$$f_{BiInteraction} = LeakyReLU(W(e_h + e_{N_h})) + LeakyReLU(W(e_h \odot e_{N_h})) \quad (7.7)$$

این فرمول در واقع از ترکیب جمع این دو تعبیه و ضرب المانی آنها تشکیل شده است و البته W_1 و W_2 ماتریس‌های وزن قابل یادگیری هستند.

متمایز از جمع کننده شبکه‌ی کانولوشن گراف و GraphSage ما علاوه بر این تعامل ویژگی بین e_h و e_{N_h} را رمزگذاری می‌کنیم. این اصطلاح باعث می‌شود که اطلاعات منتشر شده حساس به مشابهت بین e_h و e_{N_h} شوند و در واقع این نزدیکی باعث می‌شود که پیام‌های بیشتری بینشان رد و بدل شود. مزیت لایه انتشار تعبیه در بهره‌برداری صریح از اطلاعات اتصال مرتبه اول برای ارتباط با کاربر، آیتم و بازنمایی موجودیت دانش، قرار دارد. انتشار مرتبه بالا: می‌توان برای یافتن اطلاعات اتصال مرتبه بالا، جمع‌آوری اطلاعات منتشر شده از همسایه‌های لایه بالاتر، لایه‌های انتشار بیشتری در نظر گرفت. در لایه l ام، نمایش موجودیت (و تعبیه‌ها) به صورت زیر محاسبه می‌شود:

$$e_h^l = f(e_h^{l-1}, e_{N_h}^{l-1}) \quad (8.7)$$

اطلاعات منتشر شده توسط خود شبکه در لایه l ام به صورت زیر محاسبه می‌شود که در آن $e_t^{(l-1)}$ در واقع بازنمایی موجودیت t از انتشار اطلاعات در گام قبلی است و اطلاعات همسایگان لایه $l-1$ را نگهداری می‌کند:

$$e_{N_h}^{l-1} = \sum_{(h,r,t) \in N_h} \pi(h, r, t) e_t^{l-1} \quad (9.7)$$

به عنوان نتیجه، اتصالات مرتبه بالا همانند $u_1 \xrightarrow{-r_1} i_1 \xrightarrow{r_2} e_1 \xrightarrow{-r_2} i_2 \xrightarrow{r_1} u_2$ در پروسه‌ی انتشار تعبیه قابل حصول هستند و در واقع اطلاعات مربوط به گره راس (u_2) به صورت صریح در تعبیه‌ی گره انتها (u_1) که با سه گره میانی به گره راس متصل است، کد شده است.

۳.۴.۷ پیش بینی مدل

پس از بدست آوردن لایه‌ی L ام، بازنمایی‌های چنگانه از گره کاربر u به صورت $\{e_u^{(1)}, \dots, e_u^{(L)}\}$ حاصل می‌شود؛ همچنین برای گره‌های آیتمی نیز بازنمایی‌ها به صورت $\{e_i^{(1)}, \dots, e_i^{(L)}\}$ هستند. همانطور که خروجی لایه l ام، تجمع پیام از عمق l ام ساختار درختی است که یا u (i ریشه آن است) (۱۰.۷)، خروجی‌های موجود در لایه‌های مختلف بر اطلاعات اتصال مرتبه مختلف تأکید می‌کند. از این رو مکانیزم تجميع لایه را برای الحاق بازنمایی‌های هر گام به یک بردار واحد، اتخاذ شده که به صورت زیر است:

$$e_u^* = e_u^{(0)} || \dots || e_u^{(L)}$$

$$e_i^* = e_i^{(0)} || \dots || e_i^{(L)}$$

با این کار، نه تنها با انجام عملیات تکثیر تعبیه، تعبیه‌های اولیه را غنی می‌کنیم؛ بلکه امکان تنظیم قدرت انتشار را نیز با تنظیم L فراهم می‌آوریم. در پایان، ما ضرب داخلی از نمایش کاربر و آیتم‌ها را انجام می‌دهیم، تا امتیاز تطبیق آن‌ها را پیش بینی کنیم.

$$\hat{y}(u, i) = e_u^* \top e_i^* \quad (10.7)$$

۵.۷ بهینه سازی

برای بهینه سازی مدل، تابع هزینه BPR استفاده می‌شود. فرض در این قضیه این است که تعاملات مشاهده شده‌ی کاربر که نمایانگر ترجیحات کاربر است، لازم است مقدار پیش بینی بالاتری نسبت به موارد مشاهده نشده داشته باشند:

$$L_{CF} = \sum_{(u,i,j) \in O} -\ln \sigma(\hat{y}(u,i) - \hat{y}(u,j)) \quad (11.7)$$

که در فرمول بالا $O = \{(u,i,j) | (u,i) \in R^+, (u,j) \in R^-\}$ برقرار است و R^+ نمایانگر تعاملات مشاهده شده بین u و j است و R^- تعاملات مشاهده نشده.

در آخر تابع هدف به صورت زیر خواهد بود:

$$L_{KGAT} = L_{KG} + L_{CF} + \lambda \|\Theta\|_2^2 \quad (12.7)$$

که در اینجا $\Theta = \{E, W_r, \forall l \in R, W_1^{(l)}, W_2^{(l)}, \forall l \in \{1, \dots, L\}\}$ در واقع مجموعه‌ی پارامترهای مدل است و E نمایانگر جدول تعبیه‌های تمام موجودیت‌ها و روابط است.

۱.۵.۷ یادگیری

در اینجا برای بهینه سازی مداوم L_{KG} و L_{CF} از تکنیک دسته‌ی کوچک آدام^۹ استفاده می‌شود که برای بهینه سازی هزینه‌ی تعبیه سازی و پیش بینی موجود است.

در واقع برای یک دسته‌ی نمونه‌ی تصادفی (h, r, t, t') تعبیه‌های تمام گره‌ها بروزرسانی می‌شود، سپس یک دسته از (u, i, j) به صورت نمونه‌ی تصادفی در نظر گرفته می‌شود و پس از L مرحله انتشار، بازنمایی آن بازیابی شده و در آخر پارامترهای مدل با استفاده از گرادینت هزینه‌ی پیش بینی به روزرسانی می‌شود.

۲.۵.۷ تحلیل پیچیدگی زمانی

پیچیدگی زمانی این مدل از دو قسمت به وجود می‌آید: تعبیه سازی گراف دانش و انتشار تعبیه‌ها با استفاده از مکانیسم توجه. از آنجایی که سرویس‌های آنلاین نیاز به محاسبات در زمان واقعی دارند، هزینه در زمان استنتاج و نتیجه گیری بسیار مهم‌تر از زمان آموزش است. با توجه به اینکه در روش‌های موجود به این صورت است که $KGAT, GCMC, CKE, CGKG, NFM, FM$ و $MCRec$ به ترتیب دارای هزینه‌های ۷۰۰ ثانیه، ۷۸۰ ثانیه، ۸۰۰ ثانیه، ۴۲۰ ثانیه، ۵۰۰ ثانیه، ۵۶۰ ثانیه، ۲۰ ساعت و ۲ ساعت بر روی دیتاست کتاب آمازون هستند که نشان می‌دهد این روش یک روش قابل مقایسه و کارآمد در مقایسه با روش‌های یادگیری بانظارت و مبتنی بر نظم و همچنین موثرتر از مدل مبتنی بر مسیر است.

۶.۷ نتیجه گیری

در این کار، به منظور ارتقاء دانش آگاهی، ارتباط مرتبه ای با روابط معنایی در CKG بررسی می‌شود. یک چارچوب جدید KGAT معرفی شده است، که صریحاً مدل اتصالات مرتبه بالا در CKG را با روشی انتها به انتها مدل می‌کند. در هسته اصلی، لایه انتشار توجه تعبیه شده، که به صورت انطباق پذیر، تعبیه‌های همسایگان گره را برای به روزرسانی بازنمایی گره پخش می‌کند.

⁹mini-batch Adam

این کار به بررسی پتانسیل گراف شبکه های عصبی در توصیه، و نشان دهنده یک تلاش اولیه برای بهره برداری دانش ساختاری با نحوه انتشار اطلاعات است. علاوه بر گراف دانش، بسیاری از اطلاعات ساختاری دیگر در سناریوهای دنیای واقعی مانند شبکه های اجتماعی و متون آیتم وجود دارند. به عنوان مثال، با یکپارچه سازی شبکه های اجتماعی با CKG، می توان بررسی کرد چگونه نفوذ اجتماعی بر روی توصیه تاثیر می گذارد. جهت جالب دیگر ادغام انتشار اطلاعات و فرایند تصمیم گیری است که امکانات تحقیق از توصیه های توجیه پذیر را فراهم می کند.

IGNN

۱.۸ مقدمه

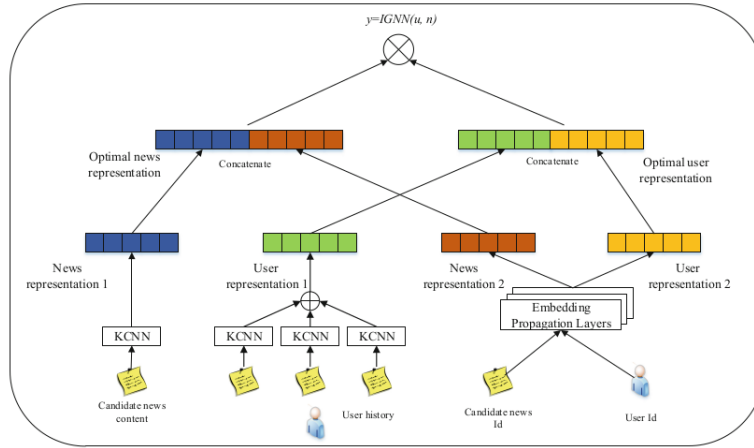
با توسعه سریع اینترنت، عادات خواندن اخبار افراد به تدریج از رسانه های سنتی مانند روزنامه ها و تلویزیون به اینترنت منتقل شده است. با این حال، به دلیل اینکه برنامه های خبری هر روز اخبار گسترده ای را ارائه می دهند، خوانندگان اخبار در این اطلاعات غرق می شوند. بنابراین برای بهبود رضایت کاربران، مشکل اصلی نحوه انتخاب خبرهایی است که کاربران علاقه مند به آن هستند و می خواهند آن ها را بخوانند. در سال های اخیر پیشنهاد اخبار به کاربران به صورت شخصی سازی شده یکی از مشکلات مطرح در سیستم های توصیه گر بوده است. رشد بسیار سریع اخبار، تعداد زیاد آن ها و ویژگی های خاص این نوع اطلاعات که فقط در زمان مناسب پتانسیل جذب کاربران را دارد این نوع پیشنهاد دهی را به خصوص کرده است؛ به دلیل ویژگی های خاص اخبار روش های مرسوم مثل فیلتر مشارکتی برای پیشنهاد دادن آن ها کارایی مناسبی ندارند، البته مدل های محتوا محور برای این سیستم ها طراحی شده اند که کارایی خوبی برای این نوع خاص داشته اند ولی روش های محتوا محور مورد استفاده به اندازه کافی از داده های مهمی که در تعاملات کاربر با اخبار وجود دارد استفاده نمی کنند؛ در این مقاله مدل جدیدی (IGNN) [۷] برای پیشنهاد دادن اخبار معرفی شده است که گراف تعاملات کاربر-آیتم و یک گراف دانش را به سیستم توصیه گر محتوا محور اضافه کرده است. IGNN برای بدست آوردن اطلاعات کاربران و آیتم ها از دو گراف استفاده می کند، این مدل ویژگی ها محتوا محور را از سطح دانش و ویژگی های معنایی را به کمک شبکه عصبی کانولوشن استخراج می کند، همچنین از گراف تعاملات کاربر-آیتم سیگنال های مشارکتی سطح بالا را استخراج کرده و در بازنمایی موجودیت ها و آموزش مدل از این سیگنال ها نیز استفاده می کند. این مدل پیشرفت زیادی در این زمینه به وجود آورده است که بعد از بررسی آزمایش های انجام شده روی دو دیتابیس در دنیای واقعی مربوط به اخبار می توانیم این موضوع را مشاهده کنیم. در ادامه به بررسی این مدل، ساختار و لایه های مختلف آن می پردازیم سپس آزمایش ها روی مدل های معروف در کنار IGNN و نتایج آن ها را بررسی می کنیم.

۲.۸ معرفی مدل

در این بخش به بررسی مدل مورد استفاده در این مقاله برای پیشنهاد اخبار می پردازیم، همانطور که در تصویر ۱.۸ هم قابل مشاهده است این فریم ورک از سه قسمت اصلی تشکیل شده است:

۱. یک شبکه عصبی کانولوشن مبتنی بر دانش^۱ (KCNN)؛ که از اطلاعات ساختاری کلمات و شباهت بالقوه محتوای اخبار که در سطح دانش به آن دسترسی داریم استفاده می کند.

¹Knowledge-aware Convolutional Neural Network



تصویر ۱.۸: مدل IGNN

۲. لایه انتشار تعبیه؛ که اطلاعات اتصالات سطح بالا را به تعبیه‌ها تزریق می‌کند.

۳. لایه پیش‌بینی؛ که بعد از دریافت اطلاعات بهینه کاربران و اخبار از مراحل قبل، امتیاز نهایی برای اخبار هدف را محاسبه می‌کند.

همانطور که در تصویر ۱.۸ قابل مشاهده است ورودی شامل محتوا و شناسه اخبار هدف و برای یک کاربر خاص محتوای اخباری که قبلاً خوانده است و شناسه کاربر خواهد بود. اطلاعات سطح دانش در این مدل از گراف دانش معروف ویکی‌پدیا استخراج شده است؛ بعد از استخراج کلمات کلیدی یک اخبار اطلاعات متناظر آن‌ها در این گراف دانش جستجو شده و روابط سه تایی موجود از گراف دانش داندلود می‌شود.

IGNN اطلاعات سطح معنایی و سطح دانش را توسط KCNN استخراج می‌کند و اطلاعات سیگنال مشارکتی سطح بالا بین کاربران و اخبار را با استفاده از لایه انتشار تعبیه استخراج می‌کند و در نهایت نتیجه بدست آمده از دو مرحله قبل را در کنار هم قرار داده و نتیجه نهایی یعنی میزان علاقه کاربر را از این نتایج پیش‌بینی می‌کند.

۱.۲.۸ شبکه عصبی کانولوشن آگاه به دانش

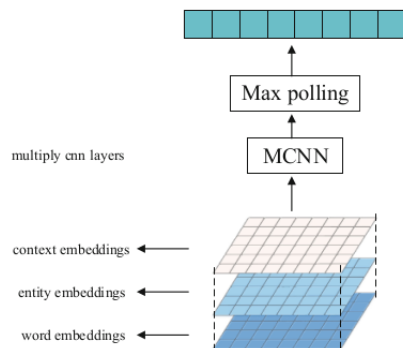
در این قسمت به بررسی ساختار KCNN می‌پردازیم.

در این مقاله از TransE برای تعبیه گراف دانش استفاده شده است و در این مقاله برای تقویت اختلاف بین سه تایی‌های درست و غلط در مدل TransE از تابع هزینه مبتنی بر حاشیه‌ی زیر استفاده شده است:

$$L = \sum_{(h, r, t) \in \Delta} \sum_{(h', r', t') \in \Delta'} \max(0, f_r(h, t) + \gamma + f_r(h', t')) \quad (1.8)$$

که γ حاشیه را مشخص می‌کند و Δ و Δ' مجموعه سه تایی‌های درست و غلط هستند.

همانطور که در تصویر ۲.۸ می‌توانیم مشاهده کنیم ورودی KCNN از سه ماتریس تشکیل شده است:



ماتریس زیرین، ماتریس تعبیه‌های کلمات در تیتیر اخبار است. اگر $t = w_{1:n}$ تیتیر یکی از اخبار باشد هر بردار w_i بردار متناظر با تعبیه این کلمه در تیتیر خبر است و در کنار هم قرار گرفتن این بردارها ماتریس اول را می‌سازد.

هر کدام از این وکتورها می‌توانند با وکتور تعبیه موجودیت و تعبیه محتوای متناظر با آن که توسط مدل TransE آموخته شده‌اند همراه باشند که دو ماتریس دیگر ورودی را می‌سازند.

تصویر ۲.۸: ورودی شبکه عصبی

محتوای یک موجودیت در این گراف از همسایه‌های بی‌واسطه هر کدام از آن‌ها به دست می‌آید، در واقع:

$$context(e) = \{e_i | (e, r, e_i) \in G \text{ or } (e_i, r, e) \in G\} \quad (2.8)$$

که G گراف دانش و r رابطه بین موجودیت‌هاست. چون موجودیت‌های محتوایی از نظر معنایی و منطقی خیلی به موجودیت اصلی مشابه هستند استفاده از آن‌ها در این مدل می‌تواند کمک کند اطلاعات بیشتری از ورودی دریافت کنیم. موجودیت‌های معنایی نیز از رابطه زیر محاسبه می‌شوند:

$$\bar{e} = \frac{1}{|context(e)|} \sum_{e_i \in context(e)} e_i \quad (3.8)$$

پس در نهایت با قرار دادن این سه ماتریس در کنار هم ورودی برای این قسمت ساخته می‌شود، سپس روی این ورودی چند فیلتر (m) را اعمال می‌کنیم تا بتوانیم اطلاعات محتوا محور را از این داده‌ها استخراج کنیم:

$$map = [[w_1, w_2, \dots, w_n], [e_1, e_2, \dots, e_n], [\bar{e}_1, \bar{e}_2, \dots, \bar{e}_n]] \in R^{d \times n \times 3 \times} \quad (4.8)$$

$$c^{m_i} = f(m_i * map + b) \quad (5.8)$$

و در آخر روی خروجی این فیلترها از یک لایه Max pooling استفاده می‌کنیم تا بزرگترین ویژگی را از خروجی مرحله قبل به دست آوریم.

$$c_{max} = \max(c^{m_1}, c^{m_2}, \dots, c^{m_l}) \quad (6.8)$$

۲.۲.۸ لایه انتشار تعبیه

در لایه انتشار، سیگنال‌های مشارکتی از ساختار گراف استخراج می‌شود و با استفاده از معماری تبادل پیام در گراف برای کاربران و آیتم‌ها بازنمایی به دست می‌آید.

در واقع می‌توانیم اینگونه نگاه کنیم که از تاریخچه خواندن اخبار هر کاربر می‌توانیم به علاقه‌ی او پی ببریم و از طرفی کاربرانی که یک خبر مشابه را می‌خوانند را می‌توانیم به عنوان یک ویژگی برای اخبار حساب کنیم که می‌تواند روی سیگنال مشارکتی برای اخبار تاثیر داشته باشد پس از همین موضوع استفاده می‌شود تا از انتشار اطلاعات بین کاربران و اخبار متصل در گراف سیگنال‌های مشارکتی استخراج شود.

این لایه از دو قسمت اصلی تشکیل شده است: ارسال پیام و تجمیع پیام‌ها. در این مقاله ارسال پیام بین یک کاربر u و خبر n متصل اینگونه فرموله بندی شده است:

$$M_{u \leftarrow n} = w(e_n, e_u, p_{u,n}) \quad (7.8)$$

که m اطلاعاتی است که بین کاربر و خبر منتقل می‌شود و $w(\cdot)$ تابعی است که این اطلاعات را مشخص می‌کند و به صورت زیر توصیف می‌شود:

$$M_{u \leftarrow n} = \frac{1}{\sqrt{|N_u| |N_n|}} (W_1 e_n + W_2 (e_n \odot e_u)) \quad (8.8)$$

در رابطه ۸.۸ W_1 و W_2 وزن‌هایی برای یادگیری اطلاعات مهم در این انتشار هستند و $e_n \odot e_u$ نشان‌دهنده ضرب نظیر به نظیر دو وکتور است که برای اعمال ارتباط بین دو موجودیت در رابطه مورد استفاده قرار می‌گیرد، این عمل باعث می‌شود که اطلاعات بیشتری بین دو خبر مشابه منتشر شود که نه تنها باعث بهتر شدن بازنمایی‌های حاصل خواهد بود بلکه اجرای این عملیات را نیز سرعت می‌بخشد.

و عبارت $\frac{1}{\sqrt{|N_u| |N_n|}}$ نرم لاپلاسین است که با $p_{u,n}$ در مرحله قبل مشخص شد، در این رابطه N_u و N_n مجموعه اولین همسایه‌های بی‌واسطه n و u هستند.

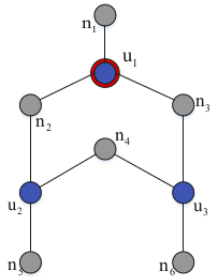
در مرحله دوم یعنی تجميع پیام‌ها، سعی می‌شود که اطلاعاتی که از طریق پیام‌ها همسایه‌های هر گره به او رسیده است جمع بندی شود. برای اینکار از تابع زیر استفاده می‌شود:

$$e_u^{(1)} = \varphi(e_u, m_{u \leftarrow u} | i \in N_u) \quad (9.8)$$

که $e_u^{(1)}$ نتیجه نهایی یعنی یک بازنمایی جدید برای کاربر u است. برای استفاده از اطلاعات به دست آمده از تابع انتقال غیرخطی زیر استفاده شده است:

$$e_u^{(1)} = \psi(m_{u \leftarrow u} + \sum_{i \in N_u} m_{u \leftarrow i}) \quad (10.8)$$

تابع فعال‌سازی مورد استفاده در این قسمت Leaky ReLU است. در این مقاله برای اینکه از اطلاعات اصلی هر کدام از موجودیت‌ها نیز به خوبی استفاده شود برای هر کدام از گره‌های گراف یک اتصال به خود نیز تعریف شده است که به صورت $m_{u \leftarrow u} = W_1 e_u$ در مرحله ارسال پیام‌ها استفاده می‌شود. و به طور مشابه بازنمایی اخبار نیز با تجميع پیام‌های دریافت شده محاسبه می‌شود.



برای استفاده از اتصالات سطح بالاتر در گراف از پخش اطلاعات روی گراف با محدودیت گام‌های بیشتر استفاده شده است در واقع هر کدام از اخبار یا کاربران می‌توانند به صورت مستقل توسط پیام‌هایی که از همسایه‌های آن‌ها با فاصله مشخص به او رسیده است بازنمایی شوند. استفاده از این سیگنال مشارکتی سطح بالا در مدل کمک می‌کند که تخمین دقیق‌تری از میزان مشابهت اخبار و کاربران به دست آوریم. با پشته‌سازی k لایه انتشار تعبیه، یک کاربر یا خبر می‌تواند پیام‌هایی که از همسایه‌های او با فاصله k هستند را به بدست آورد که به صورت زیر قابل توصیف است:

$$e_u^{(k)} = \psi(m_{u \leftarrow u}^{(k)} + \sum_{i \in N_u} m_{u \leftarrow i}^{(k)}) \quad (11.8)$$

تصویر ۳.۸: اتصالات

سطح بالا مربوط به کاربر u_1

و اطلاعات پیام‌هایی که منتقل می‌شوند نیز به صورت زیر محاسبه می‌شود:

$$m_{u \leftarrow u}^{(k)} = \frac{1}{\sqrt{|N_u| |N_n|}} (W_1^{(k)} e_n + W_2^{(k)} (e_n^{(k-1)} \odot e_u^{(k-1)})) \quad (12.8)$$

و همانطور که در رابطه ۱۲.۸ مشخص است در هر مرحله از بازنمایی مرحله قبل برای محاسبه اطلاعات پیام در مراحل بعد استفاده می‌شود. اگر عملیات توصیف شده را روی تصویر ۳.۸ که بازنمایی سطح بالای گراف است بررسی کنیم متوجه می‌شویم که سیگنال مشارکتی به صورت $i_4 \leftarrow u_2 \leftarrow i_2$ در گراف منتقل می‌شود.

۳.۲.۸ مدل پیش‌بینی

پس از استفاده انتشار توسط k لایه و استفاده از شبکه عصبی کانولوشن یک بازنمایی کامل برای هر کدام از اخبار و کاربران به دست آمده است که با متصل کردن خروجی دو مرحله قبل به دست می‌آید مثلاً برای خبر n اتصال دو قسمت یعنی $(e_n^1, e_n^2, \dots, e_n^k)$ و e_n^{cb} بازنمایی نهایی را تولید خواهد کرد:

$$e_n^* = e_n^{(0)} || e_n^{(1)} || \dots || e_n^{(k)} || e_n^{(cb)} \quad (13.8)$$

به همین صورت هم با استفاده از تاریخچه اخبار خوانده شده توسط هر کاربر و اطلاعات محتوا محور او بازنمایی نهایی حاصل می‌شود. برای استخراج بازنمایی محتوا محور کاربر از میانگین بازنمایی تاریخچه اخبار جدید وی استفاده می‌شود:

$$e_u^{cb} = \text{mean}(e_{n_1}^{(cb)}, e_{n_2}^{(cb)}, \dots, e_{n_q}^{(cb)}) \quad (۱۴.۸)$$

و با اتصال آن به خروجی لایه انتشار بازنمایی نهایی به این صورت خواهد بود:

$$e_u^* = e_u^{(0)} || e_u^{(1)} || \dots || e_u^{(k)} || e_u^{cb} \quad (۱۵.۸)$$

و در نهایت برای تخمین میزان علاقه‌مندی کاربر برای اخبار هدف از ضرب داخلی اطلاعات کاربر در هر کدام از اخبار هدف استفاده می‌شود:

$$y_{IGNN}(u, n) = e_u^* \top e_n^* \quad (۱۶.۸)$$

۴.۲.۸ بهینه سازی

مانند سایر مقالات در این زمینه برای بهینه‌سازی از تابع هزینه BPR استفاده شده است. برای هر دوتایی (u, n_{pos}) یک دوتایی رندوم (u, n_{neg}) از اخباری که یک کاربر نخوانده است انتخاب شده و هزینه با توجه به این دو جفت محاسبه می‌شود. تابع هدف مورد استفاده در این مقاله به صورت زیر محاسبه می‌شود:

$$loss_{IGNN} = \sum_{(u, n_{pos}, n_{neg}) \in r} -\ln \mu(\hat{y}_{IGNN}(u, n_{pos}) - \hat{y}_{IGNN}(u, n_{neg})) + \lambda ||\theta||_2^2 \quad (۱۷.۸)$$

۵.۲.۸ یادگیری

در این مقاله از تکنیک بخش‌بندی Adam برای بهینه سازی و به‌روزرسانی مقادیر مدل استفاده شده است. اگر چه مدل‌های یادگیری عمیق توانایی بالایی برای یادگیری و بازنمایی دارند، این مدل‌ها معمولاً مشکل تطبیق بیش از حد دارند در این پروژه برای جلوگیری از تطبیق بیش از اندازه مدل روی داده‌ها از دو لایه حذف تصادفی روی پیام‌ها و روی گره‌ها استفاده شده است.

در واقع در مرحله انتقال پیام‌ها به صورت رندوم تعدادی از پیام‌ها حذف می‌شوند و در روند یادگیری تأثیری نمی‌گذارند احتمال حذف هر پیام در این مدل با p_1 مشخص می‌شود و همین‌طور در روند یادگیری در هر مرحله تعدادی از گره‌ها با احتمال p_2 انتخاب می‌شوند تا پیام‌های خروجی آن‌ها در نتیجه مرحله l ام انتشار تأثیر نداشته باشد.

۳.۸ آزمایش‌ها

در این قسمت آزمایش‌های انجام شده برای بررسی و مقایسه مدل توصیف شده یعنی IGNN با مدل‌های دیگر در این حوزه را مورد مطالعه قرار می‌دهیم.

۱.۳.۸ دیتاست

برای بررسی کارآمدی مدل مورد نظر از دو دیتاست دنیای واقعی استفاده شده است. یکی از آن‌ها دیتاست مسابقه DC از شرکت Castle Data و دیتاست دوم دیتاست اخبار Adressa است. برای اطلاعات لازم برای گراف دانش از داده‌های

Method	DC dataset		Adressa	
	Recall@20	NDCG@20	Recall@20	NDCG@20
MF	0.58742	0.39962	0.10384	0.09046
LibFM	0.59834	0.44839	0.14758	0.11248
Deep-wide	0.60287	0.44894	0.13857	0.10295
NeuMF	0.65892	0.52183	0.16842	0.12485
GC-MC	0.63572	0.48834	0.17864	0.12047
DKN	0.67927	0.55856	0.19210	0.13385
IGNN	0.73180	0.59315	0.20728	0.15729

تصویر ۴.۸: نتایج آزمایش

ویکی‌پدیا استفاده شده است. در واقع هر کدام از موجودیت‌های موجود در دیتاست اخبار و همسایه‌های آن‌ها با یک فاصله از گراف دانش ویکی‌پدیا برای استفاده در مدل استخراج شده است.

۲.۳.۸ مقایسه کارآمدی

برای مقایسه کارآمدی روش پیشنهاد شده این روش با مدل‌های به‌روز در این زمینه مقایسه شده است که نتایج مقایسه را می‌توانیم در جدول ۴.۸ مشاهده کنیم: از نتایج حاصل می‌توانیم مشاهده کنیم که:

- مدل MF بدترین نتایج را بین مدل‌های نشان‌داده شده دارد که می‌توان از این مشاهده نتیجه‌گیری کنیم که مدل‌های یادگیری مدل‌های مناسب‌تری برای پیشنهاددهی هستند زیرا مدل‌های یادگیری عمیق توانایی بالاتری در بازنمایی دارند.
- مدل GC-MC نتایج بهتری نسبت به MF، Deepwide و LibFM داشته است. از این مشاهده نتیجه می‌گیریم که استفاده از همسایه‌های درجه اول در مدل می‌تواند نتایج بهتری برای بازنمایی موجودیت‌ها در بر داشته باشد که این موضوع نشان‌دهنده اهمیت استخراج و استفاده از اتصالات سطح بالا در گراف کاربر-آیتم است.
- در مقایسه با مدل‌های پایه DKN نتیجه بهتری نسبت به همه داشته است. این موضوع به این دلیل است که این مدل اطلاعات سطح دانش را استخراج کرده و این اطلاعات را با اطلاعات سطح معنا ترکیب می‌کند. در ضمن DKN برای تخمین علایق کاربران از network attention استفاده می‌کند.
- در آخر IGNN از همه مدل‌های پایه نتایج بهتری داشته است. به خصوص اینکه IGNN با پیشرفت ۹.۷ تا ۸.۱۶ درصدی با معیار Recall و پیشرفت ۵.۱۷ تا ۶.۳۰ درصدی با معیار NDCG در مقایسه با بهترین مدل پایه یعنی DKNN داشته است. دلیل این پیشرفت بزرگ در هر دو دیتاست نسبت به مدل‌های گذشته این است که IGNN نه تنها از اطلاعات سطح معنا و اتصالات سطح دانش استفاده می‌کند، بلکه اطلاعات اتصالات سطح بالای گراف کاربر-آیتم را نیز به کار می‌گیرد.

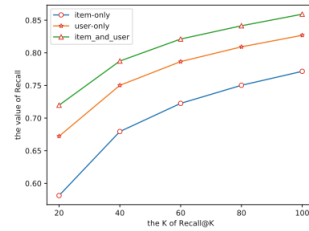
۳.۳.۸ بررسی مدل

از آنجایی که IGNN از بخش‌های مختلفی تشکیل شده است در این قسمت بررسی می‌کنیم که با تغییر هر کدام از این قسمت‌ها چه تغییری در نتیجه کلی حاصل می‌شود.

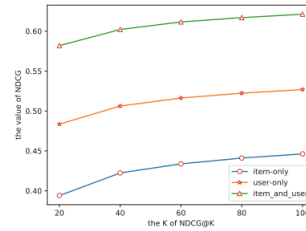
۱- در ابتدا کارآمدی مدل را با ترکیب‌های متفاوتی از لایه CNN آگاه به دانش و لایه انتشار تعبیه بررسی می‌کنیم. در مرتبه اول لایه انتشار تعبیه فقط برای کاربران استفاده می‌شود، در مرتبه دوم این لایه فقط برای اخبار استفاده می‌شود و در مرتبه سوم این لایه برای کاربران و اخبار به صورت همزمان استفاده می‌شود. نتیجه این آزمایش را می‌توانیم در نمودارهای ۵.۸ مشاهده کنیم.

همانطور که مشخص است استفاده همزمان لایه انتشار برای کاربران و اخبار بهترین نتیجه را داشته است که این همان مدل اصلی IGNN است. این موضوع را می‌توان اینگونه تحلیل کرد که استفاده از لایه انتشار برای کاربران یا اخبار به تنهایی ممکن است سیگنال مشارکتی بین آن‌ها را از دست بدهد.

۲ - می‌دانیم که IGNN می‌تواند از چند لایه به صورت انباشته شده برای لایه انتشار تعبیه استفاده کند پس بررسی می‌کنیم که تعداد لایه‌های مورد استفاده در نتیجه حاصل چه نتیجه‌ای خواهد داشت:



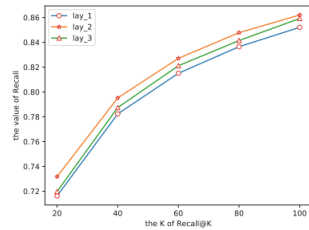
(a) Recall



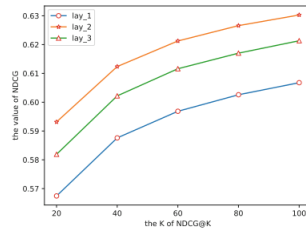
(b) NDCG

تصویر ۵.۸: تاثیر لایه انتشار

تعداد این لایه‌ها در بازه ۱ تا ۳ مورد بررسی قرار گرفته است، همانطور که در نمودارهای ۵.۸ می‌توانیم مشاهده کنیم استفاده از دو لایه برای لایه انتشار بهترین نتیجه را داشته است در صورتی که استفاده از یک لایه بدترین نتیجه را در هر دو معیار داشته است. یک لایه احتمالاً نتوانسته است که سیگنال مشارکتی بین آن‌ها را به خوبی استخراج کند و این موضوع که استفاده از سه لایه نتیجه بدتری نسبت به دو لایه دارد احتمالاً به دلیل یادگیری بیش از حد مدل بوده است.



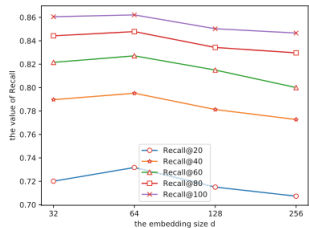
(a) Recall



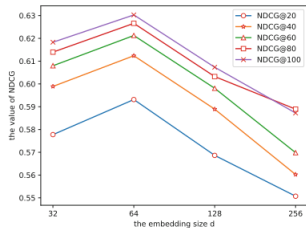
(b) NDCG

تصویر ۶.۸: تاثیر تعداد لایه‌های انتشار

۳- در آخر به بررسی تاثیر سایز تعبیه در نتیجه حاصل می‌پردازیم. این سایز را روی مقادیر ۳۲، ۶۴، ۱۲۸ و ۲۵۶ بررسی شده است. مدل IGNN با سایز تعبیه برابر با ۶۴ بهترین نتیجه را داشته است و با افزایش این مقدار کارایی مدل کاهش پیدا کرده است. این موضوع شاید به دلیل overfitting مدل در استفاده از ابعاد بسیار بزرگ باشد.



(a) Recall



(b) NDCG

تصویر ۷.۸: تاثیر سایز لایه انتشار

۴.۸ نتیجه گیری

در این مقاله یک مدل معرفی شد که برای انتخاب k خبر برتر مورد استفاده قرار گرفت. این مدل دو چالش را پشت سر می‌گذارد، در اولین مرحله با شبکه عصبی کانولوشن اطلاعات دانش محور را استخراج می‌کند و در مرحله دوم از لایه‌های انتشار تعبیه برای بهره برداری از سیگنال‌های مشارکتی در گراف کاربر آیتم مورد استفاده قرار گرفت. آزمایش‌های گسترده‌ای نیز روی این مدل برای مقایسه با مدل‌های دیگر انجام شد که نشان داده شد مدل معرفی شده برتری قابل توجهی نسبت به مدل‌های دیگر در مورد پیشنهاددهی اخبار دارد.

نتیجه‌گیری

در این قسمت سعی می‌کنیم خلاصه‌ای از مدل‌های معرفی شده ارائه دهیم و مدل‌های مورد نظر را از جهات مختلف بررسی کنیم. متدهای معرفی شده را به اختصار با نام‌های مدل انتشار فعالیت، KGAT ، KTUP ، RippleNet و IGNN بررسی می‌کنیم.

۱.۹ بررسی نهایی روش‌ها

همانطور که مشاهده شد روش انتشار فعالیت مدل نسبتاً ساده‌تری از بقیه مدل‌های معرفی شده برای پیشنهاددهی بود و شباهت بسیار بالایی به فیلتر مشارکتی داشت. این روش مانند فیلتر مشارکتی سعی می‌کند آیتم‌ها و کاربران مشابه را یافته تا براساس آن به کاربران پیشنهاد مناسب ارائه دهد. ولی تفاوت آن با فیلتر مشارکتی در اینجا بود که در این روش تلاش شد که داده‌های ناهمگون در کنار داده‌های ماتریس کاربر-آیتم قرار بگیرد و یک گراف دانش جامع برای همه کاربردها بسازد و سپس با تکنیک پخش اطلاعات روی این گراف میزان شباهت کاربران به هم را سنجیده و از این میزان شباهت به عنوان ضریبی برای به دست آوردن پیشنهاد نهایی استفاده کند. این مدل در شرایط شروع سرد پاسخ نسبتاً مناسبی به سیستم می‌دهد ولی در کل تقریباً به خوبی فیلتر مشارکتی عمل می‌کند. معیارهای استفاده شده برای بررسی کارایی این مدل نسبت به بقیه متدها ساده‌تر است و تنها روش پایه مقایسه شده با این مدل فیلتر مشارکتی است؛ در واقع پیشرفت قابل توجهی در این زمینه نداشته است.

در RKGE تلاش می‌شود که اطلاعات ناهمگون موجود در گراف دانش را استخراج کرده و از آن در کنار اطلاعات ماتریس کاربر-آیتم برای پیشنهاددهی استفاده کند. برای اینکه اطلاعات موجودیت‌های مختلف از گراف دانش استخراج شود تمام مسیرهای بین دو موجودیت بررسی شده و براساس معنی مسیر، مسیر روی بازنمایی نهایی موجودیت تاثیر خواهد گذاشت. برای اینکه ساخت و بررسی همه مسیرها کار ساده‌ای نیست و ممکن است ساختن دستی آن‌ها روی نتیجه نهایی تاثیر منفی داشته باشد یافتن و بررسی همه مسیرها برای به دست آوردن بازنمایی به صورت کاملاً اتوماتیک و به وسیله شبکه‌ی تکرار شونده انجام می‌شود؛ اینگونه که برای هر مسیر یک شبکه‌ی تکرار شونده در نظر گرفته می‌شود که وظیفه به دست آوردن بازنمایی نهایی را دارد و ممکن است چند شبکه‌ی تکرار شونده به صورت همزمان بر روی بازنمایی یک موجودیت کار کنند که این موضوع باعث می‌شود از یادگیری بیش از حد جلوگیری شود. در نهایت از بازنمایی‌های نهایی تولید شده در کنار ماتریس کاربر-آیتم برای پیشنهاد دهی استفاده می‌شود.

در روش RippleNet که برای پیش‌بینی نرخ کلیک کاربران روی آیتم‌ها طراحی شده است برای پیشنهاددهی به کاربر از پخش ترجیحات او روی گراف دانش استفاده می‌شود. در واقع ایده اصلی این روش بسیار شبیه به مدل انتشار فعالیت است با این تفاوت که اطلاعات پخش شده روی گراف به طور مستقیم بر روی پیشنهاددهی تاثیر نمی‌گذارند و مدل از این اطلاعات برای محاسبه تعبیه‌های یک کاربر و آیتم خاص استفاده می‌کند و در نهایت برای محاسبه احتمال علاقه‌مندی کاربر به آیتم از این دو بردار تعبیه استفاده می‌کند. نقطه‌های آغازین برای شروع پخش علاقه‌مندی آیتم‌هایی است که

کاربر در گذشته به آن‌ها علاقه نشان داده است و علایق کاربر با شروع از این نقاط مانند امواجی که روی آب پخش می‌شوند در گراف پیش می‌روند، دلیل انتخاب این نام برای این متد همین موضوع است. پخش اطلاعات در H مرحله انجام شده و در هر مرحله مجموعه‌ای منتسب به این مرحله با نام مجموعه موج ساخته می‌شود که برای محاسبه تعبیه کاربر و آیتم‌ها از آن‌ها استفاده می‌شود.

همانطور که می‌دانیم تقریباً غیرممکن است که اطلاعات یک گراف دانش کامل باشد چون داده‌های موجود معمولاً بسیار پراکنده‌اند و همین موضوع می‌تواند تأثیر مستقیمی روی کیفیت پیشنهاددهی داشته باشد. در روش KTUP سعی شده که مدلی طراحی شود که به طور همزمان هم برای تکمیل گراف دانش و قرار دادن اطلاعات ضمنی در آن مورد استفاده قرار گیرد و هم برای پیشنهاددهی استفاده شود. در این مدل برخلاف مدل‌های دیگر اطلاعات "روابط" به گراف دانش اضافه می‌شود که این موضوع می‌تواند علایق کاربر را بهتر نمایش دهد یا استخراج کند. در نهایت برای پیشنهاددهی یک مدل مبتنی بر ترجمه طراحی می‌کند که با گرفتن لیستی از زوج‌های کاربر-آیتم احتمال علاقه‌مندی هر کاربر به آیتم‌های مورد نظر را خروجی می‌دهد. این مدل مبتنی بر ترجمه طراحی شده مشابه به TransH است ولی مشکل در نمایش روابط $N \times N$ را حل کرده است.

در دو روش قبلی دیدیم که از گراف دانش در کنار اطلاعات ماتریس کاربر-آیتم استفاده شد و اطلاعات گراف دانش به صورت ضمنی در مدل به کار گرفته شد ولی روش KGAT سعی می‌کند اطلاعات گراف دانش را به صورت صریح به ماتریس کاربر-آیتم که در واقع یک گراف دو بخشی است اضافه کند و یک گراف یکپارچه برای عملیات پیشنهاددهی بسازد، این گراف یکپارچه بسیار مشابه گراف جامع ساخته شده در مدل انتشار فعالیت است ولی مدل محاسباتی ارائه شده روی این گراف متفاوت است. در ادامه KGAT با استفاده از مدل TransR تعبیه‌های موجودیت‌های درون گراف یکپارچه را محاسبه می‌کند و سپس در سه مرحله اطلاعات تعبیه‌های مورد نظر را پخش کرده، پاسخ داده و جمع‌آوری می‌کند و در نهایت از این اطلاعات جمع‌آوری شده برای پیشنهاددهی استفاده می‌کند. میزان پخش اطلاعات بین دو موجودیت با میزان شباهت آن‌ها که بر اساس تعبیه‌های مورد نظر محاسبه می‌شود رابطه مستقیم دارد.

روش IGNN برخلاف روش‌های قبلی بررسی شده برای کاربرد خاص‌تری که پیشنهاد اخبار است معرفی شده است. این روش علاوه بر اطلاعات ماتریس کاربر-آیتم و گراف دانش از اطلاعات محتوای اخبار نیز برای پیشنهاددهی استفاده می‌کند. در واقع این روش به دلیل ویژگی‌های خاص پیشنهاددهی اخبار تلاش می‌کند از دو مدل کلی فیلتر مشارکتی و فیلتر مبتنی بر محتوا به صورت همزمان استفاده کند. پیشنهاددهی در این مدل بر مبنای اطلاعات به دست آمده در دو قسمت استخراج اطلاعات محتوایی و استخراج اطلاعات مشارکتی و اتصال این دو مورد به هم انجام می‌شود. اطلاعات محتوایی اخبار بر اساس تیترو آن‌ها و تعبیه کلمات، موجودیت‌ها و محتوای تیترو به وسیله شبکه عصبی کانولوشن به دست می‌آید و اطلاعات مشارکتی بر اساس گراف کاربر-آیتم و تعبیه‌های استخراج شده از گراف دانش استخراج می‌شود. تعبیه‌های استفاده شده در این روش به وسیله مدل TransE محاسبه شده است و استخراج اطلاعات مشارکتی در این مدل همانند مدل KGAT توسط انتشار تعبیه‌ها روی گراف که در سه مرحله انجام می‌شود محاسبه می‌شود.

با توجه به خلاصه‌ی روش‌های بررسی شده می‌توانیم روند رشد استفاده از گراف دانش در سیستم‌های توصیه‌گر را مشاهده کنیم که نشان‌دهنده اهمیت این حوزه تحقیقاتی است. ایده‌های کلی استفاده شده در روش‌های بررسی شده در بعضی موارد شباهت نزدیکی دارند ولی هر کدام با تغییر کوچکی در روند پیشنهاددهی این عملیات را بهبود داده‌اند. به طور مثال ایده استفاده از گرافی که همه اطلاعات را به صورت یکپارچه استفاده کند در دو مدل انتشار فعالیت و KGAT مشابه است و ایده پخش اطلاعات روی گراف که ایده نسبتاً ساده‌ای مثل الگوریتم bfs روی گراف است در چهار مدل بررسی شده انتشار فعالیت، KGAT RippleNet و IGNN استفاده شده است که البته هر کدام از آن‌ها به نحوی متفاوت از پخش اطلاعات مرحله به مرحله روی گراف استفاده کرده‌اند. پخش اطلاعات در دو مدل KGAT و IGNN بسیار مشابه هم هستند ولی تفاوت آن‌ها در گراف مورد استفاده برای پخش

اطلاعات و نحوه استفاده از این اطلاعات پخش شده برای پیشنهاددهی است. تعبیه سازی گراف نیز در اکثر مدل‌های جدید ارائه شده استفاده شده است؛ برای این کار یا از مدل‌های طراحی شده موجود استفاده کرده‌اند مثل دو مدل KGAT و IGNN که به ترتیب از TransE و TransR استفاده کرده‌اند یا یک روش جدید با توجه به نیازهای موجود و ویژگی منحصر به فرد مورد نظر طراحی کرده‌اند مثل سه مدل RKGE ، RippleNet و KTUP که از ایده ارائه یک مدل نمایش جدید برای موجودیت‌ها و رابطه‌ها استفاده کرده‌اند.

۲.۹ بررسی نتایج پیشنهاددهی حاصل از روش‌ها

با توجه به اینکه هر مدلی برای یک کاربرد خاص طراحی شده است و هر کدام بر روی دیتاست و با معیار متفاوتی کارایی متد خود را سنجیده و مقایسه با سایر روش‌ها را انجام داده‌اند امکان مقایسه دقیق این متدها با یکدیگر را با توجه به داده‌های موجود نداریم و متأسفانه زمان لازم برای پیاده سازی و یا بررسی کدهای موجود را هم نداریم.

ولی با توجه به اینکه مقالات جدید هر سال به دلیل پیشرفت و نتیجه بهتر مدل، متد جدیدی را ارائه می‌دهند می‌توانیم انتظار داشته باشیم که خروجی متدهای بررسی شده به تدریجی که در این گزارش مورد بررسی قرار گرفته‌اند بهبود یافته باشند. مثلاً متد IGNN برای کاربرد خاص پیشنهاد اخبار پیشرفت قابل توجهی نسبت به سایر متدهای موجود داشته است و مدل KGAT نیز نتایج پیشنهاددهی بهتری نسبت به سایر مدل‌ها داشته است و به طور خاص با متد RippleNet نیز مقایسه شده است که نتایج بهتری نسبت به آن داشته است.

۳.۹ خلاصه ای از ویژگی‌های متدها

در جدول صفحه بعد می‌توانید خلاصه‌ای از ویژگی‌های متدها که در مقاله آن‌ها استفاده شده است را مشاهده کنید.

Method	Type	Application	Novelty	Graph Embed	Baselines	Datasets
activation spreading	Graph based	General	(1) General Graph, (2) activation spreading in recommendation	-	CF	MovieLen
RKGE	Embedding-based - Path-based	Movie ,Book ,News	(1) Creating new architecture for graph usage, (2) automaticall path discovery by batch of recurrent networks	RKGE	LibFM - CKE - BPRMF - HeteRS - HeteRec - NCF - GraphLF	MovieLens-1M - Yelp
RippleNet	Embedding-based - Path-based	Movie, Book, News	(1) Preference Propagation, (2) automatical possible path discovery	RippleNet	CKE - SHINE - DKN - PER - LibFM - DeepWide	MovieLen - Book- Crossing - Bing-News
KTUP	Embedding-based - content-based	movie ,Book	(1) a new translation-based model, which exploits the implicit preference representations to capture the relations between users and items (2) jointly modeling item recommendation and KG completion to couple the preference representations	KTUP	FM - C/FG - CKE - CoFM	MovieLen - DBbook2014
KGAT	Embedding-based	General	(1) making a new graph with user-item graph and KG with extra information and high-order connectivity, (2) using neural attention mechanism for neighbors weight	TransR	FM - NFM - CKE - C/FG - McRec - Rip- pleNet - CG-MC	Amazon-Book - Last-FM - Yelp2018
IGNN	Embedding-based - content-based	News	(1) Message passing Embedding propagation, (2) Knowledge-Aware Convolutional Neural Network (content based)	TransE	MF - LibFM - Deep- Wide - NeuMF - GC- MC - DKN	DC - Adressa

References

- [1] **Knowledge Graph Embedding: A Survey of Approaches and Applications** Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. *IEEE Transactions on Knowledge and Data Engineering*, 2019.
- [2] **Recommendations on a Knowledge Graph** László Grad-Gyenge, Peter Filzmoser, Hannes Werthner, *MLRec 2015 1st International Workshop on Machine Learning Methods for Recommender Systems*, 2015
- [3] **Recurrent Knowledge Graph Embedding for Effective Recommendation** Zhu Sun, Jie Yang, Jie Zhang, Alessandro Bozzon, Long-Kai Huang, Chi Xu. *acm.org.RecSys '18*, October 2–7, 2018, Vancouver, BC, Canada, 2018
- [4] **RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems** Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, Minyi Guo. *CIKM '18*, October 22–26, 2018, Torino, Italy, 2018.
- [5] **Unifying Knowledge Graph Learning and Recommendation: Towards a Better Understanding of User Preferences** Yixin Cao, Xiang Wang, Xiangnan He, Zikun Hu and Tat-Seng Chua. *IW3C2 (International World Wide Web Conference Committee)*, published under Creative Commons CC-BY 4.0 License, 2019.
- [6] **KGAT: Knowledge Graph Attention Network for Recommendation** Yixin Cao, Xiang Wang, Xiangnan He, Meng Liu and Tat-Seng Chua. *ACM 2019 Association for Computing Machinery*, 2019.
- [7] **Knowledge Graph Embedding: A Survey of Approaches and Applications** Yongye Qian, Pengpeng Zhao, Zhixu Li, Junhua Fang, Lei Zhao, Victor S. Sheng and Zhiming Cui. *Springer Nature Switzerland AG 2019: WISE 2019, LNCS 11881*, pp. 599–614, 2019.