دوره چهاردهم، شماره دوم، پاییز و زمستان ۱۴۰۰

مجله فناوری اطلاعات در طراحی مهندسی

Information Technology in Engineering Design

http://ited.sinaweb.net

ارایه یک مدل پیش‌بینی غیرخطی با حداکثر حاشیه با کمک اسناد توصیفی برای بهبود عملکرد سیستم‌های توصیه‌گر

مهدی رواخواه[۱] مهرداد جلالی*[۲] یحیی فرقانی[۳] رضا شیبانی[۴]

(۱) گروه مهندسی کامپیوتر، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران

(۲) گروه مهندسی کامپیوتر، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران*

(۳) گروه مهندسی کامپیوتر، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران

(۴) گروه مهندسی کامپیوتر، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران

(تاریخ دریافت : ۱۴۰۱/۰۳/۲۶ تاریخ پذیرش: ۱۴۰۱/۰۴/۲۸)

چکیده

بسیاری از مدل‌های پیش‌بینی در سیستم‌های توصیه‌گر، سعی در مدل کردن فاکتورهای پنهان کاربران و آیتم‌ها دارند. این مدل‌ها به کمک ماتریس امتیازات داده شده توسط کاربران به آیتم‌ها آموزش می‌بینند و کم بودن نسبت امتیازات داده شده کاربران به آیتم‌ها نسبت به کل امتیازات ممکن، باعث کاهش دقت این مدل‌ها شده است. لذا برای حل این مشکل، در برخی پژوهش‌ها سعی گردیده است که علاوه بر امتیازات موجود، از اطلاعات کمکی نظیر اسناد توصیفی که در مورد آیتم‌ها وجود دارند استفاده گردد. اما بسیاری از آنها از مدل‌های قدیمی‌تر پرکاربرد در متن کاوی استفاده نموده اند و همچنین عدم لحاظ نمودن حداکثر حاشیه در هنگام محاسبه ویژگی‌های کاربران و آیتم‌ها باعث گردیده است که ویژگی‌های کاربران و آیتم‌ها به شکل موثری استخراج نگردد. در این مقاله و در روش ارایه شده، مدلی غیرخطی ارایه کرده‌ایم که انعطاف بیشتری در مقایسه با مدل‌های خطی رایج دارد و همچنین علاوه بر استفاده مفید از اسناد توصیفی در مورد آیتم‌ها، با لحاظ کردن حداکثر حاشیه در هنگام استخراج ویژگی‌های کاربران باعث بهبود صحت پیش‌بینی در مقایسه با پژوهش‌های قبل گردیده است. با توجه به توانایی شبکه‌های عصبی در کار با دنباله‌ها، برای استخراج ویژگی از اسناد متنی از شبکه عصبی عمیق با حافظه کوتاه مدت در مدل پیشنهادی استفاده می‌نماییم.

واژه‌های کلیدی: فاکتورسازی ماتریس با بیشترین حاشیه، مدل غیرخطی، اسناد توصیفی، یادگیری عمیق.

* عهده‌دار مکاتبات:

مهرداد جلالی

نشانی: گروه **مهندسی کامپیوتر، واحد مشهد، دانشگاه آزاد اسلامی، مشهد، ایران**

تلفن: پست الکترونیکی: *mehradad.jalali@kit.edu*

۱. مقدمه

حجم داده‌های تولید شده توسط ابزارهای الکترونیکی و خودکار، با سرعت قابل توجهی در حال افزایش هستند. اما بهره بردن از این داده‌های حجیم در صورتی امکان‌پذیر است که ابتدا روی آن‌ها پردازش و تحلیل صورت پذیرد و سپس از نتایج و خروجی‌های این پردازش‌ها استفاده گردد. در این راه، سیستم توصیه‌گر به کاربران این امکان را می‌دهد که در بین انبوه آیتم‌های موجود، با توجه به پیشنهادهای ارایه شده توسط این سیستم، به سادگی بهترین انتخاب‌ها را انجام داده و تجربه نمایند. همچنین استفاده از این نوع سیستم‌ها باعث افزایش فروش محصولات و خدمات برای ارایه دهندگان آن‌ها می‌شود که سود هنگفتی را نصیب آنان می‌نماید. به‌طوری‌که در حال حاضر حجم قابل توجهی از فروش‌های اینترنتی سایت‌های فروش را محصولات پیشنهاد شده توسط سیستم‌های توصیه‌گر به خود اختصاص می‌دهند. امروزه اکثر فروشگاه‌های اینترنتی، وب سایت‌های ارایه خدمات اینترنتی، شبکه‌های اجتماعی و ... به عنوان جزئی از خود شامل سیستم توصیه‌گر می‌باشند که سعی در پیش‌بینی ترجیحات و سلایق کاربر و بهبود ارایه خدمات دارند. علیرغم اینکه داده‌های زیادی در سیستم‌های فعلی ذخیره می‌گردند اما با توجه به تعداد زیاد کاربران و همچنین تعداد زیاد آیتم‌ها عملا امتیازاتی که کاربران به آیتم‌ها می‌دهند درصد بسیار کوچکی از امتیازات ممکن است. می‌توان از اسناد توصیفی مانند نظرات کاربران در مورد آیتم‌ها یا خلاصه‌هایی در مورد آیتم‌ها برای مقابله با مشکل کمبود اطلاعات در مورد سلیقه کاربران در خصوص آیتم‌های مختلف که به آن مشکل خلوت بودن نیز گفته می‌شود، در جهت بهبود کیفیت پیش‌بینی امتیازات کاربران به آیتم‌ها بهره برد. این اسناد متنی در بسیاری از سیستم‌ها ذخیره می‌شوند و به راحتی در دسترس قرار دارند، همچنین با اصول مربوط به حفظ محرمانگی کاربران نیز در تضاد قرار ندارند. اگرچه اطلاعات نوشتاری کاربران به‌صورت غیرساختارمند هستند اما پیشرفت‌هایی که در حوزه مدلسازی معنایی و کاوش متن و یادگیری ماشین رخ داده است امکان تفسیر اسناد متنی و استخراج اطلاعات مفید از آن را فراهم نموده است. از روی اسناد متنی مربوط به آیتم‌ها می‌توان اطلاعات مختلفی استخراج نمود که در طراحی سیستم‌های توصیه‌گر بسیار مفید و پرکاربرد می‌باشند. مشخص‌ترین راه برای تحلیل متون مانند [1,2] شناسایی عبارات پرتکرار و استفاده از روش‌های اندازه‌گیری وزنی مانند TF-IDF برای تعیین اهمیت هر عبارت است. برخی از کاربران در نظراتشان موضوعات و جنبه‌هایی را مطرح می‌کنند. به‌عنوان مثال، در متنی که در مورد یک هتل در TripAdvisor نوشته شده است، موضوعاتی مانند کیفیت اتاق، غذا، وسایل ورزشی، مکان و ... بیان می‌شوند. برای بازنمایی سند با یک توزیع موضوعی K-بعدی و توزیع کلمات هر موضوع در این‌گونه رهیافت‌ها عموما از تکنیک‌هایی مانند تخصیص پنهان دیریکله (LDA) [3] استفاده می‌شود و مانند [4,5] موضوعات کشف شده می‌توانند برای بهبود امتیازدهی در رهیافت‌های پالایش مشارکتی مورد استفاده قرار گیرند. همچنین یک راه ساده برای تخمین نظر کلی کاربر، تجمیع احساسات همه کلمات احساسی اوست که در نظرش آمده است[6,7]. به‌عنوان جایگزین، یک الگوریتم یادگیری ماشین (مانند دسته بند بیزین Naive یا SVM) می‌تواند برای یادگیری عقیده و دسته بندی آن به درون یک رده احساسی مناسب مورد استفاده قرار گیرد [8,9]. مقایسه‌های انجام شده در نظرات کاربران را نیز می‌توان با استفاده از یک سری قوانین زبانی خاص استخراج کرد[10]. حالات روحی کاربر مانند ناراحتی، لذت بردن، پریشانی، شاد بودن و ... را می‌توان با ساخت یک دسته‌بند برای برچسب‌گذاری متون با حالات روحی مختلف استخراج نمود[11]. از تعداد آراء داده شده سایر کاربران به یک نظر می‌توان برای تعیین کیفیت امتیازدهی آن کاربر به آیتم مورد نظر استفاده نمود [12]. اما مهمترین و مفیدترین نوع استفاده از نظرات و اسناد متنی مرتبط با کاربران و آیتم‌ها در سیستم‌های توصیه‌گر مدل کردن فاکتورهای پنهان موجود در آن‌ها می‌باشد. با استفاده از این فاکتورها می‌توان ارتباط بین ذائقه کاربران و ویژگی‌های آیتم‌ها را بدست آورده و از آن در ارائه پیشنهادها استفاده نمود. با پیشرفت غیرمنتظره کاربرد تکنولوژی یادگیری

عمیق در زمینه پردازش زبان طبیعی و تصـویر و ویدیو، بکاربردن این تکنولوژی برای سـاختن یا بهبود سیستم‌های توصیه‌گر تبدیل به یک موضـوع داغ شـده اسـت. با این وجود مسائل باز بسیاری در این مسیر وجود دارد، مانند اینکه شبکه عصبی را چگونه بسـازیم تا برای اطلاعات متنی کاربرد داشـته باشـد یا اینکه چگونه ماژول شبکه عصبی را با فاکتورسازی ماتریس که جزء مهمترین روش‌های کشـف فاکتورهای پنهان کاربران و آیتم‌ها است ترکیب کنیم تا مدلی مناسب برای پیش‌بینی نظر یک کاربر در مورد یک آیتم طراحی کنیم. در این مقاله یک مدل غیرخطی ارایه نموده‌ایم که از اسناد متنی که در مورد آیتم‌ها است برای اسـتخراج بهتر ویژگی‌های آیتم‌ها اسـتفاده می‌نماید. همچنین با در نظر گرفتن حداکثر حاشـیه در استخراج ویژگی‌های کاربران و آیتم‌ها کارایی مدل در پیش‌بینی امتیاز برای آیتم‌هایی که توسط کاربران امتیازدهی نشده‌اند را حاصل می‌کند و باعث بهبود صحت می‌گردد. در ادامه و در بخش ۲، پژوهش‌های مرتبط با سیستم‌های توصیه‌گری که از اسناد متنی استفاده می‌نمایند بررسی گردیده‌اند. بخش ۳، مقدماتی را بیان می‌کند که برای ارایه روش پیشنهادی مورد نیاز است. در بخش ۴ روش پیشنهادی جدید را ارایه نموده‌ایم. بخش ۵ شامل نتایج عملی و بخش ۶ نتیجه‌گیری و کارهای آینده می‌باشد.

۲. پژوهش‌های مرتبط

در این بخش به بررسی سیستم‌های توصیه‌گر مبتنی بر اسناد متنی مرتبط با آیتم‌ها می‌پردازیم. در این روش‌ها، نویسندگان راهکارهایی برای کشف موضوعات آیتم‌ها یا به‌دست آوردن فاکتورهای پنهان مرتبط با آیتم‌ها و کاربران ارایه نموده اند و از آن برای پیش‌بینی امتیاز اعطایی کاربر به آیتم استفاده نموده اند. مقاله [13]، یک مدل بیزین سلسله مراتبی است که از SDAE [۱] و PMF [۲] استفاده می‌کند و نوع بهبود یافته CTR [18] محسوب می‌شود. ساختار عصبی عمیق باعث یادگیری ویژگی‌های پنهان قابل تفسیر از روی متون می‌شود. اشکال این روش نیز محدودیت‌های ذاتی روش‌های کیسه کلمات است. روش ارایه شده در [14] به‌عنوان ورودی علاوه بر امتیازات، تمامی نوشته‌های یک کاربر و تمامی نوشته‌های در مورد یک آیتم را دریافت می‌کند. یادگیری ویژگی‌های نهان کاربران و آیتم‌ها از روی متون به‌وسیله شبکه عصبی کانولوشن و مکانیزم توجه انجام می‌پذیرد. از سوی دیگر یادگیری مبتنی بر برهم‌کنش‌های بین کاربران و آیتم‌ها از طریق ماشین تجزیه صورت می‌گیرد. در نهایت با ترکیب کردن پویای امتیاز پیش‌بینی شده از روی ویژگی‌های بدست آمده متون و امتیاز پیش‌بینی شده از روی ویژگی‌های بدست آمده از ماشین تجزیه، امتیاز نهایی تخمین زده می‌شود. اشکال این روش نیز این است که زمانی مفید است که برای کاربر و آیتم مورد نظر سند متنی وجود داشته باشد. مقاله [15] روشی ارایه کرده است که از LDA و NMF استفاده کرده است اما اشکال مشخصش استفاده از نمایش *bow* (کیسه کلمات) برای پردازش متون است که اطلاعات زمینه ای معنایی که در کلمات محلی کد شده اند را لحاظ نکرده است. از این‌رو اطلاعات بهم چسبیده زیادی که به شکل عبارات و جملات هستند در این شیوه پردازش متنی درشت دانه گم می‌شوند. مقاله [16] علاوه بر اینکه اشکالات مقاله قبل را دارد این اشکال را هم دارد که موضوعات پنهان فقط می‌توانند به فاکتورهای پنهان کاربر (یا آیتم) مرتبط باشند در حالیکه در مقاله قبل به‌صورت همزمان به هردو وابسته شده اند. مقاله [17] همان اشکالات مقاله [15] را دارد. در این مقاله روشی سه مرحله‌ای ارایه شده است. مرحله اول یادگیری میزان توصیه‌پذیری آیتم‌ها است که در آن به‌صورت تقویتی ویژگی‌هایی که در امتیازات بالاتر کشف شده اند قابلیت توصیه بالاتری دارند. در این مرحله نیز از LDA استفاده شده است. در مرحله دوم با کمک موضوعات استخراج

[۱] Stacked Denoising Autoencoder

[۲] Probabilistic Matrix Factorization

شده آیتم‌ها در مرحله قبل و همچنین امتیازی که کاربر به آن آیتم داده است میزان علاقمندی کاربر به ویژگی‌های مختلف بدست می‌آید. در مرحله آخر نیز با ارایه رابطه‌ای که مبتنی بر مدل فاکتورهای پنهان است و با استفاده از ویژگی‌های کشف شده کاربران و آیتم‌ها در دو مرحله قبل امتیاز نهایی پیش‌بینی می‌شود. برای بدست آوردن فاکتورهای پنهان کاربر و آیتم و بایاس‌ها در این مرحله از نزول گرادیان تصادفی جهت بهینه‌سازی تابع هدف مربوطه استفاده شده است. [18] پالایش مشارکتی سنتی را با مدلسازی موضوعی احتمالاتی ترکیب می‌کند و ساختار پنهان کاربران و آیتم‌ها را بدست می‌آورد. اشکال این روش هم مانند برخی دیگر از روش‌ها، استفاده از LDA می‌باشد که از کیسه کلمات استفاده کرده و اطلاعات زمینه‌ای معنایی که در کلمات محلی کد شده‌اند را لحاظ نمی‌کند. در [19] از شبکه عصبی CNN برای استخراج ویژگی‌های پنهان کاربر یا آیتم استفاده کرده که با استفاده از بازنمایی چگال تعبیه کلمات و یک پنجره محلی برای بدست آوردن اطلاعات مفهومی، تکنیک‌های مبتنی بر CNN درک معنایی بهتری از نظرات دارند که منجر به بهبود قابل توجهی در پیش‌بینی امتیازات نسبت به مدل‌های bow می‌شود. با این وجود درک ویژگی‌های استخراج شده توسط شبکه‌های عصبی برای انسان مشکل است و بنابراین قابلیت تفسیر سیستم‌های توصیه‌گر را محدود می‌کند. این مقاله DeepCoNN ارایه شده در [21] را با افزودن یک لایه اضافی (شبکه هدف) برای یادگیری بازنمایی نظر یک کاربر هدف – یک آیتم هدف در زمان آموزش توسعه داده است و سپس از بازنمایی یادگرفته شده برای تنظیم خروجی شبکه منبع استفاده می‌کند. بنابراین شبکه منبع می‌تواند بازنمایی پنهان یک متن هدف را تقلید کند که در زمان تست در دسترس نیست. این روش‌ها بهتر از مدل‌های bow عمل می‌کنند و در حذف مشکلات شروع سرد و تنک بودن داده موثرند اما تنها ویژگی‌های کاربران و آیتم‌ها را به‌صورت استاتیک و جداگانه بدست می‌آورند که از اثرات متقابل گوناگون و پیچیده بین کاربران و آیتم‌ها غافل می‌شود. مقاله [20] روشی به نام CONVMF پیشنهاد کرده است که یک مدل شبکه عصبی مبتنی بر CNN ارایه کرده، که از CNN برای فراهم کردن بهتر بازنمایی معنایی پنهان متون به‌وسیله لحاظ کردن ترتیب کلمات و اطلاعات زمینه‌ای استفاده می‌کند. خروجی شبکه عصبی CNN ضمن ترکیب شدن با تجزیه ماتریس احتمالاتی برای بدست آوردن بردارهای پنهان آیتم‌ها مورد استفاده قرار می‌گیرد. مقاله [21] ، روشی به نام DeepCoNN ارایه کرده است که از یک مدل CNN موازی برای بدست آوردن جداگانه ویژگی‌های پنهان کاربران و آیتم‌ها بر اساس متون‌شان استفاده می‌کند. سپس ویژگی‌های پنهان کاربر و آیتم متناظر را بهم الحاق کرده و به یک ماشین تجزیه (FM) برای پیش‌بینی امتیاز ارسال می‌کند. مقاله [22] برای حل مشکل سه مقاله بالا ابتدا یک مکانیزم توجه را دخیل می‌کند تا یک مدل قابل تفسیرتر برای پیش‌بینی امتیازات تولید کند. این مقاله توجه محلی و سراسری روی متون را ترکیب می‌کند و متون وزندار تولید می‌کند که به یک مدل CNN داده می‌شوند تا بازنمایی بهتری از کاربران و آیتم‌ها بدست آید. مقاله [23] از روشی شبیه به روش‌های LDA استفاده کرده اما به جای فاکتورسازی ماتریس از مدل‌های مخلوط گوسی برای مدل کردن امتیازات استفاده کرده است. مقاله [24] از یک مدل مبتنی بر تعبیه برای یکپارچه کردن مدل تعبیه کلمات با یک مدل استاندارد فاکتورسازی ماتریس استفاده می‌کند تا اطلاعات زمینه ای کلمات در متون را فراهم کند. در این مقاله از مدل CBOW و skip-gram استفاده شده است. تفاوت این دو مدل این است که CBOW از کلمات زمینه ای محلی (کلمات قبل و بعد از یک کلمه) برای پیش‌بینی کلمه جاری استفاده می‌کنند در حالیکه در مدل skip-gram کلمات اطراف یک کلمه بر اساس آن کلمه پیش‌بینی می‌شوند. مقاله[25] روشی با نام R-ConvMF ارایه کرده است که نوع بهبود یافته روش ConvMF است که متعلق به همان نویسندگان است. تفاوت اصلی که روش جدید دارد این است که برای مدل کردن فاکتورهای نهان آیتم‌ها بر اساس تعداد امتیازاتی که به هر آیتم داده شده است، نویز گوسی (خطا) در نظر می‌گیرد. بدین شکل آیتم‌هایی که تعداد امتیازات بیشتری به آن‌ها داده شده است در هنگام بدست آوردن فاکتورهای نهانشان دارای نویز گوسی (خطا) کمتری می‌باشند. مقاله [26] یک مدل یادگیری عمیق مبتنی بر توجه برای محاسبه میزان مفید بودن متون برای توصیه پیشنهاد می‌کند. این روش می‌تواند یک توضیح سطح متنی برای

پیش‌بینی امتیازات فراهم کند. مقاله [27] از یک شبکه GRU مبتنی بر توجه برای تفسیر مفاهیم معنایی مرتبط با هر ویژگی پنهان بدست آمده به‌وسیله مدل فاکتورسازی ماتریس استفاده می‌کند. در [28] از دو شبکه عصبی جداگانه برای کاربران و آیتم‌ها استفاده شده است و مدل سلسله مراتبی ارایه شده است که از GRU دو طرفه و مکانیزم توجه سطح کلمه و سطح جمله برای استخراج اطلاعات کاربران و آیتم‌ها استفاده می‌کند. ازSkipGram برای بازنمایی کلمات استفاده شده است. بعد از بازنمایی کلمات، از GRU دوطرفه به همراه مکانیزم توجه در سطح کلمات برای بازنمایی جملات استفاده شده است و پس از آن نیز از یک GRU دو طرفه به همراه مکانیزم توجه در سطح جملات استفاده شده است که بازنمایی سند را حاصل می‌کند. خروجی بازنمایی سند مرتبط با کاربر و آیتم به لایه FC وارد می‌شوند که خروجی این لایه بردار معنایی مبتنی بر توجه کاربر و بردار معنایی مبتنی بر توجه آیتم را حاصل می‌کند. سپس از ضرب داخلی این بردارها برای تخمین امتیاز داده شده کاربر به آیتم استفاده می‌شود. از RMSE به‌عنوان تابع هزینه برای کمینه کردن تفاوت میزان امتیاز واقعی داده شده با امتیاز تخمین زده شده در زمان آموزش استفاده می‌شود. اشکال این روش این است که زمانی مفید است که برای کاربر و آیتم مورد نظر سند متنی وجود داشته باشد ضمن اینکه زمان مورد نیاز برای آموزش وزن‌های شبکه GRU دو طرفه زیاد است. مقاله [29] از ترکیب کردن RNNهای سلسله مراتبی و PMF برای استفاده از محتواهای متنی و اطلاعات امتیازات جهت طراحی یک سیستم جدید استفاده می‌کند. RNNهای سلسله مراتبی در دو سطح طراحی شده‌اند. در سطح اول با دریافت کلمات یک جمله، مفاهیم جمله را بدست می‌آورند و در سطح دوم با ترکیب کردن مفاهیم بدست آمده از سطح قبل، مفاهیم مرتبط با کل سند را بدست می‌آورند. در نهایت برای ترکیب کردن HRN و PMF جهت استفاده از این مفاهیم بدست آمده و امتیازات داده شده توسط کاربران، از تخمین حداکثر احتمال پسین استفاده شده است. مقاله [30] از سه نوع اطلاعات شامل اطلاعات متنی، اطلاعات جغرافیایی و اطلاعات اجتماعی کاربران برای ساخت مدلی جهت ارایه پیشنهادها نقاط جذاب برای کاربران استفاده می‌کند. برای استخراج ویژگی از اطلاعات متنی از شبکه عصبی کانولوشن استفاده شده است. از نزول گرادیان برای بدست آوردن پارامترها و کمینه کردن تابع هدف بیان شده، استفاده می‌شود. در مقاله [31] از دو شبکه عصبی جداگانه برای کاربران و آیتم‌ها استفاده شده است. ساختار شبکه‌های عصبی شامل لایه بازنمایی کلمات، لایه‌های CNN، لایه‌های GRU رو به جلو و رو به عقب، و لایه خروجی می‌باشد. سپس با ترکیب این شبکه با تجزیه ماتریس احتمالاتی (PMF) و ارایه یک تابع هدف، مقادیر بردارهای کاربران و آیتم‌ها و پارامترهای شبکه با کمک دیتاست آموزشی محاسبه می‌شوند. در نهایت با کمک ویژگی‌های کاربران و آیتم‌ها می‌توان مقادیر امتیازات را پیش‌بینی کرد. اشکال این روش نیز این است که زمانی مفید است که برای کاربر و آیتم مورد نظر سند متنی وجود داشته باشد. تعداد زیاد پارامترهای شبکه نیز زمان آموزش را بالا خواهد برد. مقاله[32] از RNNهای سلسله مراتبی و متقارن برای کشف مفاهیم معنایی و تغییرات در سلایق کاربران یا ویژگی‌های آیتم‌ها استفاده می‌کند. روش ارایه شده در این مقاله به‌عنوان ورودی توالی‌های محتوایی مربوط به کاربران و آیتم‌ها را به‌صورت جداگانه دریافت می‌کند و با کمک شبکه‌های عصبی RNN که در دو سطح قرار گرفته اند و با بهره‌گیری از واحدهای LSTM اقدام به استخراج مفاهیم معنایی از روی محتواهای متنی می‌نماید و از آن‌ها برای پیش‌بینی امتیازات مشخص نشده استفاده می‌کند. در [37] نویسندگان از اطلاعات امتیازات قبلی داده شده، دموگرافی کاربران، اسناد متنی، زمان مشاهده ویدیوها و ... استفاده می‌کنند و با بکاربردن شبکه عصبی سعی در بدست آوردن ویژگی‌های کاربران و آیتم‌ها و در نتیجه بهبود پیش‌بینی نظر کاربران در مورد آیتم‌ها دارند. اما این روش نیز در زمان بدست آوردن ویژگی‌های کاربران و آیتم‌ها حداکثر حاشیه را در نظر نمی‌گیرند. به‌طور خلاصه می‌توان گفت کارهای انجام شده قبل، شامل ایرادات مختلفی هستند مانند : محدودیت‌های روش‌های مبتنی بر کیسه کلمات، عدم لحاظ کردن حداکثر حاشیه در هنگام بدست آوردن ویژگی‌های پنهان، داشتن یک مدل پیشگوی خطی، زمان‌بر بودن زمان آموزش، نیاز به وجود اسناد متنی هم در مورد کاربران و هم در مورد آیتم‌ها و ... و در پژوهش فعلی سعی گردیده است این اشکالات تا حد امکان

مرتفع گردد. لذا در ادامه مدلی غیرخطی ارایه می‌کنیم که انعطاف بیشتری در مقایسه با مدل‌های خطی رایج دارد و در آن بدست آوردن ویژگی‌های پنهان به‌صورت موثرتری انجام می‌پذیرد.

۳. مقدمات

از آنجاکه روش ارایه شده از فاکتورسازی ماتریس با بیشترین حاشیه (Maximum Margin Matrix Factorization) استفاده می‌نماید، لذا به‌صورت مختصر این تکنیک را نیز بررسی می‌نماییم. فرض کنید عناصر ماتریس امتیازدهی Y دو سطح ۱+ (برای بیان علاقه داشتن کاربر به آیتم) و ۱- (برای بیان علاقه نداشتن کاربر به آیتم) را شامل شوند و صفر به معنای مشخص نبودن این مقدار باشد. هدف یافتن ماتریس‌های U و V با کمترین میزان نُرم اثر $X = UV^T$ است که Y را با توجه به مجموعه عناصر Ω (عناصری از ماتریس Y که مقدار آن‌ها مشخص است) تقریب بزنند. زمانی‌که $Y \in \{\pm 1\}$ است، در نظر گرفتن حاشیه سخت ۱، معادل این است که برای همه $(i,j) \in \Omega$ داریم $y_{ij}x_{ij} \geq 1$. با معرفی متغیرهای شل‌کننده ξ_{ij} ، محدودیت سخت می‌تواند به‌صورت $y_{ij}x_{ij} \geq 1 - \xi_{ij}$ درآید. کمینه کردن متغیرهای شل‌کننده معادل کمینه کردن تابع خطای محوری (hinge-loss) یعنی $h(z) = (1-z)_+ = max(0,1-z)$ است. خطای محوری $h(z)$ در $z=1$ مشتق پذیر نیست و نسبت به داده‌های پرت بسیار حساس است و بنابراین به‌عنوان جایگزین، خطای محوری نَرم (smooth hinge-loss) در [33] پیشنهاد شد که به‌صورت زیر تعریف می‌شود:

$$h(z) = \begin{cases} 0 & if\ z \geq 1 \\ \frac{1}{2}(1-z)^2 & if\ 0 < z < 1 \\ \frac{1}{2} - z & otherwise \end{cases} \quad (۱)$$

بنابراین می‌توان مسأله فاکتورسازی ماتریس با بیشترین حاشیه را برای ماتریس‌های امتیازدهی دو سطحی به مسأله بهینه‌سازی زیر تبدیل کرد:

$$\min_{U,V} J(U,V) = \sum_{(i,j)\in\Omega} h(y_{ij}(U_i V_j^T)) + \frac{\lambda}{2}(\|U\|_F^2 + \|V\|_F^2) \quad (۲)$$

که در آن $h(.)$ تابع خطای محوری نَرم می‌باشد.

این مسأله بهینه‌سازی می‌تواند توسط حل‌کننده‌های نیمه معین حل گردد. اما حل‌کننده‌های فعلی فقط می‌توانند ابعاد تا چند صدتایی را مدیریت کنند. بنابراین بهینه‌سازی مستقیما مبتنی بر گرادیان در [34] مورد استفاده قرار گرفت.

الگوریتم نزول گرادیان و انواع آن مانند نزول گرادیان مزدوج و نزول گرادیان تصادفی با مقادیر تصادفی U و V آغاز می‌گردد و به‌طور پی در پی U و V به‌صورت زیر بروزرسانی می‌شوند.

فرآیند تکمیل ماتریس از روی ماتریس‌های فاکتور U و V به‌وسیله قاعده زیر انجام می‌پذیرد.

$$\hat{y}_{ij} = \begin{cases} -1 & if\ (i,j) \notin \Omega \wedge U_i V_j^T \leq 0 \\ +1 & if\ (i,j) \notin \Omega \wedge U_i V_j^T \geq 0 \\ y_{ij} & if\ (i,j) \in \Omega \end{cases} \quad (۳)$$

فاکتورسازی ماتریس با بیشترین حاشیه چند سطحی

روش MMMF (فاکتورسازی ماتریس با بیشترین حاشیه) در [35] و پس از آن Fast MMMF (فاکتورسازی ماتریس با بیشترین حاشیه سریع) در [34] از ابتدا برای پالایش مشارکتی با ماتریس‌های امتیازدهی ترتیبی طراحی گردیدند که در آن علاقه‌مندی‌های کاربران به شکل اعداد گسسته بود، نه به‌صورت دوست داشتن یا نداشتن. ایده اصلی آن‌ها توسعه تابع خطای محوری به تنظیمات ترتیبی است. جهت مرتبط کردن مقدار پیش‌بینی شده $x_{ij} = U_i V_j^T$ با مقدار گسسته y_{ij} با R برچسب، R-1 عدد به‌عنوان آستانه استفاده می‌گردد که به آن‌ها $\{\theta_1, \dots, \theta_{R-1}\}$ می‌گویند. برای حالت‌های با حاشیه نَرم، زمانی‌که متغیر شل‌کننده افزوده می‌گردد، تابع خطای تغییریافته نه تنها تخطی از دو محدودیت ضروری $\theta_{y_{ij}-1} + 1 \le x_{ij}$ و $\theta_{y_{ij}} - 1 \ge x_{ij}$ را اندازه می‌گیرد، بلکه تخطی از همه محدودیت‌های آستانه $x_{ij} \ge \theta_r + 1$ برای $r < y_{ij}$ و $x_{ij} \le \theta_r - 1$ برای $r \ge y_{ij}$ نیز اندازه گیری می‌شود. این تغییر سعی می‌کند نقاط امتیازدهی را با حداکثر حاشیه جدا نماید. هدف از روش Fast MMMF کمینه کردن تابع هدف زیر است:

$$J(U,V,\theta) = \sum_{(i,j)\in\Omega} \sum_{r=1}^{R-1} h(T_{ij}^r(\theta_{ir} - U_i V_j^T)) + \frac{\lambda}{2}(\|U\|_F^2 + \|V\|_F^2) \qquad (۴)$$

که در آن $\|.\|_F$ نُرم فروبنیوس، و λ پارامتر تنظیم، و Ω مجموعه عناصر مشاهده شده و $h(z)$ تابع خطای محوری است که قبلا آن را تعریف کردیم، و θ_{ir} آستانه برای امتیاز r کاربر i اُم، و T به‌صورت زیر تعریف می‌گردد:

$$T_{ij}^r = \begin{cases} +1, & if\ r \ge y_{ij} \\ -1, & if\ r < y_{ij} \end{cases} \qquad (۵)$$

گرادیان متغیرهایی که باید بهینه گردند به‌صورت زیر تعیین می‌گردند.

$$\frac{\partial J}{\partial U_{ip}} = \lambda U_{ip} - \sum_{r=1}^{R-1} \sum_{j|(i,j)\in\Omega} T_{ij}^r . h'\left(T_{ij}^r(\theta_{ir} - U_i V_j^T)\right).V_{jp} \qquad (۶)$$

$$\frac{\partial J}{\partial V_{jq}} = \lambda V_{jq} - \sum_{r=1}^{R-1} \sum_{i|(i,j)\in\Omega} T_{ij}^r . h'\left(T_{ij}^r(\theta_{ir} - U_i V_j^T)\right).U_{iq} \qquad (۷)$$

$$\frac{\partial J}{\partial \theta_{ir}} = \lambda V_{jq} - \sum_{j|(i,j)\in\Omega} T_{ij}^r . h'\left(T_{ij}^r(\theta_{ir} - U_i V_j^T)\right) \qquad (۸)$$

در Fast MMMF هر Vi یک نقطه و هر جفت (U_j, θ_{jr}) یک ابرصفحه جداکننده نقاط با امتیاز r مربوط به کاربر j اُم می‌باشد. مثلا برای R=5 ، تعداد ۴ ابرصفحه موازی داریم که کل فضا را به ۵ ناحیه (مطابق با ۵ امتیاز) تقسیم می‌نمایند. هدف MMMF یادگیری ویژگی‌های آیتم‌ها به‌عنوان نقاط، کاربران به‌عنوان ابرصفحه‌ها و امتیازات به‌عنوان حد آستانه‌هاست به‌طوری‌که نقاط تا حد امکان به‌صورت صحیحی در نواحی امتیازدهی که توسط ابرصفحه‌های (U_j, θ_{jr}) تعریف شده است قرار گیرند. از آن‌جا که تابع خطای محوری توسعه یافته استفاده شده است، انتظار می‌رود ابرصفحه‌های جداکننده، حاشیه‌ها را بیشینه کنند. اگرچه Fast MMMF می‌تواند تا میلیون‌ها کاربر و آیتم مقیاس پذیر باشد اما هنوز مشکلاتی دارد. این روش نیاز به تعداد زیاد فاکتور پنهان (k) دارد که به‌طور غیرمستقیم منجر به سربار حافظه و زمان اجرا می‌شود. برای هر کاربر، نواحی R امتیاز به‌وسیله یک مجموعه از ابرصفحه‌های موازی جداسازی می‌شود. موازی قرار دادن ابرصفحه‌های تصمیم ساز

ممکن است منجر به خطای تعمیم بالاتری شود. ابرصفحه‌ها کلاس‌ها را با حاشیه‌های نامساوی به همراه بایاسی برای کلاس‌های با نمونه‌های کم جدا می‌کنند. در HMF در [36] برای نشان دادن بایاس MMMF چند سطحی به سمت نمونه‌های با تعداد امتیازات کمتر، آزمایشی انجام شده است. در این آزمایش یک ماتریس با ابعاد ۱۰۰۰*۵ که ۲۰ درصد آن پر است و $k=2$ (تعداد ویژگی‌ها) در نظر گرفته شد. توزیع امتیازات نامتوازن انجام گرفت به‌طوری‌که امتیازات ۱ و ۵ بیشترین تعداد نمونه را داشتند و تعداد نمونه‌های مربوط به امتیاز ۲ و ۴ نیز از تعداد نمونه‌های مربوط به امتیاز ۳ بیشتر در نظر گرفته شد. مشاهده گردید که خطوط موازی جداکننده که بوسیله Ui و آستانه θ برای امتیازات مختلف تعریف شده‌اند نتوانسته‌اند حاشیه بین این کلاس‌ها را حداکثر کنند. برای حل این مشکل، روشی با نام HMF ارایه گردید.

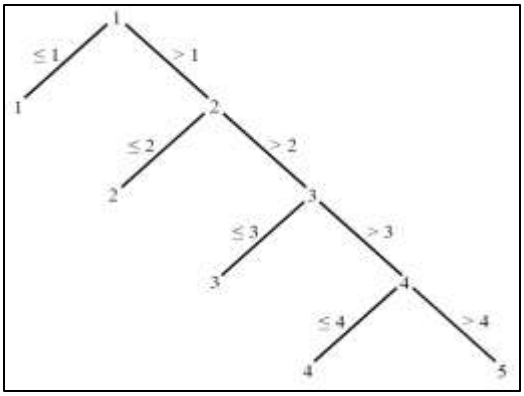
HMF شامل *R-1* گام می‌باشد. برای هر $q \in \{1,2,\dots,R-1\}$ این روش یک ماتریس امتیازدهی مبتنی بر حد آستانه به‌صورت زیر می‌سازد:

$$Y_{ij}^{q} = \begin{cases} -1 & Y_{ij} = q, \\ +1 & Y_{ij} > q, \end{cases} \qquad (۹)$$

که برای یادگیری فاکتورهای پنهان کاربران و آیتم‌ها یعنی U_i^q و V_j^q با استفاده از MMMF دو کلاسه زیر استفاده می‌شود:

$$\min_{U^q,V^q} J(U^q,V^q) = \sum_{Y_{ij} \geq q} h\left(Y_{ij}^{q}\left(U_i^q V_j^{qT}\right)\right) + \frac{\lambda}{2}(\|U^q\|_F^2 + \|V^q\|_F^2). \qquad (۱۰)$$

کل فرآیند HMF را می‌توان به شکل درختی با عمق *R-1* تصویر کرد. هر گره از این درخت با استفاده از یک MMMF دو کلاسه برای Y^q ساخته می‌شود. در مرحله تست نیز این MMMFهای دو کلاسه موجود در درخت مورد استفاده قرار می‌گیرند به این شکل که از ریشه به سمت برگ حرکت انجام می‌شود تا بتوان امتیاز تخمینی کاربر i به آیتم j مشخص کرد. در سطح q درخت اگر $U_i^q V_j^{qT} \leq 0$ امتیاز تخمینی کاربر i به آیتم j برابر q تخمین‌زده می‌شود در غیر اینصورت باید به سطح بعدی درخت رفت. شکل ۱ فرآیند کلی روش HMF را توصیف می‌کند.



شکل۱ : فرآیند کلی روش HMF.

۴. روش پیشنهادی

در بخش مرور ادبیات به بررسی برخی از روش‌های ارایه شده در حوزه سیستم‌های توصیه‌گر پرداختیم و گفته شد تنک بودن ماتریس کاربر-آیتم، صحت روش‌های پالایش مشارکتی را به شدت تحت تاثیر قرار می‌دهد و لذا برخی از روش‌ها سعی در

استفاده از اطلاعات متنی در مورد آیتم‌ها دارند تا بدین‌وسیله بتوانند با ترکیب این اطلاعات اضافی با روش‌های سنتی مانند پالایش مشارکتی اقدام به پیش‌بینی نظر کاربر در مورد آیتم نمایند. اما اشکال برخی از این مقالات، استفاده از LDA یا روش‌های دیگری است که از کیسه کلمات برای پردازش متون استفاده می‌نمایند که اطلاعات زمینه ای معنایی مانند ترتیب کلمات را لحاظ نمی‌کنند. همچنین مدل پیش‌بینی امتیاز کاربر به آیتم که در برخی از مقالات مانند روش‌های مبتنی بر فاکتورسازی ماتریس ارایه شده اند نسبت به فاکتورهای پنهان کاربر یا آیتم خطی هستند که صحت این روش‌ها را محدود می‌کند. ضمنا روش‌های مبتنی بر فاکتورسازی ماتریس با مرتبه پایین در هنگام استخراج ویژگی کاربران حداکثر حاشیه را لحاظ نمی‌نمایند. هدف از ارایه این روش بهبود صحت پیش‌بینی امتیاز کاربر به آیتم از طریق ارایه یک مدل غیرخطی و در نظر گرفتن حداکثر حاشیه در استخراج ویژگی‌های کاربران است. غیرخطی کردن مدل باعث انعطاف پذیری بالاتر مدل و بهبود صحت آن خواهد شد. همچنین لحاظ نمودن حداکثر حاشیه در استخراج ویژگی‌های کاربران، منجر به بهبود کارایی مدل در مواجهه با آیتم‌هایی است که توسط کاربر امتیاز داده نشده اند.

همانطور که در شکل ۱ مشاهده می‌شود، درخت حاصل روش HMF یک درخت دودویی کاملا نامتوازن است. لذا زمان مورد نیاز برای آموزش، تست و همچنین دقت چنین درختی کارایی لازم را ندارد. لذا در این مقاله روشی ارایه می‌کنیم که ساختاری متوازن دارد و علاوه بر رفع مشکل روش HMF ، از داده‌های متنی نوشته شده توسط کاربران جهت بهبود دقت پیش‌بینی امتیازات بهره می‌برد.

درک مفهوم یک متن و استخراج ویژگی از آن نیازمند خواندن پیوسته کلمات و حفظ و نگهداری مفاهیم کلمات قبل است. شبکه‌های عصبی متداول چندان در این مورد موفق نبودند زیرا امکان نگهداری مفاهیم کلمات قبل را نداشتند. به‌عنوان نمونه شبکه‌های عصبی کانولوشنی از پنجره کلمات برای استخراج ویژگی استفاده می‌نمایند که با توجه به محدود بودن تعداد کلمات داخل پنجره امکان نگهداری مفاهیم از کلمات تاثیرگذار قبلی وجود ندارد. شبکه‌های عصبی بازگشتی با توجه به نوع طراحی‌شان این مشکل را برطرف کرده‌اند. این شبکه‌ها در داخل خود به‌صورت زنجیره‌ای و بازگشتی عمل می‌کنند و لذا بسیار مناسب برای کار کردن با داده‌های دنباله‌ای و لیستی می‌باشند. عملکرد شبکه‌های عصبی بازگشتی به گونه‌ای است که باعث می‌شود اطلاعاتی که از کلمات موجود در متن استخراج شده‌اند از بین نرفته و در شبکه باقی بمانند. شبکه عصبی LSTM عملکرد بسیار مناسبی در حوزه کار با دنباله‌ها از خود به نمایش گذاشته است. با توجه به توانایی شبکه‌های LSTM در کار با دنباله‌ها و اینکه متن نوعی دنباله شامل کلمات به هم مرتبط می‌باشد برای استخراج ویژگی از متون نوشته شده توسط کاربران و استفاده از آن‌ها در بهبود تخمین امتیاز کاربران به آیتم‌ها، از این نوع شبکه در طراحی مدل پیشنهادی استفاده نمودیم.

الگوریتم مربوط به مرحله آموزشی روش ارایه شده یک ساختار درختی متوازن ایجاد می‌کند که به‌طور مثال برای سیستمی با امتیازات یک تا پنج، با $low=1$ و $high=5$ و Y به‌عنوان ماتریس امتیازات آغاز به‌کار می‌نماید. این الگوریتم یک الگوریتم بازگشتی است که در هر گره درخت برای ایجاد گره فرزند خود را فراخوانی می‌کند. در هر گام، low و $high$ به ترتیب کمترین و بیشترین امتیاز باقیمانده برای گره فعلی می‌باشند. در هر تکرار اگر $low<high$ باشد آنگاه $mid = \frac{\text{low}+\text{high}}{2}$ و با توجه به مقدار mid و با استفاده از رابطه زیر عناصر ماتریس Y_{ij}^{mid} بدست می‌آیند.

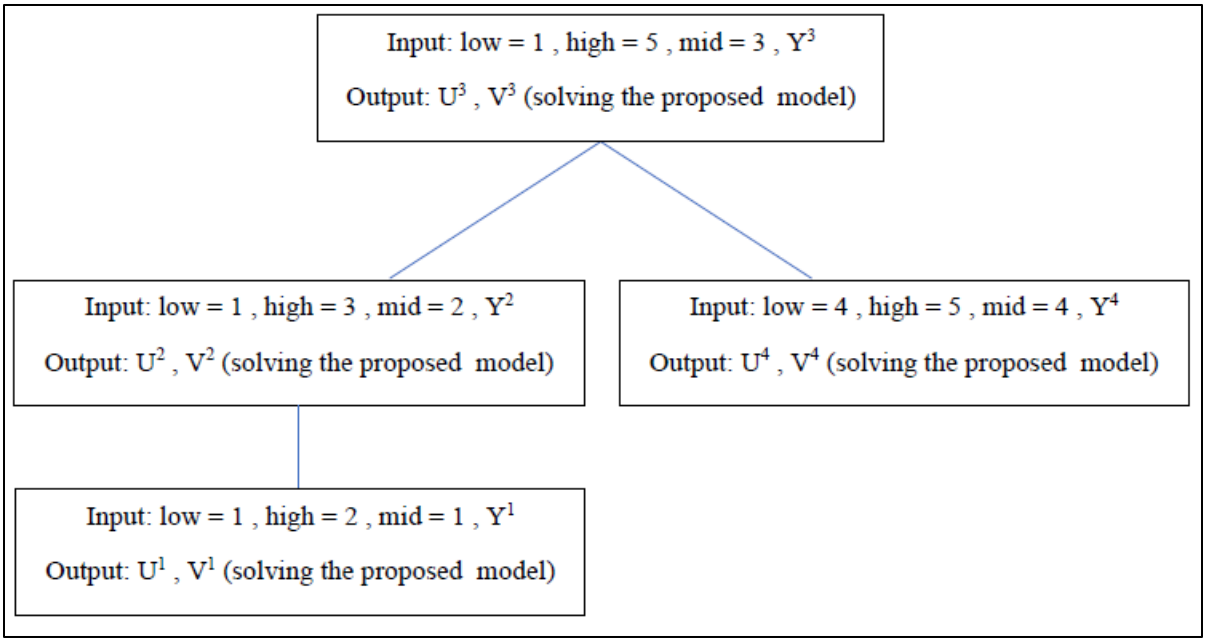
$$Y_{ij}^{mid} = \begin{cases} -1 & low \le Y_{ij} \le mid, \\ +1 & mid+1 \le Y_{ij} \le high. \end{cases} \quad (۱۱)$$

سپس با کمینه کردن تابع هزینه مدل پیشنهادی که در زیر آمده است، مقادیر ویژگی‌های پنهان کاربران یعنی U^{mid} و آیتم‌های متناسب با آن مرحله یعنی V^{mid} بدست می‌آید.

$$\min_{U^{mid},V^{mid}} J(U^{mid},V^{mid}) = \|V_j^{mid} - LSTM(X_j)\|^2 + \sum_{low \le Y_{ij} \le high} h\left(Y_{ij}^{mid}\left(U_i^{mid} {V_j^{mid}}^T\right)\right) + \frac{\lambda}{2}\left(\|U^{mid}\|_F^2 + \|V^{mid}\|_F^2\right) \quad (۱۲)$$

مقادیر U^{mid} و V^{mid} بدست آمده، برای استفاده در مرحله تست روش پیشنهادی ذخیره می‌گردند.

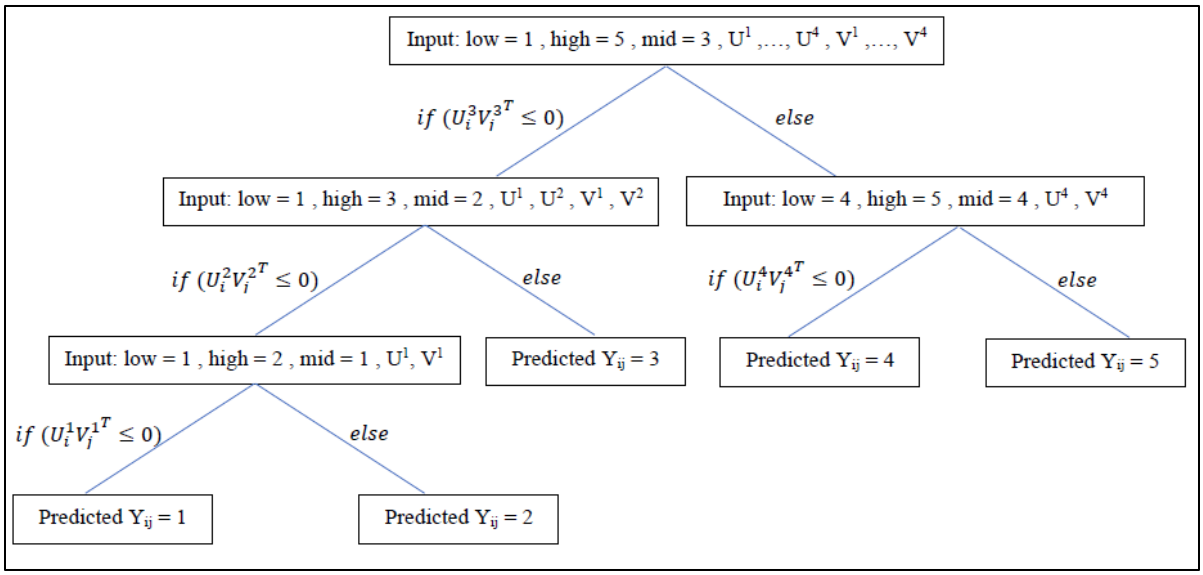
اجرای این مرحله باعث تقسیم مجموعه امتیازات باقیمانده یعنی $\{low, low+1, \dots, high\}$ به دو زیرمجموعه تقریبا برابر $\{low, low+1, \dots, mid\}$ و $\{mid+1, mid+2, \dots, high\}$ می‌گردد. سپس الگوریتم مجددا برای هر زیرمجموعه به‌صورت بازگشتی فراخوانی می‌گردد که نتیجه آن ساختن دو گره فرزند برای گره فعلی است. شکل ۲ مرحله آموزش روش پیشنهادی برای حالتی که امتیازات قابل اختصاص اعداد طبیعی ۱ تا ۵ می‌باشند را نمایش می‌دهد و همان‌طور که در شکل مشخص شده است ویژگی‌های پنهان کاربران و آیتم‌ها در مراحل مختلف با حل کردن مدل پیشنهادی محاسبه می‌گردند.



شکل ۲ : مرحله آموزش روش پیشنهادی و بدست آوردن ویژگی‌های پنهان کاربران و آیتم‌ها در مراحل مختلف.

در مرحله تست برای تخمین امتیاز کاربر i به آیتم j از ریشه به سمت برگ حرکت می‌کنیم و از مقادیر ویژگی‌های پنهان کاربران و آیتم‌ها که در مراحل مختلف مرحله آموزش بدست آمدند استفاده می‌نماییم. در هر گره پارامترهای low و $high$ وجود دارند که کمترین و بیشترین مقدار امتیاز در آن گره محسوب می‌شوند و لذا برای گره ریشه داریم $low=1$ و $high=R$. در گره برگ داریم $low=high$. در هر گره اگر آن گره برگ باشد امتیاز تخمینی برابر با low است. در غیر این‌صورت اگر

$U_i^{mid} V_j^{mid\,T} \leq 0$ ، آنگاه به سمت چپ حرکت می‌کنیم و درغیر این‌صورت به سمت راست حرکت می‌کنیم که در آن $mid = \left\lfloor \frac{low+high}{2} \right\rfloor$. شکل ۳ مرحله تست روش پیشنهادی و پیش‌بینی امتیازات نامعلوم را نمایش می‌دهد.



شکل ۳ : مرحله تست روش پیشنهادی و پیش‌بینی امتیازاتی که مقدارشان مشخص نبوده است. شبه‌کدهای زیر الگوریتم‌های آموزش و تست مربوط به روش پیشنهادی را بیان می‌کنند.

Algorithm 1. Train phase of the proposed method.

Input:

low, high, // minimum and maximum of remaining ratings

Y. // user-item rating matrix

Output:

$(U^1, V^1), (U^2, V^2), \ldots, (U^{R-1}, V^{R-1})$.

// Users and items latent factors for thresholded user-item rating matrices Y^q $(q = 1,2, \ldots, R-1)$.

Begin

If (high<=low)

return;

//Make the content of current node:

Set $mid = \left\lfloor \frac{low+high}{2} \right\rfloor$;

Set $Y_{ij}^{mid} = \begin{cases} -1 & low \leq Y_{ij} \leq mid, \\ +1 & mid+1 \leq Y_{ij} \leq high. \end{cases}$

Solve the following model:

$$\min_{U^{mid},V^{mid}} J(U^{mid},V^{mid}) = \sum_{low\le Y_{ij}\le high} h\left(Y_{ij}^{mid}\left(U_i^{mid}{V_j^{mid}}^T\right)\right) + \frac{\lambda}{2}\left(\left\|U^{mid}\right\|_F^2 + \left\|V^{mid}\right\|_F^2\right) + \left\|V_j^{mid} - LSTM(X_j)\right\|^2.$$

Call Algorithm1(low, mid, Y); //Make left child node

Call Algorithm1(mid+1, high, Y); //Make right child node

End

Algorithm 2. Test phase of the proposed method.

Input:

low, high, // Minimum and maximum of remaining ratings

// They must be set to 1 and R, respectively, by user.

i, //user No.

j, //item No.

$U^1,\dots,U^{R-1}$, //Latent feature vectors of users for each of R-1 two-class MMMFs.

$V^1,\dots,V^{R-1}$. //Latent feature vectors of items for each of R-1 two-class MMMFs.

Output:

Estimated rating of i-th user on j-th item.

Begin

If (high<=low)

return low; // It is a leaf node. Return estimated rating of i-th user on j-th item.

Set $mid = \left\lfloor\frac{low+high}{2}\right\rfloor$;

If $\left(U_i^{mid}{V_j^{mid}}^T \le 0\right)$

return Algorithm2(low, mid, i, j, $U^1,\dots,U^{R-1}, V^1,\dots,V^{R-1}$); //Go to left node

else

return Algorithm2(mid+1, high, i, j, $U^1,\dots,U^{R-1}, V^1,\dots,V^{R-1}$); //Go to right node

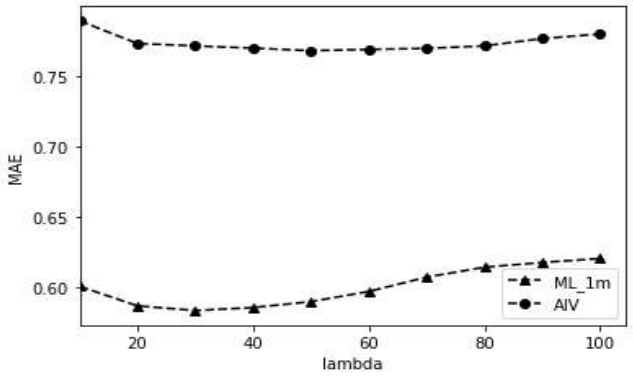
End

۵. نتایج عملی

پیاده‌سازی با استفاده از زبان پایتون ۳.۶ و کتاب‌خانه کراس انجام پذیرفت. ساختارهای مختلفی برای شبکه عصبی طراحی شده مورد بررسی قرار گرفت که توصیف نمونه‌نهایی آن بدین‌صورت است که برای بازنمایی کلمات از روش بردار تعبیه کلمات استفاده شد. برای این منظور بردارهای پنهان کلمات با اندازه ۲۰۰ در نظر گرفته شدند و با استفاده از لایه Embedding در کراس وارد شبکه می‌شوند. سه لایه LSTM با اندازه‌های ۱۲۸ و ۶۴ و ۳۲ ایجاد گردیدند. همچنین بین لایه‌های LSTM نیز لایه‌های Dropout با نرخ‌های ۰.۳ و ۰.۱ و ۰.۱ قرار داده شدند. آخرین لایه نیز Dense است که اندازه آن برابر با تعداد فاکتورهای پنهان آیتم‌ها (به‌عنوان مثال ۲۰) می‌باشد. برای ارزیابی روش پیشنهادی از مجموعه داده‌های MovieLens و Amazon استفاده گردید که در مقاله [25] نیز استفاده شده‌اند. این مجموعه داده‌ها شامل امتیازاتی در بازه *1-5* هستند که کاربران به آیتم‌های مختلف اختصاص داده‌اند. از آنجا که مجموعه داده MovieLens شامل اطلاعات متنی (مانند خلاصه فیلم) در مورد آیتم‌ها نیست لذا نویسندگان مقاله [25] اطلاعات متنی آیتم‌های مرتبط را از IMDB فراهم نموده‌اند و برای مجموعه داده Amazon نیز اطلاعات متنی همان نظرات کاربران در مورد آیتم‌هاست. همچنین نویسندگان مقاله [25] گام‌های مختلف پیش پردازش متنی را روی متون اجرا نمودند که به تفصیل بیان نموده‌اند و در اختیار سایر محققین قرار گرفته است[۳]. لذا ما نیز برای ارزیابی کارایی روش پیشنهادی از همان مجموعه داده‌ها استفاده نمودیم. میانگین خطای مطلق یا *MAE* [۴] یکی از پرکاربردترین معیارها در ارزیابی سیستم‌های توصیه‌گر است. این معیار میانگین خطای مدل را به‌صورت زیر محاسبه می‌نماید:

$$MAE = \frac{\sum_{(i,j)\in\Omega}|Y_{ij} - Predict_{ij}|}{|\Omega|} \qquad (۱۳)$$

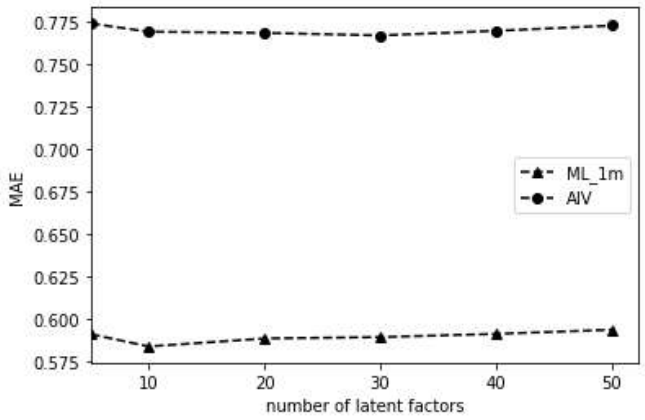
مقدار بهینه پارامتر تنظیمی λ و تعداد فاکتورهای پنهان k به‌ترتیب از مجموعه‌های {10,20,30,40,50,60,70,80,90,100} و {5,10,20,30,40,50} انتخاب گردیدند. شکل ۴ برای تعداد فاکتورهای پنهان *k=20* تغییرات *MAE* روش پیشنهادی به‌واسطه تغییر پارامتر تنظیمی λ را بیان می‌نماید. همچنین شکل ۵ تغییرات *MAE* بعد از تنظیم پارامتر λ را به ازای تعداد فاکتورهای پنهان مختلف نمایش می‌دهد. همان‌طور که در شکل نیز مشخص است استفاده از تعداد فاکتورهای پنهان بیشتر از ۵۰ تاثیر چندانی روی دقت روش پیشنهادی ندارد.



شکل ۴: تاثیر تغییر پارامتر تنظیمی λ روی MAE روش پیشنهادی برای k=20 .

[۳] http://dm.postech.ac.kr/~cartopy/ConvMF

[۴] Mean Absolute Error



شکل ۵: تاثیر تغییر تعداد فاکتورهای پنهان روی MAE روش پیشنهادی برای k=20 .

برای ارزیابی کارآیی روش پیشـنهادی، آنرا با روش ارایه شـده در مقاله [25] که آنرا R-ConvMF نامیدند و روش HMF مقایسه کرده‌ایم. جدول ۱ نتیجه مقایسه روش پیشنهادی با روش‌های بیان شده را برحسب *MAE* نشان می‌دهد. همان‌طور که دیده می‌شود میزان خطای روش HMF از روش پیشنهادی بیشتر است زیرا HMF برای مرحله آموزش، تعدادی دسته بند دو کلاسه با حداکثر حاشیه می‌سازد که منجر به ایجاد یک درخت نامتوازن یک‌طرفه می‌گردد که در شکل ۱ نمایش داده شده بود. اما روش پیشنهادی به شیوه‌ای طراحی گردیده است که منجر به آموزش تعدادی دسته‌بند دو کلاسه می‌گردد که درختی متوازن را نتیجه می‌دهد. لذا به دلیل متوازن بودن این درخت، دقت دسته‌بندهای دو کلاسه طراحی شده بیشتر از روش HMF است. مزیت دیگری که باعث بهبود عملکرد روش پیشـنهادی نسـبت به روش HMF شـده است استفاده از داده‌های متنی است که منجر به اسـتفاده بهتر از دانش موجود در سـیسـتم گردیده اسـت. همچنین همانطور که در جدول ۱ آمده اسـت دقت روش پیشـنهادی از روش ارایه شـده در R-ConvMF نیز بیشـتر اسـت. دلیل بهبود کارایی این است که روش پیشنهادی علاوه بر استفاده از اطلاعات متنی در مورد آیتم‌ها که در R-ConvMF نیز از آن‌ها استفاده گردیده است، یک مدل پیشگوی غیرخطی ارایه کرده است که در هنگام بدست آوردن فاکتورهای پنهان کاربران و آیتم‌ها حداکثر حاشیه را نیز لحاظ نموده است.

جدول ۱: خطای تخمینی برحسب *MAE*.

Method	MovieLens-1M	AIV
HMF [36]	0.6598 ± 0.0055	0.8723 ± 0.0056
R-ConvMF [25]	0.6058 ± 0.0028	0.8088 ± 0.0037
Proposed Method	**0.5825 ± 0.0044**	**0.7868 ± 0.0046**

۶. نتیجه‌گیری و کارهای آینده

همانطور که گفته شـد داده‌های متنی و اسـناد توصیفی موجود در سیستم‌های اطلاعاتی می‌توانند در بهبود عملکر سیستم‌های توصیه‌گر مورد استفاده قرار بگیرند. پژوهش‌های مختلف در این حوزه مورد بررسی قرار گرفتند. برخی از روش‌ها اطلاعات زمینه‌ای معنایی مانند ترتیب کلمات موجود در اسـناد توصـیفی را در نظر نمی‌گیرند که این‌کار باعث کاهش کیفیت استخراج ویژگی از متن می‌گردد. همچنین طبق بررسـی انجام شده، برخی از روش‌ها برای طراحی مدلی که بتواند امتیاز یک کاربر به

یک آیتم را تشـخیص دهد از روش‌های مبتنی بر فاکتورسـازی ماتریس (با مرتبه پایین یا نرم پایین) اسـتفاده می‌نمایند که این روش‌ها نسبت به فاکتورهای پنهان کاربر یا آیتم خطی هستند و لذا توانایی پیش‌بینی رفتارهای پیچیده کاربران را ندارند. علاوه براین روش‌های مبتنی بر فاکتورسازی ماتریس با مرتبه پایین در هنگام استخراج ویژگی‌های کاربران و آیتم‌ها حداکثر حاشیه را در نظر نمی‌گیرند که این موضوع نیز باعث کاهش دقت مدل پیش‌بینی می‌گردد. لذا در روش پیشنهادی از شبکه‌های بازگشتی LSTM اسـتفاده گردید که نسبت به مدل‌های مبتنی بر کیسه کلمات عملکرد بهتری دارند و ضمن لحاظ کردن ترتیب کلمات توانایی حفظ اطلاعات مفید دنباله‌های طولانی‌تر و بازنمایی معنایی پنهان بهتر متون را نیز دارند. با استفاده از استخراج ویژگی با کمک شبکه‌های بازگشتی LSTM از روی اسناد توصیفی در مورد آیتم‌ها، و لحاظ نمودن حداکثر حاشیه در هنگام استخراج ویژگی‌های کاربران و آیتم‌ها یک مدل پیش‌بینی جدید و غیرخطی طراحی گردید. مدل طراحی شده در آزمایش‌ها انجام گرفته در مقایسه با روش R-ConvMF به‌عنوان روشی موثر برای استفاده از متون توصیفی در پیش‌بینی امتیازات عملکرد بهتری را نمایش داد که علت آن لحاظ نمودن حداکثر حاشیه در استخراج ویژگی‌های کاربران و آیتم‌هاست. همچنین مدل پیشنهادی در مقایسه با HMF به عنوان روشی موثر که از فاکتورسازی ماتریس با حداکثر حاشیه بهره می‌برند نیز دقت بالاتری را نشان داد که علت آن اسـتفاده از متون توصـیفی موجود در سـیسـتم و همچنین سـاخت دسته‌بندهای دوکلاسه از داده‌هایی که متوازن می‌باشند. می‌توان در آینده برای بهبود عملکرد HMF متوازن پیشنهادی روشی ارایه کرد که ساختار درختی متفاوتی را ایجاد نمـاید. به‌عنوان مثال با لحاظ نمودن تعداد موجود از هر امتیاز ممکن اســت بتوان ایده مطرح شــده در HMF متوازن را با الگوریتم‌هایی مانند AVL ترکیب کرد تا به سـاختار درختی جدیدی دست یافت و مجموعه آموزشی متوازن‌تری داشت. تغییر دادن تابع هزینه مدل پیشنهادی که از اسناد متنی استفاده می‌نمود و همچنین بکاربردن تکنیک‌های دیگر شبکه‌های عصبی عمیق مانند مکانیزم توجه نیز به‌عنوان کارهای آینده پیشنهاد می‌گردد.

مراجع

[1] Esparza, S. G., O'Mahony, M. P., & Smyth, B. (2010, December). Effective product recommendation using the real-time web. In International Conference on Innovative Techniques and Applications of Artificial Intelligence (pp. 5-18). Springer, London.

[2] Garcia Esparza, S., O'Mahony, M. P., & Smyth, B. (2011, October). A multi-criteria evaluation of a user generated content based recommender system. In Presented at the 3rd Workshop on Recommender Systems and the Social Web (RSWEB-11), 5th ACM Conference on Recommender Systems, Chicago, IL, USA, 23-27 October 2011.

[3] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), 993-1022.

[4] Seroussi, Y., Bohnert, F., & Zukerman, I. (2011, June). Personalised rating prediction for new users using latent factor models. In Proceedings of the 22nd ACM conference on Hypertext and hypermedia (pp. 47-56). ACM.

[5] McAuley, J., & Leskovec, J. (2013, October). Hidden factors and hidden topics: understanding rating dimensions with review text. In Proceedings of the 7th ACM conference on Recommender systems (pp. 165-172). ACM.

[6] Leung, C. W., Chan, S. C., & Chung, F. L. (2006, August). Integrating collaborative filtering and sentiment analysis: A rating inference approach. In Proceedings of the ECAI 2006 workshop on recommender systems (pp. 62-66).

[7] Zhang, W., Ding, G., Chen, L., Li, C., & Zhang, C. (2013). Generating virtual ratings from chinese reviews to augment online recommendations. ACM Transactions on intelligent systems and technology (TIST), 4(1), 9.

[8] Poirier, D., Tellier, I., Fessant, F., & Schluth, J. (2010, April). Towards text-based recommendations. In Adaptivity, Personalization and Fusion of Heterogeneous Information (pp. 136-137). LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

[9] Wawre, S. V., & Deshmukh, S. N. (2016). Sentiment classification using machine learning techniques. International Journal of Science and Research (IJSR), 5(4), 819-821.

[10] Ganapathibhotla, M., & Liu, B. (2008, August). Mining opinions in comparative sentences. In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1 (pp. 241-248). Association for Computational Linguistics.

[11] Shaikh, M. A. M., Prendinger, H., & Ishizuka, M. (2009). A linguistic interpretation of the OCC emotion model for affect sensing from text. In Affective information processing (pp. 45-73). Springer, London.

[12] Raghavan, S., Gunasekar, S., & Ghosh, J. (2012, September). Review quality aware collaborative filtering. In Proceedings of the sixth ACM conference on Recommender systems (pp. 123-130). ACM.

[13] Wang, H., Wang, N., & Yeung, D. Y. (2015, August). Collaborative deep learning for recommender systems. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1235-1244). ACM.

[14] Wu, L., Quan, C., Li, C., Wang, Q., Zheng, B., & Luo, X. (2019). A context-aware user-item representation learning for item recommendation. ACM Transactions on Information Systems (TOIS), 37(2), 22.

[15] Bao, Y., Fang, H., & Zhang, J. (2014, June). Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In Twenty-Eighth AAAI conference on artificial intelligence.

[16] McAuley, J., & Leskovec, J. (2013, October). Hidden factors and hidden topics: understanding rating dimensions with review text. In Proceedings of the 7th ACM conference on Recommender systems (pp. 165-172). ACM.

[17] Tan, Y., Zhang, M., Liu, Y., & Ma, S. (2016, July). Rating-boosted latent topics: Understanding users and items with ratings and reviews. In IJCAI (Vol. 16, pp. 2640-2646).

[18] Wang, C., & Blei, D. M. (2011, August). Collaborative topic modeling for recommending scientific articles. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 448-456). ACM.

[19] Catherine, R., & Cohen, W. (2017, August). Transnets: Learning to transform for recommendation. In Proceedings of the Eleventh ACM Conference on Recommender Systems (pp. 288-296). ACM.

[20] Kim, D., Park, C., Oh, J., Lee, S., & Yu, H. (2016, September). Convolutional matrix factorization for document context-aware recommendation. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 233-240). ACM.

[21] Zheng, L., Noroozi, V., & Yu, P. S. (2017, February). Joint deep modeling of users and items using reviews for recommendation. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (pp. 425-434). ACM.

[22] Seo, S., Huang, J., Yang, H., & Liu, Y. (2017, August). Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In Proceedings of the Eleventh ACM Conference on Recommender Systems (pp. 297-305). ACM.

[23] Ling, G., Lyu, M. R., & King, I. (2014, October). Ratings meet reviews, a combined approach to recommend. In Proceedings of the 8th ACM Conference on Recommender systems (pp. 105-112). ACM.

[24] Zhang, W., Yuan, Q., Han, J., & Wang, J. (2016, July). Collaborative multi-Level embedding learning from reviews for rating prediction. In IJCAI (pp. 2986-2992).

[25] Kim, D., Park, C., Oh, J., & Yu, H. (2017). Deep hybrid recommender systems via exploiting document context and statistics of items. Information Sciences, 417, 72-87.

[26] Chen, C., Zhang, M., Liu, Y., & Ma, S. (2018, April). Neural attentional rating regression with review-level explanations. In Proceedings of the 2018 World Wide Web Conference (pp. 1583-1592). International World Wide Web Conferences Steering Committee.

[27] Lu, Y., Dong, R., & Smyth, B. (2018, April). Coevolutionary recommendation model: Mutual learning between ratings and reviews. In Proceedings of the 2018 World Wide Web Conference (pp. 773-782). International World Wide Web Conferences Steering Committee.

[28] Xing, S., Wang, Q., Zhao, X., & Li, T. (2019). A hierarchical attention model for rating prediction by leveraging user and product reviews. Neurocomputing, 332, 417-427.

[29] Zheng, L., Cao, B., Noroozi, V., Philip, S. Y., & Ma, N. (2017, December). Hierarchical collaborative embedding for context-aware recommendations. In 2017 IEEE International Conference on Big Data (Big Data) (pp. 867-876). IEEE.

[30] Xing, S., Liu, F., Zhao, X., & Li, T. (2018). Points-of-interest recommendation based on convolution matrix factorization. Applied intelligence, 48(8), 2458-2469.

[31] Wu, H., Zhang, Z., Yue, K., Zhang, B., He, J., & Sun, L. (2018). Dual-regularized matrix factorization with deep neural networks for recommender systems. Knowledge-Based Systems, 145, 46-58.

[32] Zhang, J. D., & Chow, C. Y. (2018). SEMA: Deeply learning semantic meanings and temporal dynamics for recommendations. IEEE Access, 6, 54106-54116.

[33] Rennie, J. D., & Srebro, N. (2005, July). Loss functions for preference levels: Regression with discrete ordered labels. In Proceedings of the IJCAI multidisciplinary workshop on advances in preference handling (pp. 180-186). Kluwer Norwell, MA.

[34] Rennie, J. D., & Srebro, N. (2005, August). Fast maximum margin matrix factorization for collaborative prediction. In Proceedings of the 22nd international conference on Machine learning (pp. 713-719). ACM.

[35] Srebro, N., Rennie, J., & Jaakkola, T. S. (2005). Maximum-margin matrix factorization. In Advances in neural information processing systems (pp. 1329-1336).

[36] Kumar, V., Pujari, A. K., Sahu, S. K., Kagita, V. R., & Padmanabhan, V. (2017). Collaborative filtering using multiple binary maximum margin matrix factorizations. Information Sciences, 380, 1-11.

[37] Sharma, *S., Rana, V., & Kumar, V. (2021). Deep learning based semantic personalized recommendation system. International Journal of Information Management Data Insights, 1(2), 100028.*