مجله محاسبات نرم، جلد 14، شماره 2 (پاییز و زمستان 1404)، صفحه 54ـ65
https://doi.org/10.22052/scj.2025.255465.1267

مجله محاسبات نرم
SOFT COMPUTING JOURNAL
تارنمای مجله: scj.kashanu.ac.ir

دانشگاه کاشان
University of Kashan

ارتقاء فرآیند مهندسی نیازمندی‌ها با استفاده از سیستم‌های توصیه‌گر ترکیبی✧

محمدمهدی پورهاشم کله بستی[1*]، استادیار، جمشید پیرگزی[1]، استادیار، علی قنبری سرخی[1]، استادیار، علی کرمانی[1]، استادیار

[1] دانشکده مهندسی برق و کامپیوتر، دانشگاه علم و فناوری مازندران، بهشهر، ایران.

اطلاعات مقاله

تاریخچه مقاله:
دریافت 27 شهریور ماه 1403
پذیرش 22 آذر ماه 1403

کلمات کلیدی:
مهندسی نیازمندی‌ها
سیستم‌های توصیه‌گر
پالایش مشارکتی
پالایش مبتنی بر محتوا
مدل بردارهای سراسری واژگان (GloVe)

چکیده

مهندسی نیازمندی‌ها یکی از مراحل مهم و حیاتی در فرآیند توسعه نرم‌افزار به شمار می‌آید که اجرای صحیح آن تاثیر بسزایی در موفقیت پروژه دارد. به منظور افزایش کیفیت و کارایی مهندسی نیازمندی‌ها، استفاده مجدد از نیازمندی‌های پیشین به عنوان روشی موثر مطرح شده است. سیستم‌های توصیه‌گر با بهره‌گیری از روش‌هایی نظیر پالایش مشارکتی، پالایش مبتنی بر محتوا، تکنیک‌های مبتنی بر دانش و رویکردهای ترکیبی، به ذینفعان در شناسایی و اولویت‌بندی بهینه نیازمندی‌ها کمک می‌کنند. این مقاله یک روش ترکیبی پالایش مشارکتی و پالایش مبتنی بر محتوا برای بهبود عملکرد سیستم‌های توصیه‌گر در مهندسی نیازمندی‌ها ارائه می‌دهد. در این روش، شباهت معنایی بین نیازمندی‌ها با استفاده از مدل بردارهای سراسری واژگان (GloVe) محاسبه شده و ویژگی‌های معنایی استخراج شده از توصیفات متنی با دسته‌بندی‌های از پیش تعریف شده ترکیب می‌شوند. این فرآیند دقت پیش‌بینی علاقه یا نیاز ذینفعان به نیازمندی‌ها و اولویت‌بندی آنها را افزایش می‌دهد. نتایج تجربی به دست آمده از مجموعه داده RALIC نشان می‌دهد که رویکرد پیشنهادی به طور معناداری دقت و پوشش پیش‌بینی را افزایش می‌دهد. این بهبود به ویژه در مواجهه با چالش‌های پراکندگی داده‌ها و مشکل شروع سرد مشهود است که نشان‌دهنده کارایی بالاتر این روش در مقایسه با رویکردهای موجود می‌باشد.

© 1403 نویسندگان. مقاله با دسترسی آزاد تحت مجوز CC-BY

1. مقدمه

یکی از کلیدی‌ترین مراحل در چرخه مهندسی نرم‌افزار، مهندسی نیازمندی‌ها[1] است که شامل مراحلی چون استخراج نیازمندی‌ها، تحلیل جزئیات، مذاکره، مشخص‌سازی، اعتبارسنجی و مدیریت نیازمندی‌های ذینفعان می‌باشد. این فرآیندها به طور مستقیم بر کیفیت و موفقیت پروژه تاثیر می‌گذارند [1]. از آنجا که این مرحله تاثیر قابل‌توجهی بر موفقیت پروژه دارد، از اهمیت بالایی برخوردار است. در صورتی که فرآیندهای مهندسی نیازمندی‌ها به درستی انجام نشوند، ممکن است منجر به عقب افتادن از زمان‌بندی، کمبود منابع مالی و حتی شکست پروژه به دلیل نادیده گرفتن برخی از نیازمندی‌ها شود.

برای بهبود کیفیت و کارایی استخراج نیازمندی‌ها، استفاده مجدد از نیازمندی‌ها به عنوان یک تکنیک موثر شناخته شده است. این فرآیند شامل جمع‌آوری نیازمندی‌ها از پروژه‌های قبلی و اعمال

✧ نوع مقاله: پژوهشی

* نویسنده مسئول
پست(های) الکترونیک: pourhashem@mazust.ac.ir (پورهاشم کله بستی)
j.pirgazi@mazust.ac.ir (پیرگزی)
ali.ghanbari@mazust.ac.ir (قنبری سرخی)
a_kermani@mazust.ac.ir (کرمانی)

[1] Requirements engineering

نحوه ارجاع به مقاله: محمدمهدی پورهاشم کله بستی، جمشید پیرگزی، علی قنبری سرخی، علی کرمانی، «ارتقاء فرآیند مهندسی نیازمندی‌ها با استفاده از سیستم‌های توصیه‌گر ترکیبی»، مجله محاسبات نرم، جلد 14، شماره 2، ص54ـ65، پاییز و زمستان 1404.

آنها به پروژه‌های جدید و مرتبط است. مزایای استفاده مجدد از نیازمندی‌ها شامل بهبود کارایی استخراج، کاهش زمان عرضه به بازار، کیفیت بالاتر نیازمندی‌ها و کاهش هزینه‌های توسعه است [2]. در این راستا، سیستم‌های توصیه‌گر می‌توانند نقش مهمی در بهبود فرآیند استفاده مجدد از نیازمندی‌ها ایفا کنند. به این صورت که با توجه به سابقه انتخاب‌ها و نیازهای کاربر مدنظر، داده‌های موجود را تحلیل کرده و توصیه‌هایی مبتنی بر نیازمندی‌های مشابه در پروژه‌های قبلی ارائه دهند. این تحلیل می‌تواند به شناسایی نیازمندی‌های حیاتی که ممکن است نادیده گرفته شوند، کمک کند و بدین ترتیب، فرآیند استخراج نیازمندی‌ها را کارآمدتر و هدفمندتر نماید. سیستم‌های توصیه‌گر ابزارهای هوشمندی هستند که برای مقابله با حجم بالای اطلاعات، با پالایش کردن و بازیابی مناسب‌ترین اقلام برای کاربران بر اساس سوابق تاریخی و ترجیحات آنها طراحی شده‌اند. این سیستم‌ها نقش مهمی در حمایت از ذینفعان مهندسی نیازمندی‌ها ایفا می‌کنند. این سیستم‌ها با شناسایی نیازمندی‌های ممکن برای نرم‌افزار در حال توسعه، می‌توانند مجموعه داده‌های گسترده‌ای از جمله داده‌های پروژه‌های گذشته و بازخورد کاربران را تحلیل کنند تا توصیه‌هایی برای نیازمندی‌های بالقوه ارائه دهند. با استفاده از این توصیه‌ها، ذینفعان می‌توانند تا حد زیادی اطمینان حاصل نمایند که هیچ نیازمندی حیاتی نادیده گرفته نمی‌شود. این رویکرد جامع منجر به توسعه راه‌حل‌های نرم‌افزاری کاربرمحورتر می‌شود.

سیستم‌های توصیه‌گر از روش‌های مختلفی برای ارائه توصیه‌ها استفاده می‌کنند، از جمله پالایش مشارکتی[1]، پالایش مبتنی بر محتوا[2]، روش‌های مبتنی بر دانش[3] و رویکردهای ترکیبی. روش‌های پالایش مشارکتی با استفاده از تاریخچه تعاملات کاربر-اقلام، توصیه‌ها را بر اساس شباهت‌های بین کاربران یا اقلام ارائه می‌دهند. روش‌های مبتنی بر محتوا، با تحلیل ویژگی‌های اقلام، توصیه‌هایی ارائه می‌دهند [3]- [6]. روش‌های مبتنی بر دانش با تکیه بر اطلاعات موجود درباره کاربران، اقلام و روابط میان آنها و همچنین استفاده از قوانین منطقی، به تحلیل نیازها و ویژگی‌های محصولات می‌پردازند تا پیشنهاداتی دقیق و منطبق بر نیازهای خاص کاربران ارائه دهد. در نهایت، روش‌های ترکیبی با ادغام چندین تکنیک توصیه‌گری، تلاش دارند تا از مزایای هر یک بهره‌برداری کرده و نقاط ضعف آنها را کاهش دهند. استفاده از سیستم‌های توصیه‌گر در مهندسی نیازمندی‌ها بر این فرض استوار است که ذینفعان به‌عنوان کاربران و نیازمندی‌ها به عنوان اقلام مورد توصیه در نظر گرفته می‌شوند.

این مقاله رویکرد ترکیبی نوینی را معرفی می‌کند که با بهره‌گیری همزمان از نقاط قوت پالایش مشارکتی و پالایش مبتنی بر محتوا، دقت و کارایی سیستم توصیه‌گر را بهبود می‌بخشد. هدف نهایی سیستم‌های توصیه‌گر در این زمینه تولید فهرستی مرتب از نیازمندی‌ها بر اساس اولویت‌های هر ذینفع است، به طوری که بتواند به طور خاص، میزان علاقه یا نیاز هر ذینفع به هر نیازمندی را پیش‌بینی کند. یکی از نوآوری‌های کلیدی این پژوهش محاسبه دقیق‌تر شباهت معنایی بین نیازمندی‌ها است؛ به این صورت که ابتدا با استفاده از مدل بردارهای سراسری واژگان[4] (GloVe) ویژگی‌های معنایی توصیفات متنی نیازمندی‌ها استخراج می‌شود. این ویژگی‌ها سپس با دسته‌بندی‌های از پیش تعریف شده ترکیب شده و امکان محاسبه شباهت میان نیازمندی‌ها را با دقت بیشتری فراهم می‌سازد. این فرایند موجب می‌شود علاوه بر دسته‌بندی نیازمندی‌ها، محتوای متنی و معنای ضمنی آنها نیز در نظر گرفته شود و بدین ترتیب، توصیه‌های دقیق‌تر و اولویت‌بندی شده بر اساس علاقه‌مندی هر ذینفع حاصل گردد. نتایج تجربی حاصل از ارزیابی‌ها روی مجموعه داده RALIC نشان می‌دهد که روش پیشنهادی، نسبت به روش‌های پیشین، بهبود قابل‌توجهی در دو معیار اصلی دقت و پوشش پیش‌بینی به همراه دارد. این رویکرد، علاوه بر افزایش کارایی در توصیه نیازمندی‌ها، چالش‌های پراکندگی داده‌ها و شروع سرد را بهتر از روش‌های دیگر مدیریت می‌کند.

سایر بخش‌های مقاله به شرح زیر سازماندهی شده است. بخش 3 این مقاله روش پیشنهادی را معرفی می‌کند و به تشریح مبانی

[1] Collaborative filtering
[2] Content-based filtering
[3] Knowledge-based
[4] Global Vectors for Word Representation (GloVe)

نظری و مراحل اجرایی آن می‌پردازد. بخش 4 به ارزیابی و تحلیل نتایج اختصاص دارد و در بخش 5، نتیجه‌گیری و پیشنهادهایی برای تحقیقات آتی ارائه می‌گردد.

2. کارهای مرتبط

با توجه به گسترش پروژه‌های نرم‌افزاری و افزایش تعاملات میان ذینفعان و تیم مهندسی نرم‌افزار، نیاز به ابزارهایی که بتوانند نیازمندی‌های کاربران را به طور دقیق شناسایی و اولویت‌بندی کنند، بیش از پیش احساس می‌شود. در این راستا، پژوهش‌های متعددی به بررسی کاربردها و چالش‌های سیستم‌های توصیه‌گر در این حوزه پرداخته‌اند. این پژوهش‌ها به‌ویژه بر مشکلاتی همچون پراکندگی داده‌ها، مساله شروع سرد و پوشش تمرکز دارند که می‌توانند دقت و کارایی توصیه‌ها را تحت تاثیر قرار دهند. در این بخش، به مرور تحقیقات مرتبط در زمینه استفاده از سیستم‌های توصیه‌گر در مهندسی نیازمندی‌ها پرداخته و راه‌کارهای مختلف پیشنهادی برای بهبود این فرآیندها مورد بررسی قرار می‌گیرد.

مقاله [7] به بررسی کاربرد نوآورانه سیستم‌های توصیه‌گر در حوزه مهندسی نیازمندی‌های نرم‌افزار می‌پردازد و بر نقش کلیدی آنها در تسهیل فرآیند پیچیده کشف نیازمندی‌ها تاکید می‌کند. نویسندگان با رویکردی جامع، نشان می‌دهند که چگونه این سیستم‌ها می‌توانند از فعالیت‌های کاربران در انجمن‌های برخط نیازمندی‌های مورد نظر خود را استخراج نمایند. با تحلیل تعاملات و درخواست‌های ویژگی کاربران، سیستم می‌تواند به صورت هوشمندانه موضوعات مرتبط را به آنها پیشنهاد دهد و همچنین آنها را به کارشناسان و ذینفعان خبره در آن حوزه متصل کند. این امر به ویژه در شرایطی موثر خواهد بود که ذینفعان از نظر جغرافیایی دور از هم بوده و ارتباط چهره به چهره حضوری به آسانی مقدور نباشد.

مرجع [8]، یک سیستم توصیه‌گر ارائه می‌دهد که ویژگی‌های محصول را برای یک حوزه خاص مدل‌سازی و توصیه می‌کند. این سیستم با استفاده از متن‌کاوی و یک الگوریتم خوشه‌بندی انتشار تدریجی[1]، ویژگی‌های خاص حوزه را استخراج می‌کند. سپس یک مدل احتمالی ویژگی‌ها ایجاد می‌کند که مشترکات، تفاوت‌ها و ویژگی‌های بین دسته‌بندی‌ها را نشان می‌دهد. در نهایت، با استفاده از یادگیری قانون وابستگی[2] و یادگیری ماشین k-نزدیک‌ترین همسایه[3]، ویژگی‌هایی خاص از محصول را به کاربران توصیه می‌کند.

پروژه INTELLIREQ [9]، تکنیک‌های هوشمندی را برای بهبود کیفیت فرآیندهای مهندسی نیازمندی‌ها معرفی می‌کند. رویکردهای متنوعی برای پشتیبانی از ذینفعان در فعالیت‌هایی نظیر تعریف، تضمین کیفیت، استفاده مجدد و برنامه‌ریزی انتشار موجود می‌باشد. نویسندگان تمامی این رویکردها را بستر واحدی فراهم نموده‌اند. مطالعات تجربی نشان می‌دهند که چگونه این رویکردهای توصیه‌ای می‌توانند کیفیت فرآیندهای مهندسی نیازمندی‌ها را با ارائه اطلاعات تکمیلی، شناسایی روابط پنهان بین نیازمندی‌ها و حمایت از سناریوهای تصمیم‌گیری افزایش دهند.

مذاکره و اولویت‌بندی نیازمندی‌ها نیز از جمله مسائلی است که سیستم‌های توصیه‌گر می‌توانند در آن کارآمد باشند. در مقاله [10]، روشی پیشنهاد شده است که ذینفعان را شناسایی کرده، نیازمندی‌های عملکردی و غیرعملکردی را تفکیک می‌کند. این روش از توابع عضویت فازی برای نمایش متغیرهای زبانی استفاده می‌کند تا کیفیت فرآیند مذاکره و اولویت‌بندی نیازمندی‌ها را از طریق ارائه توصیه‌های شخصی‌سازی شده به ذینفعان بهبود بخشد. مقاله [11] نیز مساله مشابه اولویت‌بندی نیازمندی‌ها را با استفاده از الگوریتم‌های ژنتیک با تمرکز بر مقیاس‌پذیری در حل مساله مورد بررسی قرار داده است. در همین راستا، مقاله [12] به چالش‌های مقیاس‌پذیری و خودکارسازی در روش‌های سنتی اولویت‌بندی نیازمندی‌ها پرداخته و یک روش نیمه‌خودکار چندمعیاره برای اولویت‌بندی پیشنهاد نموده است. این روش از تکنیک‌های پردازش زبان طبیعی و الگوریتم‌های یادگیری ماشین برای اولویت‌بندی

[1] Incremental diffusive clustering
[2] Association rule mining
[3] K-Nearest Neighbor

نیازمندی‌های عملکردی و غیرعملکردی در پروژه‌های نرم‌افزاری استفاده می‌کند.

در سیستم‌های توصیه‌گر، مشکل پراکندگی[1] داده‌ها و مساله شروع سرد[2] از جمله چالش‌های اساسی محسوب می‌شوند که می‌توانند بر کارایی این سیستم‌ها تاثیر منفی بگذارند. مشکل پراکندگی زمانی رخ می‌دهد که ماتریس‌های تعامل کاربر-اقلام تا حد زیادی خالی باشند؛ به این معنا که اکثر کاربران تنها با تعداد محدودی از اقلام موجود تعامل داشته‌اند. این کمبود داده‌ها باعث می‌شود که سیستم نتواند به درستی ترجیحات کاربران را برای بخش عمده‌ای از اقلام پیش‌بینی کند و در نتیجه، توصیه‌ها با دقت کمتری ارائه شوند. از سوی دیگر، مساله شروع سرد زمانی بروز می‌کند که کاربران یا اقلام جدیدی به سیستم اضافه شود. به دلیل عدم وجود سابقه تعامل کافی، تولید توصیه‌های شخصی‌سازی شده یا ارزیابی دقیق از آیتم‌های جدید دشوار می‌شود. هر دو این چالش‌ها می‌توانند عملکرد کلی سیستم‌های توصیه‌گر را تضعیف کنند.

مقاله [13] به بررسی فرآیند استخراج مداوم نیازمندی‌ها و اولویت‌بندی آنها از طریق ترکیب داده‌های نظارت بر استفاده و بازخورد کاربران می‌پردازد. در این مطالعه یک سیستم توصیه‌گر نیمه‌خودکار ارائه می‌شود که با همبسته‌سازی دقیق داده‌های نظارتی و بازخوردها در سطح موارد کاربرد، به اولویت‌بندی نیازمندی‌ها می‌پردازد. روش پیشنهادی شامل چهار مرحله است که در آن داده‌ها جمع‌آوری و همبسته شده و بر اساس همبستگی یافته‌ها، اولویت‌بندی پیش‌فرض تعیین می‌شود.

پژوهش‌های قید شده تلاش کرده‌اند با تکیه بر تکنیک‌هایی مانند متن‌کاوی، یادگیری قانون وابستگی و پالایش مشارکتی به بهبود فرآیند کشف و اولویت‌بندی نیازمندی‌ها کمک کند؛ با این حال، ضعف‌هایی نظیر نیاز به داده‌های زیاد و محدودیت در کارایی در شرایط داده‌های پراکنده همچنان به عنوان چالش‌هایی محسوب می‌شوند. تا جایی که ما می‌دانیم، تنها در دو کار تحقیقاتی، سیستم‌های توصیه‌گر به نحوی طراحی شده‌اند که با در نظر گرفتن مشکلات پراکندگی داده‌ها و شروع سرد با استفاده از امتیازات صریح ذینفعان به نیازمندی‌ها، می‌توانند نیازمندی‌های جدیدی را به هر ذینفع توصیه کنند. این سیستم‌ها با تحلیل بازخوردها (امتیازات) کاربران، به بهبود فرآیند توصیه و افزایش دقت آن کمک می‌کنند. شامبور و همکاران [14]، یک سیستم توصیه‌گر ترکیبی بر اساس پالایش مشارکتی کاربر-اقلام، پیشنهاد داده‌اند و پس از آن، در مقاله [15] با ارتقا سیستم قبلی، یک روش ترکیبی پالایش مشارکتی مبتنی بر محتوا برای پشتیبانی از مهندسی نیازمندی‌ها پیشنهاد نمودند. این روش با شناسایی نیازمندی‌های مرتبط از مجموعه داده نیازمندی‌های نرم‌افزار، خطر از دست دادن نیازمندی‌ها در فرآیند استخراج نیازمندی‌ها را کاهش دهد. نتایج تجربی بر روی مجموعه داده RALIC [16] نشان می‌دهد که روش ارائه شده از نظر دقت پیش‌بینی، عملکرد بهتری نسبت به روش‌های پایه مبتنی بر پالایش مشارکتی دارد و مشکلات پراکندگی داده‌ها و اقلام جدید را برطرف می‌کند.

3. روش پیشنهادی

استفاده از سیستم‌های توصیه‌گر در مهندسی نیازمندی‌ها بر این فرض استوار است که ذینفعان به عنوان کاربران و نیازمندی‌ها به عنوان اقلام مورد توصیه در نظر گرفته می‌شوند. منطق استفاده از این سیستم‌ها به گونه‌ای است که اگر ذینفعی پیش از این نیازمندی‌هایی را انتخاب کرده باشد، احتمالا تمایل به انتخاب نیازمندی‌های مشابه نیز خواهد داشت. با این حال، در شرایطی که شباهت فقط بر اساس امتیازدهی ذینفعان سنجیده شود، مشکل زمانی پدید می‌آید که اطلاعات کافی درباره امتیازدهی ذینفعان موجود نباشد، یا نیازمندی یا ذینفع جدیدی به سیستم اضافه شود. در چنین مواردی، استفاده از شباهت‌های محتوایی میان نیازمندی‌ها برای تولید توصیه‌ها ضروری می‌شود. بنابراین، استفاده از روش‌های ترکیبی می‌تواند کارآمدتر باشد. در این مقاله، روشی ترکیبی از پالایش مشارکتی مبتنی بر آیتم و پالایش مبتنی بر محتوا ارائه شده است که در هر دو روش، ملاک اصلی شباهت بین نیازمندی‌ها است؛ با این تفاوت که در پالایش مشارکتی، شباهت آیتم‌ها بر اساس امتیازات ذینفعان محاسبه

[1] Sparsity
[2] Cold start

می‌شود، در حالی که در پالایش مبتنی بر محتوا، شباهت آیتم‌ها بر اساس ویژگی‌ها و محتوای آنها سنجیده می‌شود.

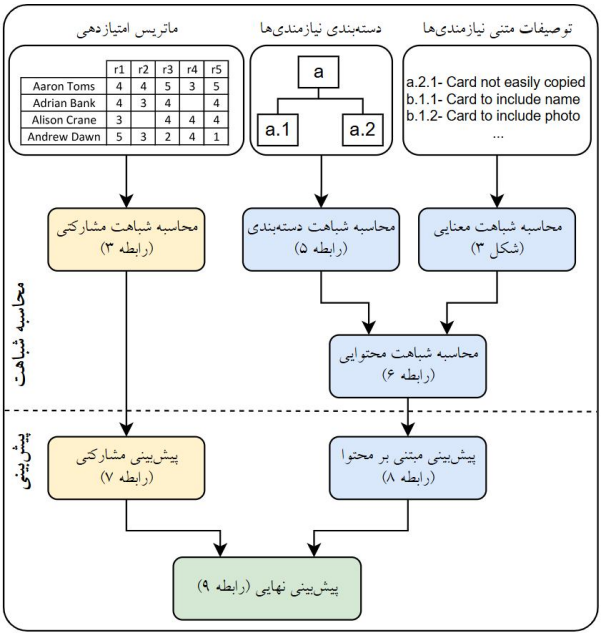
در این سیستم، مجموعه‌ای متشکل از n ذینفع $(s_1, s_2, \dots, s_n)$ و m نیازمندی $(r_1, r_2, \dots, r_m)$ تعریف شده‌اند. هر نیازمندی r_i دارای میزان مشخصی از اهمیت یا علاقه برای هر ذینفع s_i است که این مقدار با $rate(s_i, r_j)$ نمایش داده می‌شود. بدین ترتیب، یک ماتریس $m \times n$ تشکیل می‌شود که بیانگر درجه علاقه‌مندی یا نیاز هر ذینفع به هر نیازمندی است.

نیازمندی‌ها در یک ساختار سلسله‌مراتبی دو سطحی دسته‌بندی شده‌اند، به این معنا که هر دسته شامل چندین زیردسته می‌باشد و این دسته‌بندی‌ها به طور سلسله مراتبی سازماندهی شده‌اند. عضویت هر نیازمندی در یکی از این دسته‌ها و یکی از این زیردسته‌ها با استفاده از بردار C_{r_i} تعیین می‌شود. همچنین، هر نیازمندی با متنی کوتاه (کمتر از 100 کاراکتر) توصیف می‌گردد.

هدف اصلی از روش پیشنهادی، ارائه یک الگوریتم به منظور تولید فهرستی مرتب از نیازمندی‌ها بر اساس اولویت‌های هر ذینفع است. این روش به‌طور خاص باید قادر به پیش‌بینی میزان علاقه یا نیاز هر ذینفع s_i به هر نیازمندی r_j باشد. همان‌طور که در شکل (1) مشاهده می‌شود، در روش پیشنهادی در دو مرحله اصلی محاسبه شباهت بین نیازمندی‌ها و پیش‌بینی ترکیبی انجام می‌شود. لازم به ذکر است که بخش پیش‌بینی مشارکتی و روش محاسبه شباهت دسته‌بندی مطابق با روش مطرح شده در مرجع [15] بوده و سایر مراحل به عنوان نوآوری این پژوهش شناخته می‌شوند.

1.3. محاسبه شباهت بین نیازمندی‌ها

در پیش‌بینی ترکیبی روش‌های پالایش مشارکتی و مبتنی بر محتوا، هر یک از این روش‌ها به معیار خاصی برای سنجش شباهت نیاز دارند. معیاری که در پیش‌بینی پالایش مشارکتی بکار گرفته می‌شود را ColSim می‌نامیم. این معیار بر اساس میزان توافق ذینفعان در انتخاب دو نیازمندی و همچنین با در نظر گرفتن تعداد ذینفعانی که هر دو نیازمندی مورد نظر را برگزیده‌اند، محاسبه می‌گردد. در ادامه، جزئیات محاسبه این معیار را بررسی خواهیم کرد.



شکل (1): نمای کلی از روش پیشنهادی

بر اساس ماتریس امتیازدهی، از روش تفاضل میانگین مربعات[1] (MSD) برای سنجش میزان شباهت بین نیازمندی‌های r_i و r_j استفاده می‌شود. این سنجش با ارزیابی دقت پیش‌بینی نیازمندی r_i در صورتی که تنها از نیازمندی r_j برای پیش‌بینی استفاده شود عمل می‌کند. به عبارت دیگر، اگر یک پیشنهاد دقیق برای نیازمندی r_i با استفاده صرف از نیازمندی r_j به‌عنوان همسایه به دست آید، آنگاه نیازمندی‌های r_i و r_j باید نمره بالایی از شباهت ضمنی دریافت کنند. در این زمینه، روش پیش‌بینی معرفی شده در مقاله [17] برای محاسبه امتیاز پیش‌بینی شده یک نیازمندی بکار گرفته می‌شود. این روش تنها بر اساس استفاده از یک نیازمندی همسایه تعریف شده و به صورت رابطه (1) بیان می‌گردد.

$$p(s, r_i) = \overline{r_{ri}} + rate(s, r_j) - \overline{r_{rj}} \qquad (1)$$

که در آن، $\overline{r_{ri}}$ و $\overline{r_{rj}}$ به ترتیب میانگین امتیازات نیازمندی‌های r_i و r_j و $rate(s, r_j)$ امتیازی است که ذینفع s به نیازمندی r_j داده است. سپس، در رابطه (2)، از روش MSD برای اندازه‌گیری میزان شباهت ضمنی استفاده می‌شود. این اندازه‌گیری بر اساس تفاوت بین امتیازات پیش‌بینی شده و واقعی ذینفعان مشترکی که

[1] Mean Square Differences (MSD)

هر دو نیازمندی را امتیازدهی کرده‌اند، صورت می‌گیرد.

$$MSDSim(r_i, r_j) = 1 - \left(\frac{\sum_{s \in CS}(p(s, r_i) - rate(s, r_i))^2}{|CS|}\right) \quad (2)$$

در رابطه فوق، CS مجموعه ذینفعانی می‌باشد که هر دو نیازمندی r_i و r_j را انتخاب نموده‌اند. با این حال، معیار MSD ذکر شده، یک معیار شباهت مبتنی بر امتیازدهی است که تنها به امتیازدهی‌های ذینفعان مشترک توجه می‌کند. به این معنا که تعداد ذینفعان مشترک را که یک عامل مهم است، نادیده می‌گیرد. در یک معیار شباهت مبتنی بر امتیازدهی، شباهت میان نیازمندی‌هایی که بر اساس تعداد کمی از ذینفعان یا حتی یک ذینفع محاسبه می‌شود، به همان اندازه مهم در نظر گرفته می‌شود که شباهت محاسبه شده بر اساس تعداد بیشتری از ذینفعان مشترک. بنابراین، استفاده صرف از معیار MSD می‌تواند شباهت میان نیازمندی‌ها را نادرست نشان دهد و منجر به پیش‌بینی‌های نادرست شود.

برای رفع این مشکل، باید یک اندازه‌گیری شباهت ساختاری [18] که تعداد ذینفعان مشترک را هنگام محاسبه شباهت میان نیازمندی‌ها در نظر می‌گیرد، یکپارچه شود تا یک اندازه‌گیری شباهت بهبودیافته که دقت پیش‌بینی را افزایش می‌دهد، توسعه یابد. رابطه (3) نحوه محاسبه شباهت مشارکتی بین دو نیازمندی را نشان می‌دهد، که در آن S_{ri} و S_{rj} مجموعه ذینفعانی هستند که به ترتیب به r_i و r_j امتیاز داده‌اند.

$$ColSim(r_i, r_j) = \frac{|CS|}{\sqrt{|S_{ri}||S_{rj}|}} MSDSim(r_i, r_j) \quad (3)$$

در این بخش از روش که به محاسبه شباهت محتوایی ConSim بین نیازمندی‌ها می‌پردازیم، فرض بر این است که نیازمندی‌ها به صورت سلسله مراتبی در دو سطح دسته‌بندی شده‌اند و برای هر نیازمندی یک توصیف متنی موجود است. شکل (1) نمونه‌ای از 8 نیازمندی را در این قالب نمایش می‌دهد.

به طور کلی، نیازمندی‌هایی که در یک دسته قرار می‌گیرند، به میزان قابل توجهی با یکدیگر شباهت دارند. به عنوان مثال، در شکل (2)، نیازمندی‌های a.1.1 و a.2.2 هر دو در یک دسته جای گرفته‌اند؛ با این حال، نیازمندی‌هایی که علاوه بر دسته‌بندی اصلی، در یک زیردسته مشترک نیز قرار دارند، باید با شباهت بیشتری در نظر گرفته شوند. به این ترتیب، شباهت میان نیازمندی a.1.1 و a.1.2 بیشتر از شباهت میان a.1.1 و a.2.2 خواهد بود.

a- Security		
a.1- Access Control		
a.1.1- Control access to departments		
a.1.2- Control access to departmental offices		
a.2- Card Security		
a.2.1- Card not easily copied		
a.2.2- Card should have multiple security features		
b- Interface		
b.1- User Information Display		
b.1.1- Card to include name		
b.1.2- Card to include photo		
b.2- Access Rights Management		
b.2.1- View and modify access rights, time of access, online, without card being present		
b.2.2- Activate and inactivate card		

شکل (2): نمونه‌ای از نیازمندی‌های طبقه‌بندی شده

برای محاسبه شباهت نیازمندی‌هایی که در یک دسته یا زیردسته قرار دارند، از شباهت کسینوسی مبتنی بر بردار استفاده می‌شود. در این روش، هر نیازمندی به صورت برداری از مقادیر باینری نمایش داده می‌شود.

$$V_r = (V_{r,Cat1}, V_{r,Cat2}, \ldots, V_{r,SubCat1}, V_{r,SubCat2}, \ldots) \quad (4)$$

همان‌طور که در رابطه (4) نشان داده شده است، بردار مربوط به نیازمندی r شامل مجموعه‌ای از بیت‌ها است که در صورت تعلق نیازمندی به یک دسته یا زیردسته خاص، مقدار آن بیت برابر با 1 و در غیر این صورت برابر با 0 در نظر گرفته می‌شود. طول این بردار برابر با مجموع تعداد دسته‌ها و زیردسته‌ها است ($|V|$). رابطه (5) نحوه محاسبه شباهت کسینوسی دو نیازمندی بر اساس دسته‌بندی را نشان می‌دهد.

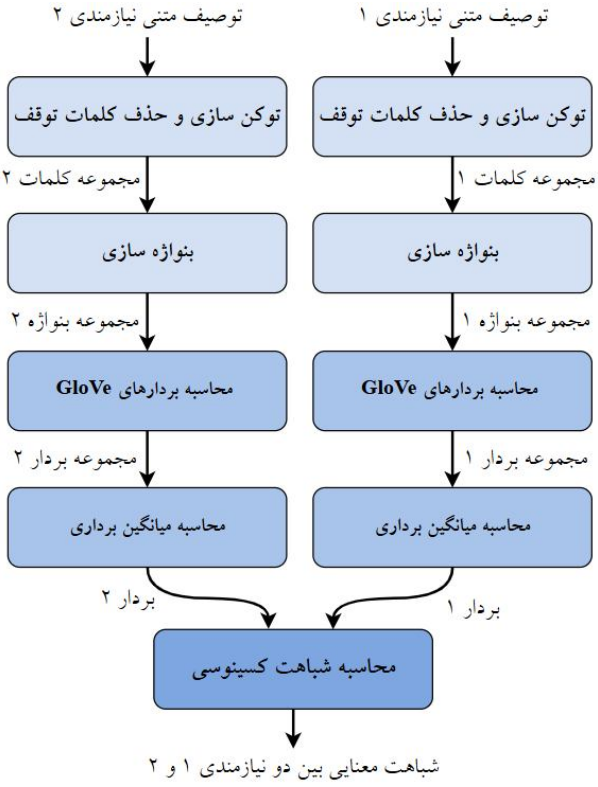
$$CatSim(r_i, r_j) = \frac{V_{ri} . V_{rj}}{\|V_{ri}\| . \|V_{rj}\|} \quad (5)$$

نقطه ضعف این معیار شباهت، که در منبع [15] نیز بکار رفته است، این است که شباهت محتوایی نیازمندی‌هایی که در دسته‌های متفاوت قرار دارند، صفر در نظر گرفته می‌شود. برای

مثال، مقدار شباهت CatSim(a.2.2, b.2.2) برابر با صفر است، با وجود اینکه هر دو نیازمندی به کارت‌ها مربوط هستند و ممکن است علاقه‌مندی یک ذینفع به یکی از آنها، علاقه‌مندی به دیگری را نیز به دنبال داشته باشد.

در روش پیشنهادی برای بهبود معیارهای شباهت مبتنی بر محتوا، از مدل GloVe به دلیل قابلیت آن در بهره‌گیری از توضیحات متنی مرتبط با نیازمندی‌ها استفاده شده است. GloVe با تحلیل آماری واژگان در مجموعه‌های بزرگ متنی، روابط معنایی میان کلمات را استخراج کرده و نمایش‌های برداری متراکم‌تری از نیازمندی‌ها ایجاد می‌کند. این نمایش‌های برداری امکان اندازه‌گیری دقیق‌تر شباهت‌ها را فراهم می‌آورد. استفاده از توضیحات متنی به این دلیل اهمیت دارد که این متون اطلاعات غنی‌تری از معنای هر نیازمندی ارائه می‌دهند و مدل GloVe با بهره‌گیری از این اطلاعات، قادر است ارتباطات معنایی دقیق‌تری را شناسایی کند، که در نهایت موجب افزایش دقت فرآیند شباهت‌سنجی می‌گردد.

شکل (3) نحوه محاسبه شباهت معنایی بین نیازمندی‌ها را نشان می‌دهد. ابتدا عملیات پیش‌پردازش به منظور بهبود کیفیت و دقت نتایج انجام می‌گیرد. این عملیات شامل چندین مرحله اساسی است. ابتدا تمامی حروف متن به حروف کوچک تبدیل می‌شوند و علائم نگارشی برای کاهش نویز حذف می‌گردند. سپس، متن به کلمات مجزا (توکن‌ها) شکسته می‌شود تا پردازش به شکل دقیق‌تری صورت گیرد. در ادامه، واژگان غیرضروری و پرتکرار که به عنوان واژگان پالایشی [1] شناخته می‌شوند (مانند to، at و the) از متن حذف می‌شوند تا تمرکز بر روی کلمات معنادار و کلیدی باقی بماند. همچنین، از تکنیک بنواژه‌سازی [2] برای تبدیل کلمات به شکل پایه یا ریشه آنها استفاده می‌گردد؛ به عنوان مثال، کلماتی مانند accesses، accessing و accessed همگی به شکل ساده access تبدیل می‌شوند. این مراحل پیش‌پردازش موجب یکپارچه‌سازی و ساده‌سازی داده‌های متنی می‌شود و به این ترتیب دقت اندازه‌گیری‌های شباهت معنایی بهبود می‌یابد.



شکل (3): نحوه محاسبه شباهت معنایی بین نیازمندی‌ها

در مرحله بعد، بردارهای GloVe برای هر بنواژه محاسبه می‌گردد و سپس میانگین این بردارها برای هر نیازمندی به دست می‌آید. در ادامه، با بهره‌گیری از شباهت کسینوسی، میزان شباهت معنایی (SemSim) بین دو نیازمندی تعیین می‌شود.

$$ConSim(r_i, r_j) = \begin{cases} CatSim(r_i, r_j) & \text{if in the same category} \\ SemSim(r_i, r_j) & \text{otherwise} \end{cases} \quad (6)$$

بر اساس آنچه در رابطه (6) ارائه شده است، تعیین شباهت محتوایی نهایی بین دو نیازمندی به چگونگی دسته‌بندی آنها بستگی دارد. به این معنا که اگر دو نیازمندی در یک دسته مشترک قرار داشته باشند، شباهت محتوایی نهایی بر مبنای شباهت دسته‌بندی (CatSim) آنها ارزیابی می‌شود. در مقابل، در صورتی که هیچ شباهتی از نظر دسته‌بندی میان نیازمندی‌ها وجود نداشته باشد، معیار سنجش شباهت محتوایی نهایی به شباهت معنایی (SemSim) آنها بر اساس توصیفات متنی مربوطه وابسته خواهد بود.

[1] Stop word
[2] Lemmatization

2.3. پیش‌بینی امتیاز به دو روش و ترکیب آنها

در مرحله پیش‌بینی، ابتدا از روش پالایش مشارکتی استفاده می‌شود که در آن معیار شباهت مشارکتی (ColSim) به کار گرفته می‌شود تا امتیاز مربوط به یک ذینفع خاص نسبت به یک نیازمندی معین پیش‌بینی گردد. سپس، همین فرآیند با استفاده از روش پالایش مبتنی بر محتوا، که بر مبنای شباهت محتوایی (ConSim) عمل می‌کند، تکرار خواهد شد. در مرحله نهایی، با اعمال یک ضریب وزنی مناسب، نتایج حاصل از هر دو روش ترکیب شده و به صورت یکپارچه ارائه می‌شوند.

روابط (7) و (8) به ترتیب نحوه پیش‌بینی با استفاده از روش‌های پالایش مشارکتی و مبتنی بر محتوا را نشان می‌دهند. در این روابط، N نمایانگر مجموعه‌ای از نیازمندی‌هایی است که بیشترین شباهت را به نیازمندی r دارند.

$$ColPredict(s,r) = \frac{\sum_{r_i \in N} ColSim(r,r_i)\, rate(s,\ r_i)}{\left|\sum_{r_i \in N} ColSim(r,r_i)\right|} \quad (7)$$

$$ConPredict(s,r) = \frac{\sum_{r_i \in N} ConSim(r,r_i)\, rate(s,\ r_i)}{\left|\sum_{r_i \in N} ConSim(r,r_i)\right|} \quad (8)$$

در نهایت، رابطه (9) پیش‌بینی‌های به دست آمده را ترکیب می‌نماید. در این رابطه، α و $(1-\alpha)$ مقداری حقیقی در بازه $[0,1]$ هستند که به ترتیب نشان‌دهنده اهمیت نسبی روش پالایش مشارکتی و روش مبتنی بر محتوا می‌باشند.

$$Predict(s,r) = \alpha\, ColPredict(s,r) + (1-\alpha)\, ConPredict(s,r) \quad (9)$$

4. ارزیابی و تحلیل نتایج

این بخش به ارزیابی روش پیشنهادی می‌پردازد که در آن ابتدا، مجموعه داده جهت مقایسه معرفی می‌شود. سپس، مجموعه روش‌های موجود جهت مقایسه معرفی و مقادیر پارامترهای مربوط به روش پیشنهادی ارائه می‌گردد. در نهایت، تحلیل نتایج در دو بخش شامل مقابله با پراکندگی داده‌ها و مساله شروع سرد صورت می‌پذیرد.

برای ارزیابی روش پیشنهادی، از مجموعه داده RALIC استفاده شده است و نتایج آن با دو روش موجود شامل روش کلاسیک پالایش مشارکتی مبتنی بر آیتم (ICF) [19] و روش ترکیبی (HCBCF) معرفی شده در مرجع [15] مقایسه گردیده است. مجموعه داده RALIC متشکل از 3113 امتیاز است که توسط 76 ذینفع برای 104 نیازمندی مختلف ارائه شده‌اند. این امتیازات با استفاده از یک مقیاس عددی از 1- تا 5 محاسبه شده و در یک ساختار سلسله مراتبی دو سطحی سازمان‌دهی شده‌اند. در این ساختار، اهداف کلی پروژه به عنوان دسته‌بندی‌های اصلی و نیازمندی‌های خاص به عنوان زیرمجموعه‌ها تعریف شده‌اند. علاوه بر اطلاعات ذکر شده، برای هر یک از نیازمندی‌های موجود در مجموعه داده RALIC، یک توصیف متنی نیز فراهم شده است. این توصیفات متنی جزئیات بیشتری درباره ماهیت و ویژگی‌های هر نیازمندی ارائه می‌دهند و به ذینفعان و تحلیل‌گران کمک می‌کنند تا درک عمیق‌تری از الزامات و اهداف مرتبط با هر نیازمندی کسب کنند.

جذر میانگین مربعات خطا[3] (RMSE) و میانگین قدر مطلق خطا[4] (MAE) دو شاخص اساسی در ارزیابی دقت مدل‌های پیش‌بینی و سیستم‌های توصیه‌گر هستند. RMSE به محاسبه میزان انحراف مقادیر پیش‌بینی شده از مقادیر واقعی با تاکید بر خطاهای بزرگ‌تر می‌پردازد، زیرا با مجذور کردن تفاضل، تاثیر خطاهای بزرگ را بیشتر می‌کند. در مقابل، MAE به محاسبه میانگین مقادیر قدر مطلق اختلافات میان مقادیر پیش‌بینی شده و واقعی می‌پردازد و به عنوان یک معیار ساده، به تمامی خطاها وزن مساوی می‌دهد. استفاده از MAE به ویژه زمانی مفید است که هدف ارزیابی دقت کلی مدل بدون در نظر گرفتن شدت خطاهای بزرگ باشد.

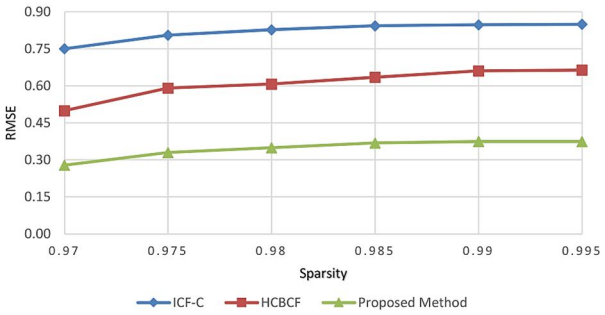
علاوه بر دو شاخص مرتبط با دقت، برای مقایسه روش پیشنهادی و روش‌های موجود، از شاخص پوشش [20]، نیز استفاده شده است. این شاخص درصد پیش‌بینی‌های انجام شده

[3] Root Mean Square Error (RMSE)
[4] Mean Absolute Error (MAE)

را نسبت به کل پیش‌بینی‌هایی که باید انجام می‌شد بیان می‌کند. شاخص پوشش ضعف روش‌ها در عدم توانایی تولید پیش‌بینی در شرایط خاص، مانند نیافتن تعداد کافی همسایه‌ها یا ناتوانی در محاسبه شباهت‌های لازم برای پیش‌بینی نهایی را آشکار می‌سازد.

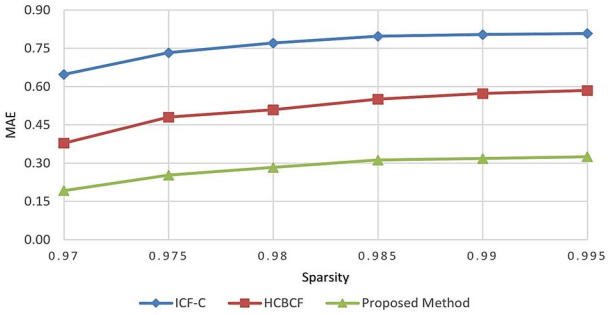
در روش پیشنهادی، دو پارامتر کلیدی باید تنظیم شوند: تعداد نیازمندی‌های مشابه یا همسایگان، که برای آن مقدار 4 اختصاص داده شده است و ضریب اهمیت α، که مقدار 0/5 برای آن در نظر گرفته شده است.

در بخش اول ارزیابی، 6 مجموعه داده با سطوح مختلف پراکندگی مورد بررسی قرار گرفته‌اند. این سطوح پراکندگی از کمترین میزان 97 درصد تا بیشترین میزان 99/5 درصد متغیر می‌باشند. همان‌طور که در شکل‌های (4) و (5) قابل مشاهده است، روش پیشنهادی در مقایسه با روش HCBCF در شاخص MAE، در شرایط با کمترین میزان پراکندگی، بهبود قابل توجهی را نشان می‌دهد و عملکرد بهتری با اختلاف 0/19 دارد. این اختلاف در شرایط با بیشترین میزان پراکندگی نیز افزایش یافته و به 0/26 می‌رسد. همچنین، شاخص RMSE نیز بهبود معناداری در عملکرد نشان می‌دهد، به طوری که در کمترین میزان پراکندگی، این اختلاف به 0/22 و در بیشترین میزان پراکندگی به 0/29 افزایش می‌یابد.

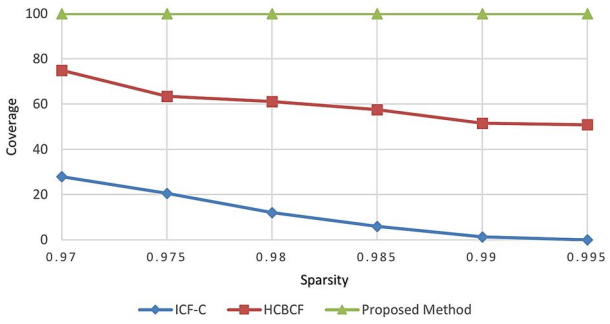


شکل (4): مقایسه بر اساس RMSE در سطوح مختلف پراکندگی

شکل (6) به مقایسه روش‌های مختلف بر اساس شاخص پوشش در سطوح مختلف پراکندگی می‌پردازد. در این مقایسه، روش HCBCF در کمترین پراکندگی توانسته است به 75 درصد و در بیشترین پراکندگی به 51 درصد از پیش‌بینی‌ها پاسخ دهد. این بدان معناست که این روش، در بهترین حالت، از هر 100 امتیاز مفقود، توانسته برای 75 امتیاز پیش‌بینی ارائه کند. در مقابل، روش پیشنهادی در تمامی سطوح پراکندگی، پوششی کامل و صد درصدی داشته و برای تمامی امتیازات مفقود پیش‌بینی ارائه می‌دهد.



شکل (5): مقایسه بر اساس MAE در سطوح مختلف پراکندگی

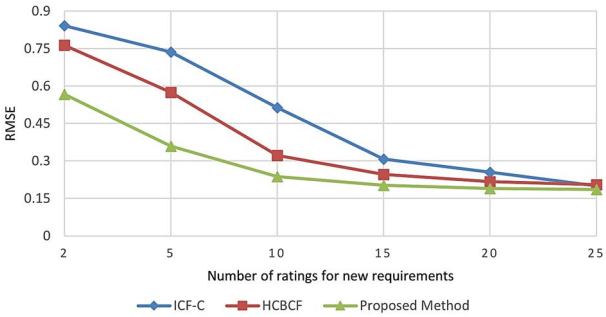


شکل (6): مقایسه بر اساس پوشش در سطوح مختلف پراکندگی

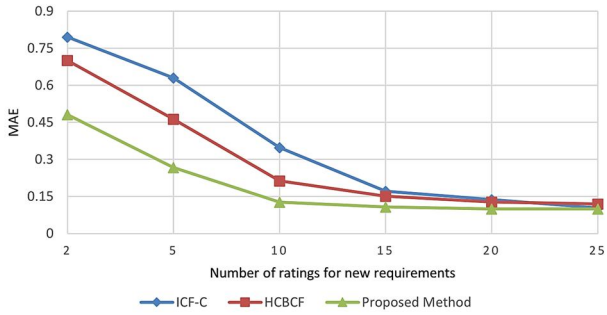
در بخش دوم ارزیابی، 6 مجموعه‌داده با سطوح مختلف شروع سرد مورد بررسی قرار گرفته‌اند. تعداد امتیازات موجود برای نیازمندی‌های جدید از حداقل 2 تا حداکثر 25 متغیر می‌باشد. به طور کلی، هر چه تعداد امتیازات موجود برای یک نیازمندی کمتر باشد، مساله شروع سرد شدیدتر بوده و پیش‌بینی‌ها دشوارتر می‌شوند.

نتایج به دست آمده در شکل‌های (7) و (8) نشان می‌دهد که روش پیشنهادی در مقایسه با روش HCBCF، بهبود قابل‌توجهی در شاخص MAE دارد. در شرایط شروع سرد با شدت کم (حداقل 25 امتیاز برای هر نیازمندی)، این روش با اختلاف جزئی 0/02 واحد بهتر عمل می‌کند. با افزایش شدت شروع سرد (حداقل 2 امتیاز برای هر نیازمندی)، این اختلاف به 0/21 می‌رسد و بهبود چشم‌گیری در عملکرد مشاهده می‌شود. در

شاخص RMSE نیز بهبود معناداری دیده می‌شود؛ به طوری که در شدت کم شروع سرد، اختلاف به 0/01 و در شرایط با شدت بالای شروع سرد به 0/19 می‌رسد. این نتایج نشان‌دهنده برتری روش پیشنهادی در شرایط مختلف شدت شروع سرد است و حاکی از کارایی بالای آن در مقایسه با روش HCBCF در پیش‌بینی دقیق‌تر است.



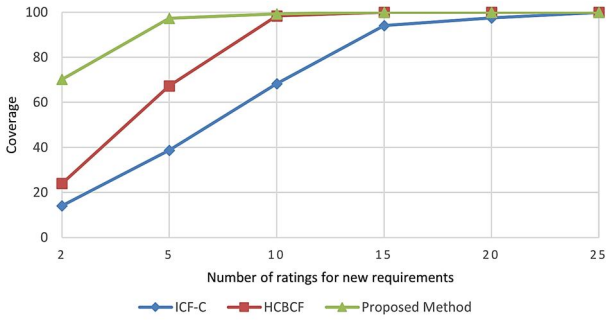
شکل (7): مقایسه بر اساس RMSE در سطوح مختلف شروع سرد



شکل (8): مقایسه بر اساس MAE در سطوح مختلف شروع سرد

همان‌طور که در شکل (9) مشاهده می‌شود در بیشترین شدت شروع سرد (حداقل 2 امتیاز برای هر نیازمندی) روش HCBCF تنها توانسته به 24 درصد از امتیازهای مفقود پاسخ دهد، در حالی که روش پیشنهادی به 70 درصد از پیش‌بینی‌ها دست یافته است. با افزایش تعداد حداقل امتیازها به 10، روش HCBCF به 98 درصد و روش پیشنهادی به 99 درصد از امتیازات مفقود پاسخ داده‌اند. در این سطح، تفاوت بین روش HCBCF و روش پیشنهادی بسیار ناچیز است. در مواردی که شروع سرد شدت کمتری دارد (حداقل 15، 20 و 25 امتیاز) هر سه روش قادر به پیش‌بینی برای تمامی امتیازات مفقود بوده و پوشش کامل صد درصدی را دارند.

به طور کلی، می‌توان نتیجه گرفت که روش پیشنهادی در تمامی سطوح پراکندگی بالاترین میزان پوشش را ارائه داده و در سطوح پایین پراکندگی نیز عملکرد بهتری نسبت به روش‌های دیگر از خود نشان داده است.



شکل (9): مقایسه بر اساس پوشش در سطوح مختلف شروع سرد

5. نتیجه‌گیری

مقاله حاضر به معرفی و ارائه روشی نوین در مدیریت نیازمندی‌های نرم‌افزاری می‌پردازد که به طور خاص به کاهش ریسک نادیده گرفتن نیازمندی‌ها در فرآیند استخراج نیازمندی‌ها کمک می‌کند. این روش، با ترکیب مزایای دو رویکرد پالایش مشارکتی و پالایش مبتنی بر محتوا و رفع محدودیت‌های موجود در هر یک از این رویکردها، عملکردی بهینه ارائه می‌دهد. در رویکرد پالایش مشارکتی، شباهت بین نیازمندی‌ها با استفاده از تعداد کاربران مشترک در فرآیند امتیازدهی ارزیابی می‌شود. از سوی دیگر، روش پالایش مبتنی بر محتوا با بهره‌گیری از اطلاعات طبقه‌بندی نیازمندی‌ها و همچنین شباهت معنایی استخراج شده توسط مدل GloVe، در مواقع کمبود داده‌های امتیازی، دقت توصیه‌ها را با کاهش اثرات پراکندگی داده‌ها بهبود می‌بخشد. نتایج حاصل از آزمایش‌های صورت گرفته بر روی مجموعه داده RALIC نشان می‌دهد که روش پیشنهادی از نظر دقت و پوشش پیش‌بینی، به طور معناداری برتر از روش‌های موجود است و نسبت به پراکندگی داده‌ها و همچنین شروع سرد مقاومت بیشتری دارد. به منظور بهبود اعتبار روش پیشنهادی، در آینده تحقیقات بیشتری برای ارزیابی و مقایسه آن با سایر روش‌های توصیه‌گر در مجموعه داده‌های بزرگ‌تر انجام

خواهد شد. همچنین، به منظور ارتقاء دقت روش، استفاده از مدل‌های دقیق‌تری برای محاسبه شباهت معنایی بین نیازمندی‌ها در دستور کار قرار دارد.

تعارض منافع: نویسندگان اعلام می‌کنند که هیچ تعارض منافعی ندارند.

مراجع

[1] R. S. Pressman, Software Engineering: A Practitioner's Approach, 6th ed. New York, NY, USA: McGraw-Hill, 2005.

[2] C. Palomares, C. Quer, and X. Franch, "Requirements Reuse and Requirement Patterns: A State of the Practice Survey," Empir. Softw. Eng., vol. 22, no. 6, pp. 2719-2762, 2017, doi: 10.1007/s10664-016-9485-x.

[3] C. C. Aggarwal, Recommender Systems. Cham, Switzerland: Springer International Publishing, 2016, doi: 10.1007/978-3-319-29659-3.

[4] F. Zarrinkalam and M. Kahani, "Using Semantic Relations to Improve Quality of a Citation Recommendation System," Soft Comput. J., vol. 1, no. 2, pp. 36-45, 2013, dor: 20.1001.1.23223707.1391.1.2.61.0 [In Persian].

[5] M. Zamzame, S. Sedighian Kashi, and A. Nikanjam, "Energy-Aware Evolutionary Multi-Objective Refactoring for Bad Code Smells Correction of Android Applications," Soft Comput. J., vol. 12, no. 2, pp. 78-95, 2024, doi: 10.22052/scj.2023.246479.1074 [In Persian].

[6] N. Moosarrezayi and J. Hamidzadeh, "Design a Hybrid Recommender System Solving Cold-Start Problem Using Clustering and Chaotic PSO Algorithm," Soft Comput. J., vol. 7, no. 1, pp. 50-61, 2018, doi: 10.22052/7.1.50 [In Persian].

[7] N. Hariri, C. Castro-Herrera, J. Cleland-Huang, and B. Mobasher, "Recommendation Systems in Requirements Discovery," in Recommendation Systems in Software Engineering, Berlin, Germany: Springer, 2014, pp. 455-476, doi: 10.1007/978-3-642-45135-5_17.

[8] H. Dumitru et al., "On-Demand Feature Recommendations Derived from Mining Public Product Descriptions," in Proc. 33rd Int. Conf. Softw. Eng. (ICSE), 2011, pp. 181-190, doi: 10.1145/1985793.1985819.

[9] G. Ninaus et al., "INTELLIREQ: Intelligent Techniques for Software Requirements Engineering," in Proc. 21st Eur. Conf. Artif. Intell. (ECAI), 2014, pp. 1161-1166, doi: 10.3233/978-1-61499-419-0-1161.

[10] S. Ahmad and M. Sadiq, "Recommender Systems for Software Requirements Negotiation and Prioritization," Int. J. Comput. Appl., vol. 117, no. 13, pp. 27-33, 2015, doi: 10.5120/20611-3261.

[11] M. I. Lunarejo, "Requirements Prioritization Based on Multiple Criteria Using Artificial Intelligence Techniques," in Proc. IEEE 29th Int. Req. Eng. Conf. (RE), 2021, pp. 480-485, doi: 10.1109/RE51729.2021.00072.

[12] S. AlZu'bi, B. Hawashin, M. ElBes, and M. Al-Ayyoub, "A Novel Recommender System Based on Apriori Algorithm for Requirements Engineering," in Proc. 5th Int. Conf. Soc. Netw. Anal. Manag. Secur. (SNAMS), 2018, pp. 323-327, doi: 10.1109/SNAMS.2018.8554909.

[13] S. S. Tanveer and Z. A. Rana, "Prioritizing Software Requirements by Combining the Usage Monitoring and User Feedback Data," IEEE Access, vol. 12, pp. 82825-82841, 2024, doi: 10.1109/ACCESS.2024.3409847.

[14] M. Muhairat, S. AlZu'bi, B. Hawashin, M. W. Elbes, and M. Al-Ayyoub, "An Intelligent Recommender System Based on Association Rule Analysis for Requirement Engineering," J. Univers. Comput. Sci., vol. 26, no. 1, pp. 33-49, 2020, doi: 10.3897/jucs.2020.003.

[15] Q. Y. Shambour, A. H. Hussein, Q. M. Kharma, and M. M. Abualhaj, "Effective Hybrid Content-Based Collaborative Filtering Approach for Requirements Engineering," Comput. Syst. Sci. Eng., vol. 40, no. 1, pp. 247-261, 2022, doi: 10.32604/csse.2022.017221.

[16] S. L. Lim, "Social Networks and Collaborative Filtering for Large-Scale Requirements Elicitation," Ph.D. dissertation, School Comput. Sci. Eng., UNSW Sydney, Sydney, Australia, 2010.

[17] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “GroupLens: An Open Architecture for Collaborative Filtering of Netnews,” in Proc. ACM Conf. Comput. Support. Coop. Work (CSCW), 1994, pp. 175-186, doi: 10.1145/192844.192905.

[18] D. Wang, Y. Yih, and M. Ventresca, “Improving Neighbor-Based Collaborative Filtering by Using a Hybrid Similarity Measurement,” Expert Syst. Appl., vol. 160, p. 113651, 2020, doi: 10.1016/j.eswa.2020.113651.

[19] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-Based Collaborative Filtering Recommendation Algorithms,” in Proc. 10th Int. Conf. World Wide Web (WWW), 2001, pp. 285-295, doi: 10.1145/371920.372071.

[20] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, “Recommendation Systems: Principles, Methods and Evaluation,” Egypt. Informatics J., vol. 16, no. 3, pp. 261-273, 2015, doi: 10.1016/j.eij.2015.06.005.