علوم رایانش و فناوری اطلاعات
نشریه علمی انجمن کامپیوتر ایران
مجلد ۱۵، شماره ۲، ۱۳۹۶
صفحات ۴۷- ۵۵
مقاله عادی

نشریه علوم رایانش و فناوری اطلاعات

سیستم پیشنهاددهنده قانون محور تکاملی

بهزاد سلیمانی نیسیانی[۱و۲] نسیم سلطانی[۳] رضا مفیدی[۳] محمد حسین ندیمی شهرکی[۴]

[۱]دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی واحد اصفهان (خوراسگان)، اصفهان، ایران
[۲]دانشکده برق و کامپیوتر، دانشگاه کاشان، کاشان، ایران
[۳]دانشکده فنی و مهندسی، موسسه آموزش عالی علامه نائینی، نائین، ایران
[۴]دانشکده کامپیوتر، دانشگاه آزاد اسلامی واحد نجف‌آباد، نجف‌آباد، ایران

چکیده

سیستم‌های پیشنهاددهنده برای ارائه محصولات به مشتریان بالقوه طراحی شده و همین طور فیلتر مشارکتی روشی رایج در سیستم‌های پیشنهاددهنده است. این سیستم با شناسایی کاربران مشابه براساس زمان ورود و انجام تراکنش‌های قبلی، پیشنهادهایی را تولید خواهد نمود. دقت پایین پیشنهادها به دلیل تنکی یکی از نگرانی‌های عمده مربوط به روش‌های پیشنهاددهنده فیلتر مشارکتی است، برای رفع این مشکل، محققان متعددی استفاده از استخراج قوانین انجمنی را ارئه داده‌اند. تولید قوانین انجمنی با سرعت بالاتر و حفظ کیفیت، می‌تواند باعث ارائه پیشنهادهای با کیفیت و سرعت بالاتر گردد. در این راستا، این مقاله سیستم پیشنهاددهنده مبتنی بر قوانین انجمنی تولید شده با استفاده از الگوریتم ژنتیک را ارائه می‌دهدکه نیاز به قوانین با رأس ثابت دارد و در نهایت قوانینی با رأس ثابت را تولید می‌کند. آزمایش‌های انجام شده روی مجموعه داده موویلنز، سرعت بالاتر تولید قوانین را نسبت به کارهای دیگر نشان می‌دهد و در مقایسه با الگوریتم ازدحام ذرات سرعت همگرایی بهتری در عین حفظ کیفیت داشته است.

کلمات کلیدی: سیستم پیشنهاددهنده، فیلتر مشارکتی، استخراج قوانین انجمنی، بهینه‌سازی ازدحام ذرات ، بهینه‌سازی چند هدفه.

۱- مقدمه

موفقیت کسب و کار آنلاین بستگی به توانایی ارائه محصولات شخصی، خدمات و اقلام[۱] اطلاعاتی به مشتریان بالقوه، دارد. این زمینه باعث تمایل به سیستم‌های پیشنهاددهنده شده است. این سیستم‌ها با اعمال روش‌های آماری و کشف دانش، خدماتی را به مشتریان ارائه می‌دهد [۱-۳]. سه نوع عمده از سیستم‌های پیشنهاددهنده عبارتند از:

۱-۱ مبتنی بر محتوا[۲] [۴]: محتوای اقلام را تحلیل نموده و سعی دارد نظم در میان آنها را درک نماید. به یک کاربر پیشنهادهایی خواهد شد در مورد شباهت زیاد آن اقلام به آن‌هایی که کاربر در گذشته ترجیح داده است.

۱-۲ مبتنی بر دانش[۳] [۵]: دانش در مورد کاربران، اقلام، چگونگی یک قلم خاص و نیازهای یک کاربر خاص را ثبت می‌کنند. پیشنهادات براساس این دانش صورت می‌گیرد.

۱-۳ فیلتر مشارکتی[۴] (CF) [۲، ۶]: از اطلاعات کاربران مشابه جهت پیشنهاد استفاده می‌کند. سیستم‌های پیشنهاددهنده مشارکتی[۵] اطلاعات متناسب با ترجیحات کاربر جمع‌آوری می‌نمایند. در فیلتر مشارکتی کاربران با ترجیحات مشابه در سایر موارد هم می‌توانند با هم شبیه باشند.

روش فیلتر مشارکتی دارای چالش‌های بسیاری است. از جمله برای مقابله با تنکی[۶] در پایگاه داده به منظور بهبود مقیاس‌پذیری و افزایش بهره‌وری پیشنهادات. تلاش‌های زیادی برای غلبه بر مشکلات فیلتر مشارکتی و در نتیجه تولید پیشنهادات با کیفیت شده است. این روش‌ها را می‌توان به این صورت دسته‌بندی نمود:

• روش‌های سنتی مبتنی بر همسایگی [۷]: سیستم‌های پیشنهاددهنده فیلتر مشارکتی مبتنی بر قلم پیش‌بینی را از طریق سابقه همسایگان اقلام محاسبه می‌کنند و انتخاب درست نزدیک‌ترین همسایگان که بیشترین شباهت را با قلم مورد نظر داشته باشند تاثیر زیادی بر افزایش دقت پیش‌بینی‌ها خواهد داشت. برای یافتن نزدیک‌ترین همسایگان باید شباهت بین تمام اقلام موجود در سیستم محاسبه شود.

• روش خوشه‌بندی [۸]: الگوریتم‌های خوشه‌بندی می‌توانند با خوشه‌بندی اقلام فضای جستجو را کاهش دهند و مقیاس‌پذیری سیستم را افزایش دهند. در نتیجه با محاسبه شباهت بین اعضایی که در یک خوشه قرار دارند، پیش‌بینی‌ها تولید خواهند شد.

• استدلال مبتنی بر مورد [۹]: استدلال مبتنی بر مورد، یک روش تصمیم‌گیری، مبتنی بر راه حل مسائل مشابه است. این روش توانایی یادگیری قدرتمندی دارد. می‌تواند از تجربیات گذشته، برای برخورد با مسائل جدید، استفاده کند. در وضعیت‌های متغیر و پویا، زمانی که شرایط ناشناخته و غیرواضح هستند،این روش استدلال مناسبی است.

• الگوریتم‌های ترکیبی [۱۰]: ترکیبی از CF با تکنیک‌های مبتنی بر حافظه و CF با تکنیک‌های مبتنی بر محتوا

• کاوش قوانین انجمنی (ARM[۷]) [۱۱، ۱۲]: برای تشخیص وابستگی‌های مهم میان اقلام موجود در یک پایگاه داده تراکنشی به کار می‌رود، به نحوی که حضور بعضی اقلام در تراکنش‌ها بر حضور برخی اقلام دیگر درهمان تراکنش‌ها دلالت داشته باشد.

تنکی اثر بخشی روش‌های فیلتر مشارکتی را محدود می‌کند. هنگامی که تراکنش داده‌ها نادر است، این سیستم در شناسایی شباهت‌ها در منافع مصرف کننده ناکارآمد است. اساساً، مشکل تنکی را می‌توان در دو بعد تجزیه و تحلیل نمود. CF مبتنی بر کاربر و CF مبتنی بر قلم. به دلیل تنکی در مجموعه داده، دو کاربر که علائق مشابه دارند اما قلم مشترکی را رای نداده‌اند، علائق دو کاربر مشابه در نظر گرفته نمی‌شود. در این شرایط، CF مبتنی بر کاربر پیشنهادهای با کیفیت پایین تولید می‌کند. مشابه این وضعیت برای سیستم‌های CF مبتنی بر قلم رخ می‌دهد، زمانی که دو قلم مشابه هستند ولی چون توسط یک کاربر امتیاز مشابه دریافت نکرده‌اند مشابه در نظر گرفته نمی‌شوند.

هانگ[۸] و همکاران یک روش برای برخورد با مشکل تنکی از طریق بررسی قوانین انجمنی در میان مصرف‌کنندگان از طریق تراکنش‌های گذشته و بازخورد، ارائه داده‌اند [۱۳]. بعدها، لئونگ[۹] و همکاران با معرفی یک چارچوب CF براساس قوانین انجمنی فازی و در مشابهت چند سطحی (FARAMS[۱۰]) به مشکل تنکی پرداخته‌اند [۱۴].

از جمله راه حل‌هایی که از طریق کشف قوانین انجمنی ارائه شده عبارتند از:

۱. الگوریتم کاوش قوانین انجمنی با ضریب حمایت وفقی (ASARM[۱۱]) [۱۲]، که در این الگوریتم تولید قانون با الگوریتم CBA-RG انجام شده که تکامل یافته الگوریتم اپریوری[۱۲] می‌باشد.

۲. الگوریتم کاوش قوانین انجمنی با استفاده از الگوریتم ژنتیک که از داده‌کاوی برای کشف الگوهای خرید مکرر مشتریان استفاده کرده است [۱۵]. در ابتدا مشتریان برگزیده و در نهایت قوانین انجمنی با استفاده از آنها اکتشاف می‌شود.

۳. الگوریتم اکتشاف قوانین انجمنی با استفاده از الگوریتم ژنتیک برای یافتن قوانین ارتباطی است که در این مقاله فرد به عنوان مجموعه‌ای از قوانین ارائه شده است. الگوریتم پیشنهادی یک جایگزین برای یافتن قوانین با پیش‌بینی دقیق و دشوار، قابل فهم و جالب است [۱۶].

۴. کاوش قوانین انجمنی با استفاده از ژنتیک و قانون ارتباط، که برای یافتن الگوهای مکرر استفاده شده است [۱۷]. برای کاهش پیچیدگی زمانی و جست‌وجوی سراسری ژنتیک و قانون ارتباط ترکیب شده تا اطلاعات مورد نیاز دریافت شوند.

۵. الگوریتم کشف قوانین انجمنی مبتنی بر بهینه‌سازی تجمعی ذرات چند هدفه (MOPSO[۱۳]) [۱۱]، که در این الگوریتم سعی شده تا ایرادات الگوریتم ASARM را برطرف نماید. این الگوریتم با ترکیب استخراج قوانین انجمنی (ARM) با الگوریتم بهینه‌سازی تجمعی ذرات چند هدفه (MOPSO) در چارچوب فیلتر مشارکتی به بهبود دقت و صحت پیشنهادات کمک نموده است.

در این مقاله شیوه به کار گرفته شده در الگوریتم MOPSO الگو برداری شده و با الگوریتم ژنتیک جایگزین شده است. سرعت همگرایی بالاتر الگوریتم ژنتیک نسبت به الگوریتم بهینه‌سازی ذرات تجمعی (PSO) می‌تواند سرعت تولید قوانین و پیشنهادها را بالا ببرد. بخش ادبیات تحقیق توضیحاتی در مورد استخراج قوانین انجمنی، الگوریتم ژنتیک و بهینه‌سازی چند هدفه آورده شده است. بخش سوم الگوریتم استخراج قوانین انجمنی مبتنی بر ژنتیک (GARM) ارائه شده است. بخش چهارم استراتژی پیشنهاد شرح داده شده است. بخش ۵ الگوریتم GARM با کارهای مشابه مقایسه شده و بخش پایانی در مورد کارهای آینده صحبت شده است.

۲- پیش‌زمینه

در این بخش، زمینه‌های کلی استخراج قوانین انجمنی و الگوریتم ژنتیک ارائه شده است. در قسمت سوم بهینه‌سازی چند هدفه به صورت خلاصه شرح داده شده است.

۲-۱- استخراج قوانین انجمنی (ARM)

استخراج قوانین انجمنی برای اولین بار توسط آگراوال[۱۴] معرفی شده است و یکی از مهم‌ترین تکنیک‌های داده‌کاوی است [۱۸]. هدف آن کشف قوانین انجمنی مطلوب از پایگاه داده‌های تراکنش تجاری است. بنابراین، ARM می‌تواند در تصمیم‌گیری کسب و کار به تصمیم‌گیرندگان در جهت خرید محصولات کمک کند.

از دیدگاه کلی کاوش قوانین انجمنی را می‌توان یک فرایند دو گامی در نظر گرفت. گام اول یافتن کلیه مجموعه اقلام مکرر و گام دوم تولید قوانین انجمنی قوی با کمک مجموعه اقلام مکرر. به دلیل هزینه بالای محاسباتی در گام اول، به طور معمول الگوریتم‌ها بر روی بهینه‌سازی عملیات این مرحله متمرکز می‌شوند، چون کارایی الگوریتم با توجه به پیچیدگی این مرحله سنجیده می‌شود.

قوانین به صورت $A \rightarrow B$ بوده که A دنباله[۱۵] قانون و B راس[۱۶] قانون می‌باشد. به این معناست که اگر A اتفاق بیفتد به احتمال زیاد B هم اتفاق خواهد افتاد. معیار قوی بودن قوانین انجمنی که در گام دوم مورد بررسی قرار می‌گیرد، براساس دو پارامتر ضریب حمایت[۱۷] و ضریب اطمینان[۱۸] سنجیده خواهد شد. این دو پارامتر به صورت زیر محاسبه می‌شود:

$$\sup(A \rightarrow B) = \frac{\text{تعداد تراکنش‌های شامل A و B}}{\text{تعداد تراکنش‌های مجموعه داده}} \quad (۱)$$

$$\text{Conf}(A \rightarrow B) = \frac{\text{تعداد تراکنش‌های شامل A و B}}{\text{تعداد تراکنش‌های شامل A}} \quad (۲)$$

مجموعه داده شامل m تراکنش روی n قلم داده می‌باشد. A و B مجموعه اقلام هستند، به طوری که $A \cap B = \emptyset$ همواره صادق است. قوانینی که ضریب

حمایت و ضریب اطمینان آنها بالاتر از حداقل ضریب حمایت و حداقل ضریب اطمینان تعیین شده توسط مساله باشد، قانون قوی نامیده می‌شوند.

الگوریتم اپریوری برای کشف قوانین انجمنی استفاده می‌شودکه توسط آگراول و همکاران طراحی شده است. این الگوریتم در نظر دارد برای پیدا کردن مجموعه اقلام مکرر از حداقل ضریب حمایت استفاده نموده و پس از آن قوانین انجمنی با کیفیت را با استفاده از حداقل ضریب اطمینان معین کند. این الگوریتم به عنوان پایه‌ای برای تجزیه و تحلیل سبد بازار به وسیله پیدا کردن ارتباط بین اقلام مختلف در سبد خرید، استفاده می‌شود.

۲-۲- الگوریتم ژنتیک

الگوریتم ژنتیک[۱۹] راه حل مناسبی برای جایگزینی روش‌های بهینه‌سازی سنتی، با استفاده از جستجوهای تصادفی در محیط‌های پیچیده است [۱۹، ۲۰]. الگوریتم ژنتیک نوع خاصی از الگوریتم‌های تکامل است که از تکنیک‌های زیست‌شناسی برگشتی مانند وراثت و جهش استفاده می‌کند، بدین صورت که هر نسل با پیشرفت دائم خود باعث بهبود و تکامل نسل بعدی می‌گردد که در الگوریتم ژنتیک با استفاده از جهش و وراثت اینکار صورت می‌گیرد. این الگوریتم برای اولین بار توسط جان هلند معرفی شد.

ایده اساسی این الگوریتم انتقال خصوصیات موروثی توسط ژن‌هاست. مجموعه خصوصیات انسان توسط کروموزوم‌های او به نسل بعدی منتقل می‌شوند [۲۱، ۲۲]. هر ژن در این کروموزوم‌ها نماینده یک خصوصیت است. به عنوان مثال ژن ۱ می‌تواند رنگ چشم باشد، ژن ۲ طول قد، ژن ۳ رنگ مو و الی آخر. کروموزم‌ها توسط عملیات ترکیب[۲۰] و جهش[۲۱] باز تولید شده و نسل جدید را بوجود می‌آورند.

۲-۳- بهینه‌سازی چند هدفه

الگوریتم‌های بهینه‌سازی چند هدفه به صورت بهینه‌سازی $\{f_1(p), f_2(p), \dots, f_l(p)\}$ طراحی شده که $p \in P$ (منطقه امکان‌پذیر) بوده و در آن $f_1, f_2, \dots, f_l$ توابع هدف مختلف هستند. این توابع به طور همزمان بهینه‌سازی می‌شوند. اغلب توابع هدف متضاد هستند، بنابراین هیچ راه‌حل واحد به طور همزمان نمی‌تواند به بهینه‌سازی تمام توابع منجر شود. مجموعه‌ای از راه‌حل‌ها برای مشکل بهینه‌سازی چند هدفه وجود دارد [۲۳، ۲۴].

در اینجا مقادیر ضریب اطمینان و ضریب حمایت به عنوان دو تابع هدف برای جستجو راه‌حل بهینه (قوانین) به کار گرفته شده است. یک قانون بهینه گفته می‌شود اگر دارای کیفیت بالا باشد. کیفیت قوانین با در نظر گرفتن متوسط وزنی مقادیر ضریب اطمینان و ضریب حمایت قانون ارزیابی می‌گردد. بنابراین، برای یک قانون انجمنی $A \rightarrow B$، کیفیت قانون به صورت زیر تعریف می‌گردد:

$$\text{quality}(A \rightarrow B) = W_1 \text{sup}(A \rightarrow B) + W_2 \text{conf}(A \rightarrow B) \quad (۳)$$

در این رابطه، W1 و W2 وزن اختصاص یافته به مقادیر ضریب اطمینان و ضریب حمایت بوده به طوریکه $W_1+W_2=1$ و $W_1, W_2 \in [0,1]$ است. W1 با ارزش بیشتر و W2 با ارزش کمتر نشان می‌دهد که کیفیت قانون بیشتر بستگی به مقدار ضریب حمایت قانون دارد. در مقابل، W1 با ارزش کمتر و W2 با ارزش بیشتر نشان می‌دهدکه کیفیت قانون بیشتر بستگی به مقدار ضریب اطمینان قانون دارد. این وزن‌ها به صورت تجربی انتخاب می‌گردند [۱۱].

۳- الگوریتم استخراج قوانین انجمنی مبتنی بر ژنتیک[۲۲] (GARM)

قوانین انجمنی تولید شده برای سیستم پیشنهاددهنده فیلتر مشارکتی به صورت قوانین با رأس ثابت است، بدین معنی که اگر کاربری به راس ثابت f وابستگی داشته باشد سیستم پیشنهاددهنده باید بتوانید تمام اقلام مرتب با f را بیابد و به مشتری پیشنهاد دهد.

به عنوان مثال چهار رأس a, b, c, d به رأس f منجر می‌شوند و اگر کاربر به رأس f وابستگی داشته سیستم پیشنهاددهنده باید بتواند سایر رأس‌هایی که منجر به f می‌شوند مثل a, b, c, d را به کاربر پیشنهاد دهند.

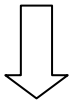
رأس ثابت می‌تواند قلم و یا کاربری باشد که فیلتر مشارکتی برای آن پیشنهاد تولید خواهد کرد. تا پایان مقاله به جای قلم/کاربر هدف که قرار است برای آنها پیشنهاد تولید شود از واژه شی هدف به جای هر دو استفاده شده است.

۳-۱- طرح نمایش[۲۳]

قانون انجمنی $A \rightarrow B$ دو مجموعه قلم غیرمتصل A و B است. در این مقاله سیستم پیشنهاددهنده، قوانین با رأس ثابت نیاز دارد، از این‌رو الگوریتم استخراج قوانین انجمنی مبتنی بر ژنتیک (GARM) به استخراج قوانین تنها برای یک شی هدف (کاربر/قلم) می‌پردازد.

برای کاربرد این الگوریتم، هر قانون به عنوان یک کروموزم ارائه می‌گردد. از آنجا که رأس قانون برای یک شی هدف خاص بوده و ثابت است، روی دنباله قانون عملیات انجام می‌گردد. بنابراین هر یک از کروموزم‌ها در فضای جستجو توسط $(A = \{A_1, A_2, \dots, A_d\})$ نشان داده می‌شود. به عنوان مثال قانون $I_5 \rightarrow I_8$ به صورت $8 \rightarrow 5$ در نظر گرفته شده است. چون رأس قانون (I_8) ثابت است، عملیات ژنتیک روی دنباله قانون (I_5) انجام می‌گیرد. عدد ۵ به صورت دودویی به صورت (۰۱۰۱) خواهد بود. هرکدام از بیت‌های صفر و یک تشکیل ژن‌های کروموزم را می‌دهند.

شماره کاربرها	شماره اقلام			
	I_1	I_2	I_3	I_4
User 1	3	0	1	2
User 2	0	1	4	0
User 3	0	0	0	2
User 4	1	5	2	4
User 5	4	0	1	2



شماره کاربرها	شماره اقلام			
	I_1	I_2	I_3	I_4
User 1	1	0	0	1
User 2	0	0	1	0
User 3	0	0	0	1
User 4	0	1	0	1
User 5	1	0	0	0

شکل ۱- طرح تبدیل

۳-۲- طرح تبدیل[۲۴]

به منظور استخراج قوانین انجمنی از مجموعه داده‌های تراکنشی، طرح انتقال مورد نیاز است که می‌تواند فرآیند مرور داده‌ها را سرعت بخشد [۲۵]. در شکل ۱ در جدول بالایی، پنج کاربر را نشان می‌دهد که به چهار قلم رای داده‌اند. آرا به صورت یک تا پنج داده شده است. در جدول پایینی طرح تبدیل یافته نمایش داده شده است. برای تبدیل، معدل رای‌های داده شده برای هر کاربر محاسبه شده و آرا کمتر از معدل صفر و بالاتر از معدل یک در نظر گرفته می‌شود.

۳-۳- طرح الگوریتم GARM

با توجه به طرح تبدیل گفته شده مجموعه داده مورد استفاده، به صورت تعدادی کاربر بوده که به اقلام رای داده‌اند. پس از تبدیل مجموعه داده به صورت پسند[۲۵] و یا ناپسند[۲۶] (صفر و یک) برای هر کاربر در می‌آید. سیستم پیشنهاددهنده برای چنین مجموعه داده‌ای دو حالت برای پیشنهاد دارد.

۱. پیشنهادها را براساس شباهت میان اقلام تولید می‌کند که به صورت آفلاین صورت می‌پذیرد. در این حالت قوانین انجمنی بر پایه قلم نیاز دارد. (مانند $I_5 \rightarrow I_8$)

۲. پیشنهادها را براساس شباهت میان کاربرها تولید می‌کند که به صورت آنلاین صورت می‌پذیرد. برای این حالت قوانین انجمنی بر پایه کاربر نیاز دارد. (مانند $U_7 \rightarrow U_8$)

همان‌طور که در طرح نمایش گفته شد. کروموزم‌ها به صورت عدد دودویی شی هدف خواهد بود. اگر کاربر ۱ در شکل ۱، کاربر هدف باشد. اقلام I_1 و I_4 قلم هدف در نظر گرفته می‌شود. در این شرایط الگوریتم قوانین با رأس I_1 و I_4 را تولید می‌کند. سیستم پیشنهاددهنده براساس این قوانین برای کاربر ۱ پیشنهاد تولید می‌کند. شبه کد این الگوریتم در شکل ۲ نشان داده شده است که در آن مجموعه داده تبدیل شده (دودویی) D_t و شیء هدف t به عنوان ورودی دریافت شده و k مورد برتر قوانین انجمنی مستقیم R_t به عنوان خروجی بر می‌گرداند.

GARM Algorithm:
Input: Trans Formed Dataset D_t, Target Object t
Output: R_t←Top-k Direct Association Rule in the Archive R_t
Step 1: Initial Population Randomly and set R_t
Step 2: for g=1 to q //q: Maximum Generation Length
Select Some Rules in R_t
Do Cross Over and Mutation
Evaluate Fitness by quality (A→B) =
W_1 sup (A→B) +W_2 conf (A→B)
Update R_t
End for
Step 3: R_t←Find Top-k rules in R_t
Step 4: send Top-k Rules to Recommendation System

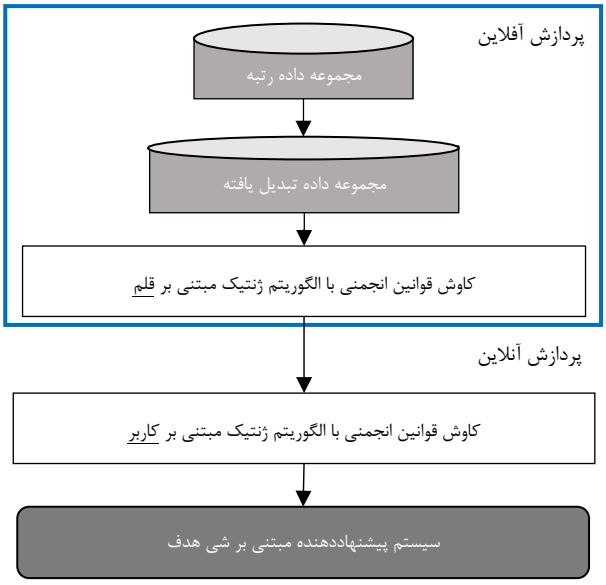
شکل ۲- شبه کد الگوریتم GRAM

- گام اول: برای نسل اول تعدادی کروموزوم به صورت تصادفی تولید می‌شوند.
- گام دوم: به تعداد مشخص q مرتبه این اعمال را انجام می‌دهد: کروموزم‌ها به صورت دو به دو با تابع انتخاب[۲۷] برای عملیات برش[۲۸] و جهش[۲۹] انتخاب می‌شوند؛ خروجی این عملیات، نسل جدید است که برای هر کدام از آنها تابع ارزیاب با توجه به مجموعه داده محاسبه می‌شود و بهترین آنها جایگزین نسل قبل خواهند شد. این روال ادامه می‌یابد تا زمانی که تعداد q مرتبه خاتمه یافته یا کیفیت قوانین در نسل‌های جدید بهبودی نیابد و در واقع الگوریتم به همگرایی برسد. کیفیت قوانین با رابطه ۳ سنجیده می‌شود.
- گام سوم: قوانین استخراج شده با کیفیت بالا در هر نسل به همراه کیفیت آنها در یک پایگاه داده ذخیره می‌شود.
- گام چهارم: از پایگاه داده، k قانون برتر از نظر کیفیت، انتخاب شده و به عنوان ورودی به سیستم پیشنهاددهنده ارسال می‌شوند.

همانطورکه گفته شد، این سیستم پیشنهاددهنده قوانین با رأس ثابت را نیاز دارد. به همین علت الگوریتم فوق قوانین با رأس شی هدف را استخراج می‌کند. هنگام اعمال ترکیب و جهش نیز تنها دنباله قانون تغییر می‌کند.

۴- استراتژی پیشنهاد

سیستم پیشنهاددهنده براساس قوانین انجمنی استخراج شده، پیشنهاد را تولید می‌کند. دو گونه قوانین انجمنی در بخش قبل گفته شد. قوانین بر پایه قلم به صورت آفلاین و قوانین بر پایه کاربر به صورت آنلاین، هر دو برای دقیق‌تر شدن پیشنهادها به کار گرفته می‌شوند. مدل پیشنهاد شده این مقاله در شکل ۳ نشان داده شده است.



شکل ۳- مدل سیستم پیشنهاددهنده

- سیستم پیشنهاددهنده بر پایه کاربر: ساختار قوانینی که برای سیستم پیشنهاددهنده بر پایه کاربر استفاده می‌شود به صورت $U_x \rightarrow U_t$ است. برای این قانون، اقلامی که کاربر U_t در آینده خواهد پسندید، در گذشته U_x پسندیده است. در این شرایط اقلامی که U_x پسندیده به کاربر U_t پیشنهاد می‌گردد [۱۱].
- سیستم پیشنهاددهنده بر پایه قلم: ساختار قوانینی که برای سیستم پیشنهاددهنده بر پایه قلم استفاده می‌شود به صورت $I_x \rightarrow I_t$ است. برای این قانون، کاربری که It را پسندیده در آینده Ix را خواهد پسندید. در این شرایط Ix پیشنهاد می‌گردد [۱۱].

۵- ارزیابی و تفسیر نتایج

در این قسمت سه آزمایش انجام شده است. آزمایش روی الگوریتم GARM انجام شده تا بهبود عملکرد سرعت این الگوریتم را نشان دهد. الگوریتم پیشنهادی با الگوریتم MOPSO-ARM مقایسه شده است [۱۱]. آزمایش‌ها روی مجموعه داده موویلنز 100k انجام شده است. در این بخش ابتدا مجموعه داده مورد استفاده شرح داده شده

است. سپس تاثیر پارامترهای مختلف روی الگوریتم GARM بررسی شده و در قسمت آخر مقایسه انجام شده است.

۵-۱- محیط ارزیابی

برای ارزیابی الگوریتم‌های GARM و MOPSO از کامپیوتر با مشخصات زیر استفاده شده است.

- RAM 2G
- CPU Doul-core E5400 @ 2.70 GHz
- ویندوز ۷
- نرم‌افزار متلب[۳۰]

از نرم‌افزار متلب به عنوان ابزار ارزیابی استفاده شده است. نرم‌افزار متلب به صورت تخصصی روی مجموعه داده‌های بزرگ کار می‌کند و کدهای خاص برای انجام عملیات روی ماتریس را دارد. این نرم‌افزار برای پیاده‌سازی انواع الگوریتم‌های تکاملی به کار گرفته می‌شود. به همین علت برای ارزیابی این پژوهش، مناسب تشخیص داده شده است.

۵-۲- مجموعه داده

مجموعه داده استفاده شده در آزمایش‌ها مویلنز[۳۱] 100k است. این مجموعه داده توسط پروژه تحقیقاتی گروه لنز از دانشگاه مینسوتا جمع‌آوری شده است [۲۶]. این مجموعه داده شامل ۱۰۰ هزار رکورد بوده که ۹۴۳ کاربر به ۱۶۸۲ فیلم رای داده‌اند. آرا در بازه ۱-۵ است. مشخصات دقیق مجموعه داده در جدول ۱ آمده است.

جدول ۱- مشخصات مجموعه داده

تعداد کاربر	۹۴۳
تعداد قلم	۱۶۸۲
بازه آرا	۱تا ۵
حداقل تعداد رای	۲۰
حداکثر تعداد رای	۷۳۷
نوع اعداد	صحیح

برای آزمایش‌ها تعداد مختلف کاربر هدف به تصادف نمونه برداری شده است. برای هر آزمایش از میان کاربرهای با تعداد رای کم (حدود ۲۰ رای)، با تعداد رای زیاد (حدود ۶۰۰ رای) و با حدود ۳۰۰ رای، به طور مساوی نمونه‌برداری شده است.

۵-۳- معیار ارزیابی و بررسی اثر پارامترها

میانگین کیفیت قوانین در هر تکرار[۳۲]، معیار مقایسه میان پارامترهای مختلف است. زمان اجرای الگوریتم در لحظه همگرایی معیار مقایسه نهایی خواهد بود. برای بررسی اثر پارامترها سه کاربر هدف به تصادف انتخاب شده است.

کیفیت قوانین انجمنی تولید شده توسط رابطه ۳ سنجیده می‌شود. در این رابطه w1=۰.۶ و w2=۰.۴ در نظر گرفته شده است [۱۱]. تعداد جمعیت اولیه و تابع انتخاب با همین معیار سنجیده می‌شود. سیستم پیشنهاددهنده براساس قوانین قوی پیشنهاد تولید می‌کند.

پارامتر کیفیت که برای سنجش قوانین به کار می‌رود، در رابطه ۴ معرفی شده است. این رابطه هر دو پارامتر ضریب حمایت و ضریب اطمینان را برای سنجش کیفیت در نظر می‌گیرد. تعدادی از قوانین با کیفیت برای سیستم پیشنهاددهنده استفاده می‌شود. اینکه چه تعداد از قوانین با کیفیت در نظر گرفته شود، مورد ارزیابی قرار می‌گیرد. ضرایب W1 و W2 پارامترهای کنترلی هستند که مورد ارزیابی قرار گرفته‌اند. این پارامترهای کنترلی توسط پارامترهای recall، precision، accuracy وF1-measure ارزیابی می‌شوند.

$$\text{accuracy} = \frac{\text{تعداد اقلامی که صحیح دسته‌بندی شده‌اند}}{\text{کل اقلام دسته‌بندی شده}} \quad (۴)$$

$$\text{precision} = \frac{\text{تعداد اقلام پیشنهاد شده صحیح}}{\text{کل اقلام پیشنهاد شده}} \quad (۵)$$

$$\text{recall} = \frac{\text{تعداد اقلام پیشنهاد شده صحیح}}{\text{کل اقلام پسندیده شده بوسیله کاربر}} \quad (۶)$$

$$F1_measure = \frac{2_*precision_*recall}{precision_+recall} \quad (۷)$$

۵-۴- تنظیمات ارزیابی

در این بخش ابتدا پارامترهای کنترلی W1 و W2 مورد بررسی قرار می‌گیرد. سپس تابع انتخاب برای الگوریتم ژنتیک بررسی می‌شود. سایر پارامترهای ارزیابی در جدول ۲ نشان داده شده است.

جدول ۲- پارامترهای ارزیابی

نوع برش	تک نقطه‌ای
نوع جهش	تصادفی
W1	۰.۶
W2	۰.۴

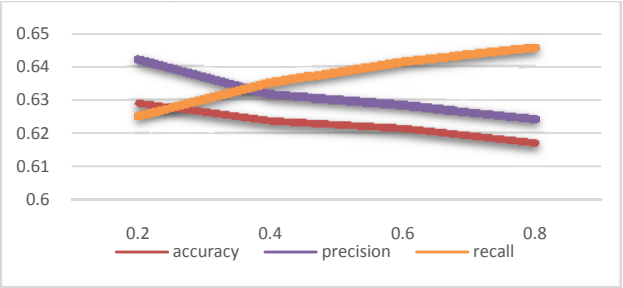
۵-۵- پیش آزمایش روی پارامترهای کنترلی W_1، W_2 و تعداد قوانین با کیفیت

در این قسمت ابتدا ضرایب w1 و w2 بررسی می‌شود. سپس تعداد قوانین با کیفیت بررسی می‌گردد. پارامترهای زیر برای سنجش کیفیت پیشنهادات به کار می‌رود. در شکل ۴ نشان می‌دهد رابطه ۳ با w2 های مختلف، پیشنهادات دقت متفاوتی را خواهند داشت (با توجه به آنکه مجموع w1 و w2 برابر ۱ است). در این شرایط هنگامی که w2 = ۰.۴ است، w1 = ۰.۶ خواهد بود. روی w2 =۰.۴ مقادیر هر سه پارامتر به یک توازنی می‌رسند. این شکل نشان می‌دهد به ازای مقادیر مختلف w2، کیفیت پارامترها خیلی تفاوت نمی‌کنند و حدود ۶۰ درصد هستند. اما روی مقدار ۰.۴ تقریبا متوازن‌تر هستند.

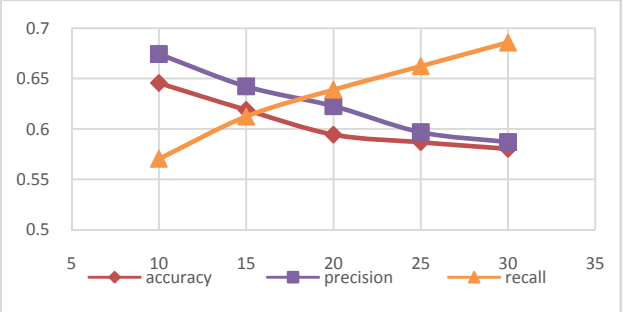
در شکل ۵ دقت پیشنهادها را به ازای تعداد قانون با کیفیت نشان می‌دهد. به ازای ۲۰ قانون با کیفیت هر سه پارامتر به توازن رسیده‌اند.

در پیش آزمایش اول تعداد جمعیت اولیه برای ۴، ۱۰ و ۱۶ مورد بررسی قرار گرفته است. الگوریتم GARM برای اجرای الگوریتم در تعداد تکرارهای ۳۰، ۴۰، ۵۰ و غیره در نظر گرفته شده است تا به همگرایی برسد. همان‌گونه که در شکل ۶ و جدول ۳ نشان داده شده است. با توجه به زمان اجرای حداقل الگوریتم در تعداد

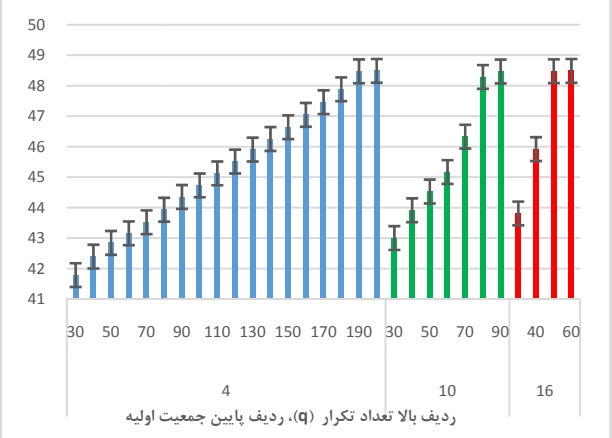
جمعیت ۱۰ نفر، برای اجرای نهایی جمعیت ۱۰ نفره مورد نظر قرار می‌گیرد.



شکل ۴- کیفیت پیشنهادها به ازای w2های مختلف (w1+w2=1)



شکل ۵- کیفیت پیشنهادها به ازای تعداد قانون مختلف (k)



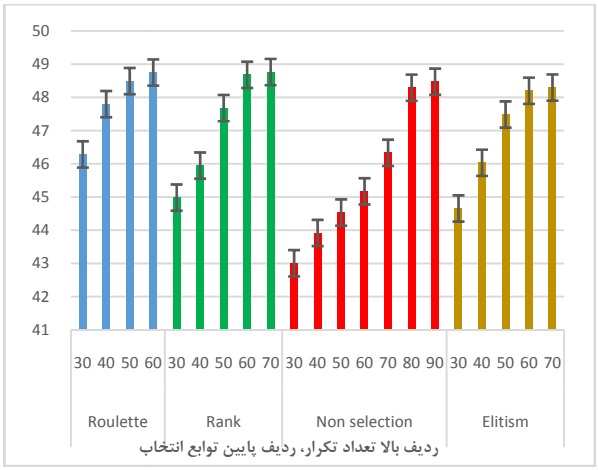
شکل ۶- میانگین کیفیت در هر تکرار برای جمعیت های اولیه مختلف

جدول ۳- همگرایی الگوریتم به ازای جمعیت‌های مختلف

جمعیت (نفر)	تعداد تکرار الگوریتم	زمان اجرا (دقیقه)
۴	۱۹۰	۱۶۷
۱۰	۸۰	۱۶۳
۱۶	۵۰	۱۷۳

۵-۶- پیش آزمایش برای توابع انتخاب الگوریتم ژنتیک

با توجه به تنوع موجود در توابع انتخاب الگوریتم ژنتیک، در پیش آزمایش دوم الگوریتم GARM به ازای توابع انتخاب چرخ رولت[۳۳]، رتبه[۳۴] و نخبه‌گرایی[۳۵] و بدون انتخاب[۳۶] اجرا شده است. در حالت Non selection دقیقا هر نسل جدید جایگزین نسل قبل می‌گردد. شکل ۴ میانگین همگرایی توابع انتخاب را برای تکرارهای مختلف نشان می‌دهد. زمان اجرا در لحظه همگرایی برای تابع انتخاب چرخ رولت ۱۰۲ دقیقه، تابع انتخاب رتبه ۱۲۰ دقیقه، تابع انتخاب نخبه گرایی ۱۲۱ دقیقه و بدون تابع انتخاب ۱۶۳ دقیقه می‌باشد. با توجه به میانگین کیفیت برابر در همه‌ی توابع مختلف در نهایت از تابع انتخاب چرخ رولت با توجه به زمان اجرای حداقل، برای اجرای نهایی مورد نظر قرار خواهد گرفت.



شکل ۷- میانگین کیفیت در هر تکرار برای توابع انتخاب مختلف

۵-۷- مقایسه

اجرای نهایی برای تعداد ۱۰۰، ۲۵۰ و ۵۰۰ قلم هدف انجام گرفته است. زمان اجرا و میانگین کیفیت الگوریتم GARM و الگوریتم MOPSO پس از همگرایی در جدول ۴ نشان داده شده است. زمان اجرای الگوریتم GARM تقریبا ۲۵ درصد نسبت به الگوریتم MOPSO بهبود یافته است. خروجی قوانین انجمنی به سیستم پیشنهاددهنده داده می‌شود. خروجی قوانین انجمنی الگوریتم MOPSO با سیستم پیشنهاددهنده آزمایش شده است و دقت قوانین استخراجی از آن ارزیابی شده که مطلوب و مورد قبول بوده است [۱۱]. سرعت بالاتر الگوریتم GARM نسبت به الگوریتم MOPSO در تولید قوانین انجمنی در کل باعث سرعت بالاتر در تولید پیشنهاد می‌گردد.

جدول ۴- مقایسه الگوریتم‌های GARM و MOPSO

نام الگوریتم	تعداد قلم	زمان اجرا (ساعت)	میانگین کیفیت
GARM	۱۰۰	۱.۷	۴۸.۵۴
	۲۵۰	۴.۲۳	۴۸.۵۴
	۵۰۰	۸.۳۳	۴۸.۵۶
MOPSO	۱۰۰	۲.۲	۴۸.۴۸
	۲۵۰	۵.۴۷	۴۸.۵۳
	۵۰۰	۱۰.۹۷	۴۸.۵۰

الگوریتم MOPSO به دلیل سربار محاسباتی نسبت به الگوریتم GARM کندتر عمل می‌کند. علت این سربار محاسباتی، محاسبه روابط ۶ و ۷ در هر تکرار است. در حالی که الگوریتم GARM با انجام عملیات ترکیب و جهش، سربار بسیار

پایینی دارد و هدف‌مندتر و منطقی‌تر عمل می‌کند. همچنین لازم به ذکر است به عنوان مثال برای ۵۰۰ کاربر، اجرای الگوریتم GARM حدودا ۸.۳۳ ساعت طول کشیده است یعنی به طور متوسط برای هر قلم ۱ دقیقه طول کشیده است تا قوانین انجمنی محاسبه شده و پیشنهادها مشخص گردند که این مسئله با هدف اول الگوریتم‌های تکاملی سازگار است یعنی در زمان خیلی کم، بدون محاسبات پیچیده و زمان‌بر الگوریتم‌هایی مانند Apriori موفق به انجام پیشنهاد با کیفیت متوسط شده‌ایم که این زمان در الگوریتم MOPSO برای هر قلم ۱.۳ دقیقه طول کشیده است.

۶- نتیجه‌گیری و کارهای آینده

کاوش قوانین انجمنی یک مساله چند هدفه است که این مقاله با استفاده از یک روش اندازه‌گیری وزنی برای الگوریتم ژنتیک به صورت چند هدفه وزنی پرداخته است. الگوریتم ژنتیک با معرفی تابع ارزیاب خود به کاوش قوانین کاربردی در فضای جستجو می‌پردازد که این روش از روش‌های سنتی سریع‌تر و ضریب اطمینان بالاتری دارد. در نهایت الگوریتم GARM به دلیل سربار محاسباتی کمتر نسبت به الگوریتم MOPSO، سرعت بالاتری دارد و همین طور از لحاظ سرعت همگرایی با توجه به آزمایشات انجام شده می‌توان گفت الگوریتم GARM سرعت بالاتری نسبت به الگوریتم MOPSO دارا است که متعاقب آن باعث تولید پیشنهادهای بیشتر می‌شود.

دلیل اصلی بهبود پیشنهادات ذات الگوریتم می‌باشد که الگوریتم MOPSO ماهیتاً برای محیط‌های پیوسته مناسب است علی‌رغم اینکه الگوریتم GA سعی دارد با جایگشت‌های مختلف، نسل جدید به وجود آورد که برای سیستم گسسته این پژوهش مناسب‌تر می‌باشد، نتایج نیز این مساله را نشان می‌دهند.

قوانین قوی براساس همین معیار انتخاب و به سیستم پیشنهاددهنده داده می‌شوند. در حالی که معیارهای دیگری مانند لیفت[۳۷]، خی دو و غیره هستند که می‌تواند قوانین بهتری را انتخاب کند. سیستم پیشنهاددهنده‌ای که در این مقاله معرفی شد، قبلا در مقاله الگوریتم MOPSO معرفی شده بود. بهبود سیستم پیشنهاددهنده برای تولید پیشنهادات با کیفیت‌تر می‌تواند برای کارهای بعدی مورد نظر باشد.

مراجع

[1] G. Adomavicius, and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 17, pp. 734-749, 2005.

[2] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994, pp. 175-186.

[3] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender systems: an introduction*: Cambridge University Press, 2010.

[4] A. Ansari, S. Essegaier, and R. Kohli, "Internet recommendation systems," *Journal of Marketing research,* vol. 37, pp. 363-375, 2000.

[5] R. Burke, "Hybrid recommender systems: Survey and experiments," *User modeling and user-adapted interaction,* vol. 12, pp. 331-370, 2002.

[6] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *Internet Computing, IEEE,* vol. 7, pp. 76-80, 2003.

[7] H. Ma, I. King, and M. R. Lyu, "Effective missing data prediction for collaborative filtering," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, 2007, pp. 39-46.

[8] G.-R. Xue, C. Lin, Q. Yang, W. Xi, H.-J. Zeng, Y. Yu, and Z. Chen, "Scalable collaborative filtering using cluster-based smoothing," in Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2005, pp. 114–121.

[9] S. Tyagi, and K. K. Bharadwaj, "A Collaborative Filtering Framework Based on Fuzzy Case-Based Reasoning," in *Proceedings of the International Conference on Soft Computing for Problem Solving (SocProS 2011) December 20-22, 2011*, 2012, pp. 279-288.

[10] D. M. Pennock, E. Horvitz, S. Lawrence, and C. L. Giles, "Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach," in *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, 2000, pp. 473-480.

[11] S. Tyagi, and K. K. Bharadwaj, "Enhancing collaborative filtering recommendations by utilizing multi-objective particle swarm optimization embedded association rule mining," *Swarm and Evolutionary Computation,* vol. 13, pp. 1-12, 2013.

[12] W. Lin, S. A. Alvarez, and C. Ruiz, "Efficient adaptive-support association rule mining for recommender systems," *Data mining and knowledge discovery,* vol. 6, pp. 83-105, 2002.

[13] Z. Huang, H. Chen, and D. Zeng, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Transactions on Information Systems (TOIS),* vol. 22, pp. 116-142, 2004.

[14] C. W.-k. Leung, S. C.-f. Chan, and F.-l. Chung, "A collaborative filtering framework based on fuzzy association rules and multiple-level similarity," *Knowledge and Information Systems,* vol. 10, pp. 357-381, 2006.

[15] M. Grami, R. Gheibi, and F. Rahimi, "A novel association rule mining using genetic algorithm," in *Information and Knowledge Technology (IKT), 2016 Eighth International Conference on*, 2016, pp. 200-204.

[16] W. Soto, and A. Olaya-Benavides, "A genetic algorithm for discovery of association rules," in *Chilean Computer Science Society (SCCC), 2011 30th International Conference of the*, 2011, pp. 289-293.

[17] S. Ghosh, S. Biswas, D. Sarkar, and P. Sarkar, "Association rule mining algorithms and Genetic Algorithm: A comparative study," in *Emerging Applications of Information Technology (EAIT), 2012 Third International Conference on*, 2012, pp. 202-205.

[18] T. I. R. Agrawal, and A. Swami, "Mining association rules between sets of items in large databases," *Proceeding of the ACM International Conference on Management of Data, ACM,* pp. 207–216, 1993.

[19] M. Srinivas, and L. M. Patnaik, "Genetic algorithms: A survey," *Computer,* vol. 27, pp. 17-26, 06 August 2002 1994.

[20] A. Ganjehkaviri, M. M. Jaafar, S. Hosseini, and H. Barzegaravval, "Genetic algorithm for optimization of energy systems: Solution uniqueness, accuracy, Pareto convergence and dimension reduction," *Energy,* vol. 119, pp. 167-177, 2017.

[21] T. İnkaya, and M. Akansel, "Coordinated scheduling of the transfer lots in an assembly-type supply chain: A genetic algorithm approach," *Journal of Intelligent Manufacturing,* vol. 28, pp. 1005-1015, 2017.

[22] M. M. J. Kabir, S. Xu, B. H. Kang, and Z. Zhao, "A New Multiple Seeds Based Genetic Algorithm for Discovering a Set of Interesting Boolean Association Rules," *Expert Systems with Applications,* 2017.

[23] F. Ramezani, J. Lu, J. Taheri, and F. K. Hussain, "Evolutionary algorithm-based multi-objective task scheduling optimization model in cloud environments," *World Wide Web,* vol. no. 6, pp. 1-21, 2015.

[24] F. Ramezani, J. Lu, and F. Hussain, "Task Scheduling Optimization in Cloud Computing Applying Multi-Objective Particle Swarm Optimization," in *Service-Oriented Computing*, ed: Springer, 2013, pp. 237-251.

[25] S.-Y. Wur, and Y. Leu, "An effective Boolean algorithm for mining association rules in large databases," in *Database Systems for Advanced Applications, 1999. Proceedings., 6th International Conference on*, 1999, pp. 179-186.

[26] GroupLens. (1998, 2016-11-17). *Movielens Dataset*. Available: https://grouplens.org/datasets/movielens/

بهزاد سلیمانی نیسیانی دانشجوی سال پنجم دکتری در رشته مهندسی نرم‌افزار کامپیوتر است. حوزه تحقیقاتی که تا کنون در آنها پژوهش انجام داده و مقاله نیز به چاپ رسانده است بیشتر شامل الگوریتم‌های داده‌کاوی در زمینه قواعد انجمنی، طبقه‌بندی، سیستم‌های پیشنهاددهنده بوده است. از جمله زمینه‌های تحقیقاتی دیگر ایشان در زمینه سیستم‌های توزیع شده مانند شبکه‌های نظیر به نظیر، محاسبات ابری، همچنین در زمینه‌های مهندسی نرم‌افزار و هوش مصنوعی می‌توان اشاره کرد. وی هم‌اکنون مدرس دانشگاه کاشان و دانشگاه آزاد اسلامی واحد اصفهان (خوراسگان) و دانشگاه شهید اشرفی اصفهانی است که بر روی الگوریتم‌های متن کاوی و کشف گزارش‌های متنی تکراری در پروژه دکترای خود کار می‌کند.

آدرس پست‌الکترونیکی ایشان عبارت است از:
b.soleimani@grad.kashanu.ac.ir

نسیم سلطانی تحصیلات تکمیلی خود را در رشته مهندسی کامپیوتر گرایش نرم‌افزار به انجام رسانده و اکنون فارغ‌التحصیل مقطع کارشناسی‌ارشد و پژوهشگر در حوزه‌ی محاسبات ابری است. تحقیقات مورد علاقه‌ی نامبرده هوش مصنوعی، الگوریتم‌های فرا اکتشافی، داده‌کاوی و بهبود زمانبندی وظایف می‌باشد.
آدرس پست‌الکترونیکی ایشان عبارت است از:
n.soltani@naeini.ac.ir

رضا مفیدی فارغ‌التحصیل رشته مهندسی کامپیوتر گرایش نرم‌افزار می‌باشد که اکنون فارغ‌التحصیل مقطع کارشناسی‌ارشد است که در حوزه‌ی سیستم‌های پیشنهاددهنده و قوانین انجمنی فعالیت دارد. تحقیقات مورد علاقه‌ی ایشان الگوریتم‌های فرا اکتشافی و داده‌کاوی می‌باشد.
آدرس پست‌الکترونیکی ایشان عبارت است از:
rezamofidi66@naeini.ac.ir

محمد حسین ندیمی شهرکی استادیار و رئیس مرکز تحقیقات مه داده در دانشگاه آزاد اسلامی واحد نجف‌آباد است. موضوعات پژوهشی مورد علاقه‌ی ایشان داده‌کاوی، کاوش شبکه‌های اجتماعی، سیستم‌های هوشمند و کاوش داده‌های پزشکی می‌باشد.
آدرس پست‌الکترونیکی ایشان عبارت است از:
nadimi@ieee.org, nadimi@iaun.ac.ir

اطلاعات بررسی مقاله:

تاریخ ارسال: ۱۳۹۵/۱۲/۰۵
تاریخ اصلاح: ۱۳۹۶/۱۱/۲۰
تاریخ قبول شدن: ۱۳۹۷/۰۲/۰۵
نویسنده مرتبط: بهزاد سلیمانی نیسیانی، دانشکده فنی و مهندسی، دانشگاه آزاد اسلامی واحد اصفهان (خوراسگان)، اصفهان، ایران.

[1] Item
[2] Content-Based
[3] Knowledge-Based
[4] CF: Collaborative Filtering
[5] Collaborative Recommender Systems
[6] Sparse
[7] ARM: Association Rule Mining
[8] Huang
[9] Leung
[10] FARAMS: Fuzzy Association Rule And Multiple-Level Similarity
[11] ASARM: Adaptive-Support Association Rule Mining
[12] Apriori
[13] MOPSO: Multi-Objective Particle Swarm Optimization Association Rule Mining
[14] Agrawal

[15]Tail
[16]Head
[17]Support
[18]Confidence
[19]Genetic Algorithm
[20]Crossover
[21]Mutation
[22]Genetic Algorithm Association Rule Mining
[23]Representation Schema
[24]Transformation Schema
[25]Like
[26]Dislike
[27]Selection
[28]Cross Over
[29]Mutation
[30]Matlab
[31]Movielens
[32]Iteration
[33]Roulette
[34]Rank
[35]Elitism
[36]Non Selection
[37]Lift