روش تکاملی بهبود انتخاب الگوریتم در سامانه‌های توصیه‌گر فیلترینگ مشارکتی

مژده رباطی انارکی[1*]، نوشین ریاحی[2]

[1و2] گروه مهندسی کامپیوتر، دانشکدهٔ فنی مهندسی، دانشگاه الزهرا، تهران، ایران

چکیده

سامانه توصیه‌گر می‌تواند به‌عنوان نرم‌افزاری که به افراد، مناسب‌ترین آیتم‌ها را پیشنهاد می‌کنند، تعریف شود. این سامانه‌ها به‌عنوان یک مشاور کار می کنند تا کاربران را در یافتن محصولات موردعلاقه‌شان راهنمایی کنند. امروزه برای پیاده‌سازی سامانه‌های توصیه‌گر، الگوریتم‌های متفاوتی به کار برده می‌شود؛ دسته‌ای از این الگوریتم‌ها، الگوریتم‌های فیلترینگ مشارکتی نام دارند، که در آن‌ها از شباهت کاربران یا شباهت روابط ایجادشده توسط کاربران میان آیتم‌ها برای تعیین پیشنهادها استفاده می‌شود. روش‌های فیلترینگ مشارکتی، می‌توانند آیتم‌هایی موردپسند کاربر امّا با محتوایی به‌طور کامل متفاوت نسبت به سلایق پیشین او پیشنهاد دهند که این پیشنهادها بر اساس علایق کاربران مشابه به کاربر هدف تولید شده است. در روش‌های فیلترینگ مشارکتی، برای ایجاد الگو یا محاسبهٔ شباهت بین کاربران، معیارها و توابع فاصله متفاوتی استفاده شده است و روش بهینه که بهترین فهرست از پیشنهادها را تولید کند، همواره یکسان نیست و متناسب با داده‌های موجود، این الگوریتم در میان الگوریتم‌های فیلترینگ مشارکتی، تغییر می‌کند؛ به همین دلیل، انتخاب روش مناسب برای ایجاد یک سامانهٔ توصیه‌گر، به چالشی برای طرّاحان این سامانه تبدیل شده است. در این مقاله روشی مبتنی بر الگوریتم ژنتیک برای اجتماع نتایج روش‌های همسایه‌محور و انتخاب بهترین پیشنهادها از بین پیشنهادهای تولیدشده با روشهای مختلف با معیارهای فاصله متفاوت، برای یک سامانهٔ توصیه‌گر با هدف پیشنهاد N آیتم برتر، ارائه شده است. در پیاده‌سازی این روش‌ها، علاوه‌بر محاسبهٔ شباهت مستقیم کاربران، تعیین اطمینان استنتاج‌شدهٔ کاربران نیز در نظر گرفته شده است تا اطّلاعات موجود از ارتباط بین علایق کاربران افزایش یابد. روش پیشنهادی، برای هر مجموعه داده یک ترکیب از روش‌های فیلترینگ مشارکتی ایجاد می‌کند که علاوه‌بر در نظرگرفتن محدودیّت‌های زمانی در تولید آن، دقّت مناسبی دارد. این روش با روش‌های فیلترینگ مشارکتی همسایه‌محور به‌صورت مجزا و همچنین،، سامانه‌های مشابه با استفاده از مجموعه‌دادگان Movielens 100k و Movielens 1M و Hetrec2011 مقایسه شده است. آزمایش‌ها برتری و توانایی تولید پیشنهادهای دقیق‌تر به کاربران با این روش را نشان می‌دهد.

واژگان کلیدی: سامانهٔ توصیه‌گر، فیلترینگ مشارکتی، الگوریتم ژنتیک، توابع فاصله، اجتماع نتایج، اطمینان استنتاج شده.

An evolutionary Approach for automating the Selection of Optimum Algorithm in Collaborative Filtering Recommender Systems

Mojdeh Robati Anaraki[1*], Nooshin Riahi[2]

Computer Engineering, Alzahra University, Tehran, Iran[1,2]

Abstract

The expansion of online resources over the last few decades, has made the internet the main source for multimedia information, such as movies, books, music, etc. When the amount of choices becomes overwhelmingly large, the need for a recommender system arises. Recommender systems can be defined

* نویسندهٔ عهده‌دار مکاتبات

* Corresponding author

• تاریخ ارسال مقاله: ۱۳۹۹/۱۱/۱۱ • تاریخ پذیرش: ۱۴۰۱/۱۲/۳ • تاریخ انتشار: ۱۴۰۲/۵/۲۰ • نوع مطالعه: پژوهشی

as software programs to suggest the most appropriate and closest items to the user's taste. They act as a counselor, behaving in such a way to guide people through the discovery of products of interest.
Nowadays, there are numerous recommendation methods used to implement a recommender system. These methods, generally, fall into three main categories. 1) Content-based techniques that recommend items similar to the user's highest rated items. 2) Collaborative filtering techniques, which use similarity between users or items according to the user's rating pattern, for generating recommendations. These recommendations are based on the preferences of users who have similar tastes to the target user. 3) Hybrid techniques that combine these two techniques to address the shortcomings of each one. Collaborative filtering methods are categorized as 1) Model-based methods, in which they use the user's rating to create a model for recommendations, and 2) Memory-based (Neighborhood-based) methods, which find the most similar users to the target user (neighbors) to create recommendations for the user. Various similarity functions and metrics have been used to create the model or compute the similarity in collaborative filtering methods. When using collaborative filtering methods, one of the challenges encountered, is the cold start problem. This occurs because in e-commerce systems, most items have only been rated by a few users. Because there are not enough ratings, it becomes difficult to calculate the similarity between users or items. There are several approaches to tackle the cold start problem in collaborative filtering systems.
When building a recommender system, it can be challenging to manually select a method from the recommendation methods, since there are many methods to choose from. The best method to generate the most relevant recommendations, may vary depending on the available data of users and items, since each approach has its own particularities and depends on the context. Furthermore, because the data in online systems are constantly changing, with a fundamental change in the data, the best method for the system might also change. Therefore, there is a need for a technique to automate the selection of recommendation technique for the system.
This paper proposes a method that allows us to choose the best combination of memory-bsed collaborative filtering algorithms that, when used on selected data, will make the optimal recommendations in a limited amount of time. Among the main methods to create recommendations, collaborative filtering methods have better results because they rely on quality data provided by users, compared to content-based methods that rely on item information. Therefore, collaborative filtering methods have been selected as the base methods for the purposed approach, and since the goal is to create a system that can choose the optimum recommendation algorithm based on general data statistics, such as the ratio of items to users, if the data changes fundamentally, we need to be able to rerun the algorithm to select a new combination of methods. Since the selected methods for this system need to be run in a relatively short time, only the memory-based methods in collaborative filtering methods have been selected, because they do not need a training step, also they can adapt themseleve to changes in the data, faster. To address the cold start issue, in addition to computing the similarity between users, inferred trust is also computed to increase the amount of available information about relations between users' interests. Since we the time is limitted, we need a technique to find the best combination of methods without testing each combination. Therefore, we used the genetic algorithm that reflects the process of natural selection, where the fittest individuals are selected for reproduction in order to produce offspring of the next generation. This process is continued to iterate and eventually, a generation with the fittest individuals will be found within a limited time. This approach, ultimately, proposes a combination of collaborative filtering techniques for each data set. This will automate the selection of the optimum algorithm for a recommender system, and the resulting combination, in addition to considering time limits, will have an acceptable precision for making recommendations.
The proposed approach, uses a genetic algorithm for ranking aggregation of memory-based collaborative filtering methods, selects the most relevant recommendations generated by different similarity techniques to create a Top-N recommender system. This approach has been compared to individual memory-based collaborative filtering methods and other similar methods. The experiments were performed using 1M MovieLens and 100k MovieLens and HetRec2011 data sets. The results show that the proposed approach in this paper, is performmed better and it has a higher precision in generating recommendations for users, than the other similar algorithms.

Keywords: recommender systems, collaborative filtering, genetic algorithm, similarity metrics, rank aggregation, inferred trust.

۱- مقدمه

با گسترش شبکه‌ها در دهه‌های گذشته، اینترنت تبدیل به منبع اصلی دستیابی به اطّلاعات چندرسانه‌ای مانند فیلم، کتاب، موسیقی و... شده است. هنگامی که تعداد انتخاب‌ها در اینترنت برای کاربران زیاد و گیج‌کننده می‌شود، نیاز به سامانه‌ای احساس می‌شود که اطّلاعات را برای آنها اولویت‌بندی و فیلتر کند. سامانه‌های توصیه‌گر برای رفع این مشکل ایجاد شده‌اند تا با جستجو در میان این حجم عظیم اطّلاعات که به‌صورت پویا افزایش

می‌یابد، محتواها و سرویس‌های شخصی‌سازی‌شده‌ای به کاربران ارائه دهند [1] .

طرّاحی یک سامانهٔ توصیه‌گر می‌تواند اهداف متفاوتی داشته‌باشد. به‌دلیل‌این‌که کاربردهای مختلف نیازهای متفاوتی دارند، طرّاح سامانه باید ویژگی‌های مدنظر سامانه را به درستی انتخاب کند. این ویژگی‌ها شامل مواردی از جمله دقّت پیش‌بینی مناسب، نوآوری در پیشنهادها[1]، توانایی تطبیق با تغییرات[2]، مقیاس‌پذیری[3]، تنوع[4] و جدید بودن پیشنهادها[5] است [2].

روش‌ها و راهکارهای گوناگونی در ادبیات سامانه-های توصیه‌گر وجود دارد که امکان تولید پیشنهادها را ممکن می‌کند. الگوریتم‌های توصیه‌گر بر اساس روشی که برای تولید پیشنهادها استفاده می‌کنند، به سه دستهٔ اصلی تقسیم شده است [3]: (۱) روش‌های مبتنی بر محتوا[6] که در این روش آیتم‌هایی با اطّلاعات مشابه مواردی که کاربر از پیش به آن‌ها امتیاز بالایی داده است، پیشنهاد می‌کند. (۲) روش‌های فیلترینگ مشارکتی[7] که آیتم‌هایی را که توسّط افراد با سلایق مشابه کاربر هدف انتخاب شده است، پیشنهاد می‌دهد. (۳) روش‌های ترکیبی[8] که هر دو روش را ترکیب می‌کنند تا مشکلاتی را که یک روش به‌تنهایی دارد، برطرف کند [4]. انتخاب از میان این مجموعه گوناگون برای تعیین روش تولید پیشنهادها می‌تواند باعث سردرگمی طرّاحان سامانه در انتخاب بهترین روش برای استفاده در سامانهٔ توصیه‌گر شود.

انتخاب بهترین روش برای پیاده‌سازی یک سامانهٔ توصیه‌گر که با داده‌های موجود در سامانه، باتوجه‌به نسبت تعداد آیتم‌ها و کاربران و تعداد امتیازدهی‌های کاربران به آیتم‌های موجود در سامانه، کارایی بهینه داشته باشد، به‌صورت دستی امکان‌پذیر نیست، زیرا برای پیاده‌سازی سامانه‌های توصیه‌گر انتخاب‌های فراوانی وجود دارد و هریک از این روش‌ها در شرایط ویژه‌ای بهتر کار می‌کنند و همچنین، اطّلاعات موجود سامانه‌های برخط پیوسته در حال تغییرند و با تغییر اساسی داده‌ها ممکن است روش بهینه برای این سامانه نیز، تغییر کند.بنابراین، به سامانه‌ای نیاز است که بتواند روشی برای خودکارکردن انتخاب روش پیاده‌سازی سامانهٔ توصیه‌گر در اختیار طرّاحان آن قرار دهد.

از میان دسته‌های اصلی روش‌های تولید پیشنهادها، الگوریتم فیلترینگ مشارکتی مانند روش مبتنی بر محتوا، پیشنهادها را نه بر اساس اطّلاعات آیتم‌ها، که نشانگر نامناسبی از کیفیت آیتم‌هاست، بلکه بر اساس کیفیت آیتم‌ها، که توسط کاربران تعیین می‌شود، تولید می‌کند [2]. به همین دلیل در این مقاله از میان روش‌های تولید پیشنهادها، الگوریتم‌های فیلترینگ مشارکتی بررسی شده است؛ که خود شامل روش‌های الگومحور[9] و همسایه‌محور[10] (حافظه‌محور) است [3].

روش‌های الگومحور به‌طور معمول، زمان طولانی‌تری در مقایسه با روش‌های همسایه‌محور برای تولید پیشنهادها نیاز دارند، چون شامل مرحلهٔ آموزش هم می‌شوند، ولی روش‌های همسایه‌محور می‌توانند در هر لحظه بر اساس اطّلاعات کاربر پیشنهادهایی برای او تولید کنند [2]. هدف این مقاله ایجاد سامانه‌ای کاربردی برای انتخاب الگوریتم سامانه توصیه‌گر است که نتیجهٔ آن به وضعیّت کلی اطّلاعات سامانه، مانند نسبت تعداد آیتم‌ها به کاربران بستگی دارد؛ به همین دلیل باید در زمان تغییرات اساسی داده‌ها در سامانه بتوان آن را در زمانی کوتاه دوباره اجرا کرد تا مجموعهٔ جدیدی از روش‌ها را برای سامانه ایجاد کند. از همین رو، روش‌های همسایه‌محور را برای تولید ترکیب‌های این سامانه بررسی کرده‌ایم که نیاز به مرحلهٔ آموزش ندارند و در لحظه می‌توانند نتایج را محاسبه کنند. این روش‌ها شامل الگوریتم‌های گوناگونی می‌شود که هریک ویژگی‌های خاصی دارند و نتایج آن‌ها بر اساس داده‌های سامانه‌ای که در آن اجرا می شوند، تغییر می‌کند و برخی روش‌ها نیز برای بعضی سامانه‌ها بهتر عمل می‌کنند. برای تعیین بهترین روش و همچنین، بهره‌گیری از چند الگوریتم برای ایجاد پیشنهادها می‌توان ترکیبی از پیشنهادهای تولیدشده از این روش‌ها را انتخاب کرد.

برای پیدا کردن ترکیب بهینهٔ الگوریتم‌های توصیه با هدف بالا بردن دقّت فهرست تولیدشده برای هر کاربر، نیاز به روشی داریم که بدون نیاز به ارزیابی همهٔ حالت‌های ممکن ترکیب نتایج روش‌های موجود، بتواند ترکیب مناسبی را در زمانی محدود پیدا کند؛ زیرا بررسی

[1] Novelty
[2] Adaptivity
[3] Scalability
[4] Diversity
[5] Serendipity
[6] Content-based
[7] Collaborative filtering
[8] Hybrid
[9] Model-based
[10] Neighborhood -based

تمام حالات ممکن به علت بزرگ بودن فضای جستجو امکان‌پذیر نیست. به همین دلیل برای حل این مسئله نیاز به روشی داریم که بتواند نتایج نامناسب را حذف کرده و بر اساس ترکیب‌هایی به‌نسبت بهتر، ترکیب‌های مناسب جدیدی تولید کند و در زمان مشخّصی به نتیجه‌ای قابل قبول برسد. به این دلیل، از الگوریتم تکاملی ژنتیک استفاده شده است که با تولید جمعیّت اولیه با تعدادی محدود از ترکیب نتایج الگوریتم‌ها، می‌تواند در هر نسل این ترکیب‌ها را بهبود داده و پس‌از زمان مشخصی نتیجهٔ قابل‌قبولی تولید کند. انتخاب الگوریتم ژنتیک از میان روش‌های تکاملی به این دلیل است که می‌توان به‌سادگی در آن یک پاسخ برای مسئله (شامل تعداد آیتم‌های انتخاب‌شده از هر روش) نمایش داد؛ زیرا یک کروموزوم در الگوریتم ژنتیک به‌طور معمول به‌صورت رشته‌ای است و می‌توان در این مسئله هر کاراکتر این رشته را برابر تعداد آیتم‌های انتخاب‌شده از هر روش در نظر گرفت؛ و همچنین، نیازی به اعمال تابعی برای تغییر نمایش پاسخ مسئله برای قرار گرفتن در الگوریتم برای کاهش پیچیده‌تر شدن روش نهایی ندارد و تنها نیاز است که در آن مجموع اعداد یک ترکیب همیشه ثابت نگه داشته شود. همچنین، الگوریتم ژنتیک به انعطاف‌پذیری و سادگی پیاده‌سازی شناخته شده است [4] و می‌تواند انتخاب مناسبی برای این مسئله باشد.

در سامانه‌های تجاری اینترنتی[11] بیشتر آیتم‌ها فقط از سوی تعداد اندکی از کاربران امتیازدهی شده‌است، و به‌دلیل ناکافی بودن امتیازدهی‌ها در سامانه، محاسبه شباهت بین کاربران و بین آیتم‌ها برای سامانه توصیه‌گر بسیار مشکل است[5] . به این چالش، پراکندگی در مجموعه داده‌ها[12] گفته می‌شود. برای رفع پراکندگی داده‌ها روش‌های گوناگونی در سامانه‌های توصیه‌گر استفاده شده و در این مقاله از روش محاسبه اطمینان استنتاج‌شده[13] بین کاربرانی که به‌دلیل نداشتن آیتم‌های مشترک، اطّلاعی از میزان اطمینان آن‌ها به یکدیگر نداریم، استفاده شده است. اطمینان می‌تواند در شبکه منتشر شود، به همین دلیل این سامانه‌ها می‌توانند بر چالش‌های پراکندگی داده‌ها و شروع سرد غلبه کنند[6] . با کمک معیار اطمینان می‌توان با استفاده از اطّلاعات امتیازدهی کاربران، میزان اطمینان آن‌ها به یکدیگر را به دست آورد و با استفاده از ویژگی تراگذر بودن[14] اطمینان (به معنای اینکه اگر کاربر یک به کاربر دو اطمینان داشته باشد و کاربر دو به کاربر سه اطمینان داشته باشد، می‌توان نتیجه گرفت کارب یک۱ نیز تا حدودی به کاربر سه اطمینان دارد [6]) و انتشار اطمینان در شبکه، میزان اطمینان استنتاج‌شده بین این کاربران، که در حالت معمول شباهتی برایشان محاسبه نمی‌شود، به دست آورد و از این اطّلاعات برای تولید پیشنهادها استفاده کرد [7].

بر اساس موارد ذکرشده، در این مقاله با هدف پیدا کردن بهترین ترکیب نتایج روش‌های فیلترینگ مشارکتی، سامانهٔ توصیه‌گری ارائه شده است که با استفاده از اطمینان و با کمک الگوریتم ژنتیک، برای هر سامانه ترکیبی مناسب از نتایج روش‌های توصیه انتخاب می‌کند که می‌تواند پیشنهادهای بهتری به نسبت پیشنهادهای یک روش مجزّا تولید کند و همچنین، فرایند انتخاب مناسب‌ترین الگوریتم در سامانهٔ توصیه‌گر را برای طرّاحان سامانه خودکار کند.

در ادامه، و در بخش دو، مفاهیم اصلی روش پیشنهادی تشریح خواهد شد و مروری بر کارهای پیشین انجام‌شده خواهیم داشت. در بخش سه روش پیشنهادی بیان‌شده و دلایل انتخاب بخش‌های مختلف آن شرح داده شده است. در بخش چهارم، سامانهٔ پیشنهادی آزمایش و ارزیابی شده است و با روش‌های مشابه آن مقایسه می‌شود و در پایان، نتیجه‌گیری و پیشنهادهایی برای ادامهٔ این پژوهش مطرح می‌شود.

۲- ادبیات موضوع و کارهای پیشین

در این بخش مفاهیم اصلی استفاده‌شده در روش پیشنهادی بیان شده و بخشی از کارهای مرتبط انجام‌شده در هر زمینه بررسی می‌شود. همچنین، در پایان، تفاوت‌های روش پیشنهادی با کارهای پیشین بیان شده است.

۲-۱- توصیه‌گر فیلترینگ مشارکتی

سامانهٔ توصیه‌گر ابزاری است که برای کمک کردن به کاربران برای پیدا کردن اطّلاعات مناسب در بین هزاران محتوای برخط در دسترس، طرّاحی شده است. این سامانه‌ها به‌صورت پویا آیتم‌هایی را که بر اساس پروفایل کاربر برای او مناسب است، به کاربر پیشنهاد می‌دهند. یکی از پراستفاده‌ترین روش‌ها در سامانه‌های توصیه‌گر، فیلترینگ مشارکتی است. این روش آیتم‌هایی به کاربر پیشنهاد می‌دهد که کاربرانی با سلایق مشابه او، پیشتر پسندیده‌اند. شباهت سلیقهٔ دو کاربر بر اساس شباهت در

[11] e-commerce
[12] Data Sparsity
[13] Trust Inferences
[14] Transitivity

امتیازدهی‌های آن‌ها به آیتم‌ها، محاسبه می‌شود [2]. این روش مستقل از دامنه بوده و می‌تواند برای محتواهایی که به مقدار کافی اطّلاعات در مورد آن ها موجود نیست، استفاده شود. مانند فیلم و موسیقی [3].

روش‌های فیلترینگ مشارکتی بـه دو دسـتهٔ همسـایه-محور و الگو محور تقسیم می‌شود [3]:

۲-۱-۱- روش‌های همسایه‌محور:

این روش‌ها، روش‌هایی هستند که تنهـا بـر روی مـاتریس امتیازدهی‌های کاربرها به آیتم‌ها عمل می‌کنند [8].

آیتم‌هایی که پیشتر توسط کاربر امتیازدهی شده-اند، نقشی اساسی در جستجوی همسایه‌هایی که به او شبیهند، دارند. هنگامی که یک همسایه‌ی کاربر پیدا شود، می‌توان الگوریتم‌های مختلفی را برای تولید پیشنهادهایی برای کاربر هدف، از آیتم‌های موردعلاقهٔ این همسایه، به‌کار برد [3].

این روش‌ها به دو دستهٔ روش‌های کاربرمحور و آیتم‌محور تقسیم می‌شوند. در روش کاربرمحور، شباهت بین کاربران با مقایسهٔ امتیازدهی‌های آن‌ها بر روی آیتم‌های یکسان به دست می‌آید و سپس، بر اساس سلیقهٔ کاربران مشابه کاربر هدف، آیتم‌هایی به او پیشنهاد می‌شود. روش آیتم‌محور، بر اساس شباهت بین آیتم‌ها - بر پایهٔ کاربران مشترکی که به آن‌ها امتیاز داده‌اند- عمل می‌کند و آیتم‌هایی با الگوی امتیازدهی مشابه آیتم مورد علاقهٔ کاربر، به او پیشنهاد می‌دهد [3].

• روش‌های محاسبه شباهت

تاکنون برای محاسبهٔ شباهت بین کاربران و شباهت میان آیتم‌ها، از توابع مختلفی استفاده شده است. میزان شباهت در الگوریتم‌های همسایه‌محور نقش بسیار مهمی دارد، زیرا با استفاده از آن، همسایه‌های قابل اطمینان کاربر، که از امتیازهای آنان در پیش‌بینی استفاده می‌شود، مشخص می‌شود و همچنین، این معیار ابزاری برای تعیین میزان اهمیّت همسایه‌ها در انجام پیش‌بینی‌ها برای کاربر، در اختیار الگوریتم قرار می‌دهد. محاسبهٔ میزان شباهت، یکی از حیاتی‌ترین بخش‌های ایجاد یک سامانهٔ توصیه‌گر همسایه‌محور است، زیرا نقش مهمی بر میزان دقّت و کارایی سامانه دارد [3] [2].

با استفاده از روش‌های (جدول-۱) و روش city block با (رابطه-۱) در حالت‌های کاربرمحور و آیتم‌محور در الگوریتم فیلترینگ مشارکتی می‌توان میزان شباهت را محاسبه کرد.

روش city block با (رابطه-۱) میزان شباهت را محاسبه می‌کند.

$$w_{a,u} = \sum_{i=1}^{m} |r_{a,i} - r_{u,i}| \qquad (۱)$$

در این رابطه $r_{u,i}$ امتیاز کاربر u به آیتم i است و m تعداد آیتم‌ها است.

(جدول-۱) روش‌های محاسـبه شـباهت استفاده‌شـده در [4] را نشان می‌دهد. در این رابطه‌ها $r_{u,i}$ امتیاز کـاربر u به آیتم i و $\overline{r_u}$ میانگین امتیازهای کاربر u بـرای آیتم‌هـای مختلف است. m_u آیتم‌هایی است کـه کـاربر u پسـندیده - است و m همهٔ آیتم‌های موجود را در بر می‌گیرد.

• پیش‌بینی امتیاز آیتم‌ها

بر اساس مقدار شباهت محاسبه‌شده میان کاربران، امتیاز پیش‌بینی‌شده کاربر a برای آیتم i بر اساس (رابطه -۲) محاسبه می‌شود.

$$p_{a,u} = \overline{r_a} + \frac{\sum_{u=1}^{h} (r_{u,i} - \overline{r_u}) \times w_{a,u}}{\sum_{i=1}^{m} |w_{a,u}|} \qquad (۲)$$

در این رابطه $W_{a,u}$ مقدار محاسبه‌شده شباهت بین کاربران a و u، r_{ui} امتیاز کاربر u به آیتم i ، $\overline{r_u}$ میانگین امتیازهای کاربر u برای آیتم‌های مختلف و h تعداد همسایه‌های کاربر هدف است. با این روش امتیاز کاربر u برای آیتم a، با استفاده از میانگین وزنی امتیازی که کاربران همسایه کاربر u به آیتم a داده‌اند، با ضریب نسبت شباهت بین کاربر u و کاربر همسایه‌اش، پیش‌بینی می‌شود.

۲-۱-۲- روش‌های الگو محور:

این روش‌ها از امتیازدهی‌های پیشین کاربر در ایجاد الگو برای تولید پیشنهادها استفاده می‌کنند و پس‌از انجام مرحلهٔ آموزش می‌توانند با سرعت بیشتری فهرستی از آیتم‌ها را به کاربر پیشنهاد دهند، به‌دلیل‌این‌که از الگوهای ازپیش‌محاسبه‌شده استفاده می‌کنند [3].

روش‌هایی مانند استفاده از شبکهٔ عصبی[15] و فاکتورگیری ماتریس[16] و خوشه‌بندی[17] و درخت

[15] Neural Networks
[16] Matrix factorization
[17] Clustering

تصمیم‌گیری[18] برای تولید پیشنهادها، در دستهٔ روش‌های الگومحور فیلترینگ مشارکتی جای می‌گیرند [10] , [11], [12], [2].

(جدول -۱) : روش‌های محاسبهٔ شباهت بازنویسی‌شده از [4] با تغییرات

(Table-1): Techniques for calculating similarity proposed by [4] with some modifications

روش	روش محاسبهٔ شباهت دو کاربر
Pearson correlation	$w_{a,u} = \frac{\sum_{i=1}^{m} (r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i=1}^{m} (r_{a,i} - \bar{r}_a)^2 \sum_{i=1}^{m} (r_{u,i} - \bar{r}_u)^2}}$
Euclidean	$w_{a,u} = \sqrt{\sum_{i=1}^{m} (r_{a,i} - r_{u,i})^2}$
Cosine	$w_{a,u} = \frac{\sum_{i=1}^{m} (r_{a,i} \times r_{u,i})}{\sqrt{\sum_{i=1}^{m} (r_{a,i})^2 \sum_{i=1}^{m} (r_{u,i})^2}}$
Spearman rank	روش spearman مانند pearson است ولی w را بر اساس رتبهٔ نسبی امتیاز، در بین امتیازهای کاربر به دست می‌آورد. pearson مقدار w را بر اساس مقادیر نسبی امتیازها محاسبه می‌کند.
Tanimoto	$w_{a,u} = \frac{m_a \cap m_u}{(\lvert m_a \rvert + \lvert m_u \rvert) - (\lvert m_a \cap m_u \rvert)}$
Log likelihood test	$w_{a,u} = \text{logLikelihoodRatio}(\lvert m_a \cap m_u \rvert, \lvert m_a \rvert - \lvert m_a \cap m_u \rvert, \lvert m_u \rvert - \lvert m_a \cap m_u \rvert, n - (\lvert m_a \rvert + \lvert m_u \rvert) + \lvert m_a \cap m_u \rvert)$ *پیاده سازی log likelihood ratio بر اساس مرجع [9] انجام شده‌است.

۲-۱-۳- مقایسهٔ روش‌های همسایه‌محور و الگومحور

روش‌های الگومحور می‌توانند ویژگی‌هایی از آیتم‌ها استخراج کنند که به‌صورت صریح تعیین نشده است. برای مثال، در یک سامانهٔ توصیه‌گر فیلم، این روش می‌تواند تعیین کند که یک کاربر به فیلم‌هایی که هم‌زمان کمدی و عاشقانه هستند علاقه دارد؛ بدون‌این‌که در عمل، برچسب‌های کمدی و عاشقانه را تعریف کند. این سامانه ممکن است فیلم کمدی-عاشقانه‌ای را که برای کاربر شناخته‌شده نباشد، به او پیشنهاد دهد، امّا پیشنهاد فیلمی که درست در این نوع کلی جای نمی‌گیرد، دشوار است. برای مثال، یک فیلم طنز در تقلیدی از فیلم‌های ترسناک را نمی‌تواند در دستهٔ به‌خصوصی جای دهد. در برابر این روش، الگوریتم‌های همسایه‌محور ارتباطات محلی داده‌ها را نیز در نظر می‌گیرند؛ در نتیجه، یک سامانهٔ توصیه‌گر فیلم با این الگوریتم می‌تواند یک فیلم متفاوت از سلایق پیشین کاربر به او پیشنهاد دهد[2] .

روش‌های همسایه‌محور شهودی‌اند و پیاده‌سازی به‌نسبت ساده‌تری دارند. و در ساده‌ترین حالت الگوریتم، تنها به تنظیم شاخص تعداد کاربران همسایه برای تولید پیشنهادها نیاز دارند [2].

روش‌های همسایه‌محور برخلاف روش‌های الگو محور نیازی به مرحلهٔ آموزش که به‌طور معمول برای الگوریتم‌ها پرهزینه است، ندارند. این ویژگی برای الگوریتم‌هایی که نیاز به اجرا در بازه‌های زمانی متناوب در کاربردهای تجاری بزرگ دارند، مفید است. به‌دلیل‌این‌که مرحلهٔ تولید پیشنهادها به نسبت روش‌های الگو محور پرهزینه است، می‌توان نزدیک‌ترین همسایه‌های کاربر را در مرحلهٔ برون‌خط از پیش محاسبه کرد تا تولید پیشنهادها زمان کمتری نیاز داشته باشد. ذخیرهٔ این همسایه‌ها نیاز به حافظهٔ کمی دارد و استفاده از روش‌های همسایه‌محور را برای کاربردهایی با میلیون‌ها کاربر و آیتم امکان‌پذیر می‌کند [2].

برتری دیگر سامانه‌های توصیه‌گر همسایه‌محور این است که این سامانه‌ها با اضافه شدن دائمی کاربرها و آیتم‌ها و همچنین، وارد شدن امتیازدهی‌ها در سامانه، تأثیر اندکی می‌پذیرند. برای مثال، وقتی تعداد کمی امتیاز برای یک آیتم جدید وارد شده است، برخلاف روش‌های الگومحور، بدون نیاز به آموزش دادن دوبارهٔ سامانه، فقط نیاز است که شباهت بین این آیتم و آیتم‌های موجود در سامانه محاسبه شود [2].

۲-۲- اطمینان

به‌دلیل‌این‌که تعداد کاربران و آیتم‌ها در سامانه‌های توصیه‌گر تجاری اینترنتی بسیار زیاد است، حتی کاربران بسیار فعال در این سامانه‌ها تنها می‌توانند به تعداد بسیار کمی از آیتم‌های مجموعه داده‌های سامانه امتیازدهی کنند؛ که این موضوع سبب پراکندگی در مجموعه داده‌ها می‌شود. به‌دلیل پراکندگی زیاد داده‌ها، سامانهٔ توصیه‌گر ممکن است نتواند شباهت بین دو کاربر را تعیین کند و یا وقتی‌که محاسبهٔ شباهت بین دو کاربر امکان‌پذیر است، به‌دلیل کافی نبودن اطّلاعات پردازش‌شده، این شباهت قابل اطمینان نباشد؛ به همین دلیل الگوریتم فیلترینگ مشارکتی در این مجموعه داده، غیرقابل استفاده می‌شود [7].

[18] Decision tree

روش‌های بسیاری برای حل این مشکل ارائه شده است. مرجع [7] روشی ارائه می‌دهد که در آن ویژگی تراگذر اطمینان بین کاربران در یک شبکهٔ اجتماعی تعریف می‌شود که اطّلاعات جدیدی در سامانه تولید می‌کند. پژوهش‌ها نشان داده‌است که در نظر گرفتن روابط اطمینان بین کاربران در سامانه‌های توصیه‌گر و همچنین، روابط حاصل از استنتاج اطمینان میان کاربرانی که به‌صورت مستقیم به یکدیگر اطمینان ندارند، منبعی ارزشمند از اطّلاعات برای حل مسئلهٔ پراکندگی داده‌ها و بعضی چالش‌های دیگر در سامانه‌های توصیه‌گر، در اختیار ما قرار می‌دهد [13]. هرچه شباهت دو کاربر به یکدیگر بیشتر باشد، اطمینان بیشتری بین آن‌ها وجود دارد و می‌توان این اطمینان را بین کاربران منتشر و تجمیع کرد تا شبکهٔ اطمینان بین کاربران ایجاد شود [14].

در یک سامانهٔ توصیه‌گر می‌توان اطمینان میان دو کاربر را که به آیتم‌های مشترکی امتیازدهی کرده‌اند، با استفاده از معیار محاسبهٔ اطمینان تعیین‌شده برای سامانه به‌دست آورد. به‌دلیل‌این‌که اطمینان در زمینهٔ شباهت تعریف‌شده است، هرچه دو کاربر به یکدیگر بیشتر شبیه باشند، اطمینان بین آن‌ها بیشتر در نظر گرفته می‌شود [7].

۲-۲-۱- کارهای مرتبط با اطمینان استنتاج‌شده در سامانه‌های توصیه‌گر

از جمله مقالاتی که از محاسبهٔ اطمینان برای حل چالش شروع سرد استفاده کرده‌اند، می‌توان به [7] و [15] اشاره کرد. روش استفاده‌شده در [7] اطمینان را بر اساس شباهت Pearson بین کاربران به دست می‌آورد و مقدار این شباهت را به‌عنوان اطمینان بین دو کاربر در نظر گرفته و آن را در شبکه منتشر می‌کند.

در روش استفاده‌شده در [15] مقداری دودویی برای اطمینان تولید می‌شود که با اعمال آستانه بر میزان شباهت Pearson به دست می‌آید. این روش قابل انتشار در شبکه بوده و متقارن است.

۲-۳- اجتماع نتایج[19] در سامانه‌های توصیه‌گر

روش‌های مختلفی برای پیشنهاد آیتم‌ها به کاربران در پژوهش‌های گذشتهٔ پژوهشگران سامانه‌های توصیه‌گر ارائه شده است؛ تحلیل نتایج تولیدشده برای پیشنهاد N آیتم برتر توسط این روش‌ها، نشان می‌دهد که به‌طور معمول این روش‌ها بر روی این نکته که کدام آیتم‌ها باید در آغاز فهرست پیشنهادها قرار بگیرند، توافق ندارند. زمانی که رتبه‌بندی آیتم‌ها توسط روش‌ها به اندازه کافی متفاوت باشد، می‌توان به‌جای پیدا کردن بهترین روش برای حل مسئله، نتایج این روش‌ها را در کنار هم قرار داد تا بهترین ویژگی‌های روش‌ها ترکیب شوند و فهرست جامعی از پیشنهادها به کاربر ارائه شود [4].

در سامانه‌های توصیه‌گر، اجتماع نتایج برای ایجاد توافق در نظرات مختلف الگوریتم‌های توصیه استفاده می‌شود تا فهرستی مرتب‌سازی‌شده از آیتم‌ها که از زیرمجموعه‌ای از آیتم‌های هر روش تشکیل شده‌است، به کاربر پیشنهاد شود.

مجموعه $T = \{\tau^1.\tau^2.\cdots.\tau^K\}$ از فهرست‌هایی شامل آیتم‌های مرتب‌شده توسط الگوریتم‌های $S = \{S^1.S^2.\cdots.S^K\}$ به دست آمده است که در آن k نشان‌دهندهٔ تعداد روش‌ها و در نتیجه، تعداد فهرست‌های آیتم‌ها است. در این دو مجموعه بین فهرست‌های آیتم‌ها و الگوریتم‌ها تناظر وجود دارد و فهرست τ^i توسط الگوریتم S^i تولید شده است. در مسئلهٔ اجتماع نتایج در سامانهٔ توصیه‌گر هرکدام از الگوریتم‌ها در مجموعه‌ی S، الگوریتم پیشنهاد N آیتم برتر هستند. این الگوریتم‌ها ماتریس امتیازدهی $M_{m\times n}$ را به‌عنوان ورودی دریافت می‌کنند. هر ردیف در ماتریس M نشان‌دهندهٔ یک کاربر و هر ستون نشان‌دهندهٔ یک آیتم است. مقدار M_{ij} در این ماتریس نشان‌دهندهٔ مقدار امتیازدهی کاربر i به آیتم j است. برای تولید اجتماع نتایج R، ماتریس M به‌عنوان ورودی به هریک از الگوریتم‌های توصیه در مجموعهٔ S داده می‌شود و سپس، نتایج این روش‌ها ترکیب می‌شود [16].

۲-۳-۱- کارهای مرتبط با اجتماع نتایج در سامانه‌های توصیه‌گر

با اینکه روش‌های ارائه‌شدهٔ بسیاری به‌عنوان اجتماع نتایج در حوزهٔ سامانه‌های توصیه‌گر وجود ندارد، بعضی از سامانه‌های توصیه‌گر ترکیبی[20] در واقع نتایج را ترکیب می‌کنند. برای مثال، در [17] سامانهٔ توصیه‌گر ترکیبی معرفی شده است که برای افزایش دقّت، نوآوری و تنوع، نتایج حاصل از الگوریتم‌های توصیه‌گر را ترکیب می‌کند.

[19] Rank aggregation

[20] Hybrid

در این مقاله ترکیب روش‌ها، با ترکیب وزن‌دار خطی خروجی الگوریتم‌های توصیه انجام می‌شود که در آن وزن‌ها با فرایند تکاملی ارائه‌شده به دست می‌آیند.

با استفاده از روش‌های تکاملی، در مقالات [4] و [18] روش‌های اجتماع رتبه‌بندی برای ترکیب نتایج تولیدشده از سامانه‌های توصیه‌گر N آیتم برتر[21] ارائه شده است. [18] از برنامه‌نویسی ژنتیک برای تکامل جمعیّتی از توابع جمعی[22] استفاده می‌کند. از توابع جمعی برای امتیازدادن به آیتم‌های ورودی استفاده می‌شود و سپس، برای تولید اجتماع رتبه بندی، آیتم‌ها بر اساس امتیازهایشان مرتب می‌شوند.

مرجع [4] برای تولید یک سامانهٔ توصیه‌گر N آیتم برتر، روشی تکاملی برای ترکیب نتایج روش‌های توصیه‌گر پیشنهاد می‌دهد تا انتخاب روش را خودکار کند و روشی با خطای کمتری در پیشنهادها به دست آورد. میزان این خطا در این مقاله با RMSE مقایسه، و روش پیشنهادی آن که با ترکیب روش‌های فیلترینگ مشارکتی ایجاد شده است، خطای کمتری در مقایسه با سامانه توصیه‌گر با استفاده از فقط یک روش، دارد. این مقاله از الگوریتم ژنتیک برای اجتماع نتایج روش‌های فیلترینگ مشارکتی همسایه‌محور استفاده می‌کند. الگوریتم ژنتیک تعداد آیتم‌های انتخاب‌شده را از بالای فهرست تولیدشده توسط هر روش، برای تولید اجتماع رتبه‌بندی‌ها مشخص می‌کند. همچنین، این مقاله نشان می دهد که با روش ارائه‌شدهٔ آن، حتی در حالتی که کاربر تعداد کمی امتیاز به آیتم‌ها داده‌است، به نسبت سامانه‌های توصیه‌گر با استفاده از فقط یک روش، افت کیفیت کمتری دارد.

۲-۴- الگوریتم ژنتیک

الگوریتم ژنتیک از نظریهٔ تکامل داروین الهام گرفته شده و در آن حفظ سازگارترین موجودات و ژن‌های آن‌ها شبیه‌سازی شده است. الگوریتم ژنتیک، الگوریتمی جمعیّت‌محور[23] است. در این روش هر راه‌حل متناظر با یک کروموزوم و هر ژن نشان‌دهندهٔ یک شاخص است. این الگوریتم میزان سازگاری[24] هر موجود از جمعیت را با کمک تابع ارزیابی[25] محاسبه می‌کند. برای بهبود راه‌حل‌های ضعیف، راه‌حل‌ها با یک سازوکار انتخاب[26] به‌صورت تصادفی انتخاب و ترکیب می‌شوند. این روش راه‌حل‌های بهتر را با احتمال بیشتری انتخاب می‌کند؛ زیرا احتمال یک راه‌حل انتخاب در این روش متناسب با میزان سازگاری آن است. الگوریتم ژنتیک با حفظ بهترین راه‌حل‌ها در هر نسل و استفاده از آن‌ها برای تولید نسل بعدی می‌تواند راه‌حل بهینهٔ یک مسئله را تخمین بزند؛ زیرا با این روش تکاملی جمعیّت در هر نسل بهتر می‌شود و در نهایت با همگرا شدن همهٔ راه‌حل‌ها می‌توان به یک نتیجه مناسب رسید [19].

این الگوریتم از گام‌های زیر تشکیل می‌شود [4]:

۱. تولید جمعیّت اولیه کروموزم‌ها
۲. ارزیابی کروموزم‌ها در جمعیت کنونی
۳. انتخاب کروموزم‌های والد برای تولید کروموزم‌های جدید
۴. اعمال اپراتورهای ژنتیک بر روی والدین برای تولید کروموزم‌های جدید
۵. از بین بردن جمعیّت قبلی
۶. ارزیابی کروموزم‌های جدید

این الگوریتم تا زمانی که به تعداد نسل موردنظر برسد، یا بهترین کروموزم را بیابد، از مرحلهٔ سه تکرار می‌شود.

۲-۴-۱- کارهای مرتبط با الگوریتم ژنتیک در سامانه‌های توصیه‌گر

در سامانه‌های توصیه‌گر از الگوریتم ژنتیک به شکل‌های مختلفی استفاده شده است. برای مثال، در [20] یک سامانه توصیه‌گر بر پایهٔ الگوریتم ژنتیک ارائه شده است که به‌جای ارزیابی آیتم‌ها، فهرست آیتم‌ها را ارزیابی، و سپس، فهرست بهینهٔ پیشنهادها را توصیه می‌کند. این روش از الگوریتم ژنتیک برای پیداکردن بهترین فهرست برای پیشنهاد دادن به کاربر هدف بهره می‌گیرد.

در [21] روشی پیشنهاد شده‌است که توسط آن روش فیلترینگ مشارکتی و الگوریتم ژنتیک ترکیب می‌شوند. این روش الگوریتم ژنتیک را برای محاسبه مقدار بهینه شباهت بین کاربران استفاده می‌کند و هر کروموزوم در جمعیت نشان‌دهندهٔ یک ماتریس شباهت بین کاربران است. روش پیشنهادی به‌صورت مستقیم هیچ معیار شباهتی را محاسبه نمی‌کند و تنها شباهت بین کاربران را یاد می‌گیرد.

در [22] برای یافتن تابع محاسبه شباهت بهینه، از الگوریتم ژنتیک برای بهین سازی وزن‌ها در این تابع استفاده شده است. در این مقاله یک سامانه فیلترینگ مشارکتی آیتم‌محور ارائه شده که با الگوریتم ژنتیک

[21] Top-N recommendation
[22] Aggregation functions
[23] Population based
[24] Fitness
[25] Fitness function
[26] Selection

ترکیب شده و از اطّلاعات امتیازدهی کاربران استفاده می‌کند.

۲-۵- مقایسه روش پیشنهادی و کارهای پیشین

در این پژوهش با هدف بهبود روش ارائه‌شده در [4] با اعمال تغییراتی در بخش الگوریتم ژنتیک و روش‌های استفاده‌شده برای تولید پیشنهادها، تلاش شده‌است تا با حفظ کارایی الگوریتم برای سامانه‌های برخط، که با تغییرات اساسی مجموعه‌داده‌ها نیاز به اجرای سریع دوبارهٔ الگوریتم برای پیداکردن ترکیب مناسب روش‌ها دارند، میزان دقّت پیشنهادها در توصیه‌گر N آیتم برتر افزایش داده‌شود. به‌طورکلی نوآوری‌های انجام‌شده برای افزایش دقّت سامانهٔ توصیه‌گر پیشنهادی به‌صورت زیر است:

- **بهبود تابع ارزیابی در الگوریتم ژنتیک**

هدف الگوریتم ژنتیک، پیداکردن جوابی مناسب برای مسئله است. در این پژوهش، مسئله، ایجاد ترکیب مناسب نتایج الگوریتم‌های توصیه، با هدف تولید فهرست آیتم‌ها با کمترین خطای پیش‌بینی است؛ بنابراین، باید معیاری از معیارهای ظرفیت تصمیم‌گیری موفّق در سامانهٔ توصیه‌گر به‌عنوان تابع ارزیابی الگوریتم ژنتیک به‌کار گرفته شود. به همین جهت در روش پیشنهادی به‌عنوان تابع ارزیابی الگوریتم ژنتیک، معیار precision استفاده شده است و میزان precision خروجی هر راه‌حل که الگوریتم ژنتیک تولید می‌کند، برای همهٔ کاربران به دست می‌آید و با سایر راه‌حل‌های تولیدشدهٔ الگوریتم ژنتیک مقایسه می‌شود تا در نهایت، با این الگوریتم تکاملی به جوابی مناسب برسیم.

- **افزایش گوناگونی روش‌های پایه برای تولید پیشنهادها**

اجتماع نتایج الگوریتم‌های توصیه با روش‌های گوناگون نتیجه بهتری دارد؛ زیرا در این حالت، این روش‌ها می‌توانند پیشنهادهای متفاوتی تولید کنند. این موضوع کیفیت اجتماع نتایج را افزایش می‌دهد، زیرا درغیراین‌صورت، در اجتماع نتایج با وجود آیتم‌های تکراری در فهرست پیشنهادها، مجبور به حذف آن‌ها و استفاده از آیتم‌هایی با کیفیت پایین‌تر که در پایان فهرست‌ها قرارگرفته‌اند، می‌شویم. به همین دلیل، روش‌های حافظه‌محور استفاده‌شده برای تولید پیشنهادها با اضافه کردن دو روش دیگر، افزایش داده شده و حالت‌های کاربرمحور و آیتم‌محور برای هر روش در نظر گرفته شده است. در صورت استفاده از همهٔ حالت‌های ممکن در الگوریتم ژنتیک، برای اجرای الگوریتم به زمان زیادی نیاز است؛ که این حالت برای سامانه‌های برخط مطلوب نخواهد بود؛ پس برای انتخاب بهترین روش‌ها برای استفاده در الگوریتم ژنتیک، دقّت یک سامانهٔ توصیه‌گر فیلترینگ مشارکتی با استفاده از روش‌های مختلف حافظه‌محور، در حالت کاربرمحور و آیتم‌محور محاسبه می‌شود و پس‌از محاسبهٔ دقّت برای هر حالت، برای ایجاد تنوّع کافی در پیشنهادهای تولیدشده، از هر روش بهترین حالت (حالتی که بالاترین دقّت را دارد) انتخاب می‌شود.

- **رفع مشکل پراکندگی داده‌ها با استفاده از اطمینان استنتاج‌شده بین کاربران**

برای رفع پراکندگی داده‌ها با کمک معیار اطمینان می‌توان با استفاده از اطّلاعات امتیازدهی کاربران، میزان اطمینان آن‌ها به یکدیگر را به دست آورد و با استفاده از ویژگی تراگذر بودن اطمینان و انتشار اطمینان در شبکه، میزان اطمینان استنتاج‌شده بین این کاربران، که در حالت معمول شباهتی برایشان محاسبه نمی‌شود، به دست آورد و از این اطّلاعات برای تولید پیشنهادها استفاده کرد.

- **ایجاد ترتیب در پیشنهادهای روش‌های پایه**

در مرحلهٔ ارزیابی الگوریتم ژنتیک، از بالای فهرست پیشنهادهای تولیدشدهٔ الگوریتم‌های توصیه‌های مختلف، تعداد مشخصی آیتم برای هر کاربر انتخاب می‌شود و در مرحلهٔ ارزیابی الگوریتم ژنتیک، امتیاز آیتم‌های انتخاب‌شده برای هر کاربر پیش‌بینی می‌شود؛ سپس، آیتم‌ها بر اساس مقدار این امتیازها مرتب شده و از بالای فهرست آن‌ها به تعداد مشخص‌شده در هر ژن، تعدادی پیشنهاد برای هر کاربر انتخاب می‌شود، پس هرچه مواردی که در ابتدای فهرست آیتم‌های هر روش قرار دارند با احتمال بیشتری موردپسند کاربر باشد، نتیجه بهتر خواهد بود. به همین دلیل، برای افزایش دقّت پیشنهادها، آن‌ها را با روشی مرتب می‌کنیم تا آیتم‌هایی که مناسب‌تر هستند، در فهرست پیشنهادها بالاتر قرار گیرد و احتمال انتخاب آن بیشتر باشد.

۳- روش پیشنهادی

در این بخش روش پیشنهادی را برای تولید فهرست Nتایی از آیتم‌ها برای پیشنهاد به کاربر، با هدف پیدا کردن بهترین ترکیب روش‌ها برای افزایش دقّت فهرست پیشنهادها، بررسی می‌کنیم.

در این مسئله از الگوریتم ژنتیک برای بهینه‌سازی ترکیب نتایج روش‌های تولید پیشنهادها استفاده شده است تا دقّت فهرست نهایی به بهترین مقدار ممکن نزدیک شود. در واقع، در این مقاله هدف الگوریتم ژنتیک، یافتن یک راه‌حل برای حل مسئله پیداکردن تعداد مناسبِ آیتم‌هایی است که از بالای فهرست‌های تولیدشدهٔ هر الگوریتم فیلترینگ مشارکتی انتخاب می‌شود.

در آغاز، الگوریتم ژنتیک جمعیت اولیّه را تولید می‌کند. هر کروموزم با تابع ارزیابی بررسی، و مقدار دقّت آن محاسبه می‌شود. دقّت هر کروموزوم با تولید پیشنهادهای مشخص‌شده توسط آن برای هرکاربر و ارزیابی این پیشنهادها به دست می‌آید. الگوریتم ژنتیک تلاش می‌کند تا کروموزومی پیدا کند که اعداد مشخص شده در ژن‌های آن، مناسب‌ترین تعداد آیتم‌ها را از هر الگوریتم توصیه انتخاب کند.

در هر کروموزوم، تعداد ژن‌ها برابر تعداد روش‌های انتخاب‌شده برای تولید پیشنهادها در سامانهٔ توصیه‌گر است و مقدار هر ژن، نشان‌دهندهٔ تعداد آیتم‌های انتخاب‌شده از بالای هر فهرست پیشنهادهاست که با سامانهٔ توصیه‌گر مربوط به آن ژن تولید شده است.

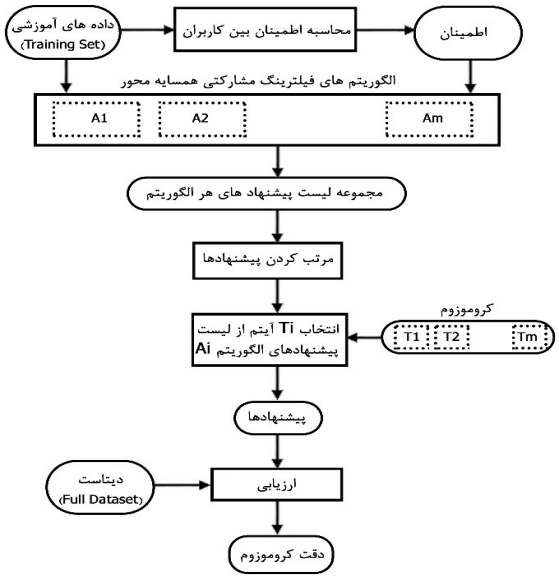
در بخش ارزیابی الگوریتم، فهرست آیتم‌های پیشنهادی بر اساس کروموزوم در حال ارزیابی تولید می‌شود که فرایند کلی آن بر اساس (شکل-۱) است.

این فهرست دربرگیرندهٔ بهترین آیتم‌ها از روش‌های فیلترینگ مشارکتی همسایه‌محور آیتم‌محور یا کاربرمحور گوناگونی است که هریک، از روش محاسبهٔ شباهت متفاوتی برای تعیین همسایه‌های یک کاربر یا آیتم استفاده می‌کنند.

برای رفع مشکل پراکندگی داده‌ها، در مواردی که الگوریتم توصیه‌گر توانسته است با داشتن اطّلاعات اضافهٔ اطمینان استنتاج‌شده بین کاربران بهتر عمل کند، مقدار اطمینان نیز در روابط کاربران لحاظ شده است.

به‌دلیل‌این‌که فهرست نهایی طول مشخصی دارد و هرچه آیتم‌های بالای فهرست دقّت بیشتری داشته باشند، فهرست نهایی بهترخواهد بود، پیشنهادهای حاصل از هر روش را به‌صورتی مرتب سازی کرده‌ایم که احتمال درست بودن آیتم‌هایی بالای هر فهرست بیشتر باشد.

در (شکل-۱) کروموزم از m ژن تشکیل شده است که در فرایند ارزیابی کروموزوم برای یک کاربر، به تعداد T1 پیشنهاد از فهرست پیشنهادهای تولید شده توسط روشA1 انتخاب می‌شود و به همین صورت تا ژن mام فرایند انتخاب آیتم‌ها از فهرست‌ها ادامه پیدا می‌کند و سپس، فهرست آیتم‌های انتخاب‌شده برای کاربر ارزیابی می‌شود. در هر کروموزم در این مسئله، مقدار مجموع اعداد ژن‌ها باید برابر N باشد تا فهرست Nتایی از آیتم‌ها به ازای هر کروموزم تولید شود.



(شکل-۱): فرایند کلی ارزیابی کروموزوم در روش پیشنهادی
(Figure-1): General flow evaluating a chromosome in the proposed approach

پس‌از ارزیابی تمامی کروموزم‌ها در جمعیت، الگوریتم ژنتیک بر اساس مقادیر به‌دست‌آمدهٔ دقّت هر کروموزم، برخی کروموزوم‌ها را برای تولید نسل بعدی انتخاب می‌کند و با ترکیب هر دو کروموزوم (Crossover) یا تغییر کروموزومها (Mutation) نسل بعد را ایجاد می‌کند. همچنین، تعدادی کروموزوم از بهترین نتایج هر نسل بدون تغییر به نسل بعد وارد می‌شوند. پس‌از تولید نسل جدید، کروموزوم‌های آن ارزیابی می‌شوند و این فرایند تا رسیدن به کروموزوم مناسب تکرار می‌شود.

۳-۱- معیارهای ارزیابی سامانه‌های توصیه‌گر

برای انتخاب تابع ارزیابی الگوریتم ژنتیک معیارهای ارزیابی سامانه‌های توصیه‌گر بررسی شده است تا مناسب ترین معیار برای تابع ارزیابی در الگوریتم ژنتیک انتخاب شود تا بتوان با استفاده از کروموزم به‌دست آمده توسط الگوریتم ژنتیک، فهرستی از پیشنهادها برای هرکاربر تولید کرد که برای آن‌ها مطلوب باشد.

معیارهای ارزیابی سامانه‌های توصیه‌گر را می‌توان بر اساس روشی که مقدار کمّی صحیح بودن رفتار سامانه توصیه‌گر را تعیین می کنند، در سه دسته قرارداد [23]:

۱) پیش بینی امتیازدهی

این معیارها، توانایی سامانه توصیه‌گر برای پیش‌بینی امتیاز هرکاربر به آیتم‌ها را اندازه‌گیری می‌کنند. مانند MAE[1] که توسط (رابطه -۳) محاسبه می‌شود.

$$MAE = \frac{\sum_{u,i}|p(u,i) - P(u,i)|}{N} \qquad (۳)$$

در این رابطه P برابر مقدار پیش‌بینی‌شده و p مقدار واقعی امتیاز کاربر u به آیتم i است. این معیار اختلاف بین امتیازدهی پیش‌بینی‌شده و امتیاز حقیقی کاربر را محاسبه می‌کند.

۲) ظرفیت تصمیم‌گیری موفق

این معیارها، توانایی سامانهٔ توصیه‌گر برای تصمیم‌گیری درست را اندازه‌گیری می‌کنند. مانند دقّت[1] که با (رابطه -۴) و recall که با (رابطه -۵) محاسبه می‌شود و همچنین، معیارهای ارزیابی بازیابی اطّلاعات.

در این رابطه‌ها منظور از پیشنهادهای درست، تعداد آیتم‌هایی است که در فهرست پیشنهادهای کاربر، مورد پسند او هستند و تعداد آیتم‌های مطلوب برابر کل آیتم‌ها در مجموعه‌دادگان است که موردعلاقهٔ کاربر هستند.

۳) پیش‌بینی رتبه‌بندی

این معیارها، توانایی سامانه توصیه‌گر برای پیش‌بینی رتبه‌بندی مجموعه‌ای از آیتم‌ها توسط کاربر را اندازه‌گیری می‌کنند.

۳-۱-۱- انتخاب تابع ارزیابی الگوریتم ژنتیک

هدف الگوریتم ژنتیک، پیداکردن جوابی مناسب برای مسئله است. در این مقاله مسئله، ترکیب روش‌های توصیه با هدف تولید فهرست آیتم‌ها، با کمترین خطای پیش بینی آیتم‌ها است. پس هدف این سامانهٔ توصیه‌گر تولیدN آیتم برتر، یافتن آیتم‌هایی است که کاربر به آن‌ها علاقه دارد.

با مجموعه‌دادگان موجود که در آن امتیازدهی کاربران به آیتم‌ها، با عدد مشخص شده است، می‌توان بالاترین امتیازها را نشانهٔ علاقهٔ کاربر به آن آیتم دانست؛ پس اگر چنین آیتم‌هایی در فهرست پیشنهادی وجود داشته باشد، روش به‌درستی کار کرده است. برای رسیدن به این حالت، باید معیاری از معیارهای ظرفیت تصمیم‌گیری موفق در سامانهٔ توصیه‌گر انتخاب کرد تا به‌عنوان معیاری برای ارزیابی خروجی این سامانه و همین‌طور به‌عنوان تابع ارزیابی الگوریتم ژنتیک به‌کار گرفته شود.

افزایش دقّت به معنای افزایش تعداد آیتم‌های مطلوب کاربر در فهرست پیشنهادها و افزایش recall به معنای پیشنهادکردن تعداد بیشتری از آیتم‌های موردعلاقهٔ کاربر است که به‌طور معمول با افزایش تعداد پیشنهادها می‌توان آن را افزایش داد. در حالتی که سامانهٔ توصیه‌گر تعداد ثابتی آیتم می‌تواند به کاربر پیشنهاد کند، هدف این است که همهٔ آیتم‌ها موردعلاقهٔ کاربر باشند و درصورتی‌که آیتمی نامطلوب در این فهرست باشد، در برابر اینکه نسبت کمتری از همهٔ آیتم‌های موردعلاقه کاربر پیدا شوند، بدتر است. پس بهتر است که در این حالت دقّت افزایش پیدا کند.

بر اساس دلایل بررسی‌شده، به‌عنوان تابع ارزیابی الگوریتم ژنتیک در این سامانهٔ توصیه‌گر، معیار دقّت استفاده شده است که در آن آیتم‌هایی را که کاربر به آن-ها امتیاز بالاتر از حد آستانه مشخص شده داده است، صحیح می‌دانیم. مقدار این حدِ آستانه باتوجّه‌به امتیازهای ممکن در مجموعه داده‌ها انتخاب می‌شود. این معیار به‌عنوان تابع ارزیابی الگوریتم ژنتیک استفاده می‌شود که در آن مقدار سازگاری هر راه‌حل را، بر اساس دقّت فهرست پیشنهادها (با استفاده از رابطه ۴) که توسط کروموزوم آن راه‌حل برای هرکاربر مشخص می‌شود، محاسبه می کند.

$$precision = \frac{\text{تعداد پیشنهادهای درست}}{\text{تعدادپیشنهادها}} \qquad (۴)$$

۳-۲- تولید و مرتب کردن پیشنهادها

امتیاز آیتم‌های تولیدشده با هر یک از روش‌های مشخص شدهٔ هر ژن، در مرحلهٔ ارزیابی الگوریتم ژنتیک، برای هر کاربر پیش‌بینی می‌شود؛ سپس، آیتم‌ها بر اساس مقدار این امتیازها مرتب شده و از بالای فهرست آن‌ها به تعداد مشخص‌شده در هر ژن، تعدادی پیشنهاد برای هر کاربر انتخاب می‌شود.

هرچه مواردی که در ابتدای فهرست آیتم‌های هر روش قرار دارند بهتر باشد، نتیجه بهتر خواهد بود. به همین دلیل برای افزایش دقّت، پیشنهادها بر اساس میزان معروف بودن و امتیاز پیش‌بینی‌شده‌شان توسط سامانهٔ توصیه‌گر، مرتب می‌شود؛ به این صورت که اگر امتیاز یک آیتم بیشتر باشد، بالاتر قرار می‌گیرد و اگر امتیاز دو آیتم یکسان باشد، آیتمی که تعداد بیشتری از کاربران به آن امتیاز داده‌اند، در فهرست توصیه‌ها بالاتر می‌رود.

۳-۳- استفاده از اطمینان برای رفع پراکندگی داده‌ها

برای محاسبه اطمینان بین کاربران در سامانه‌های توصیه‌گر، از روش‌های گوناگون محاسبه شباهت بین کاربران استفاده شده است مانند:

Pearson correlation, cosine vector similarity, Spearman correlation و روش‌های دیگر محاسبه شباهت [7].

این روش‌های محاسبهٔ شباهت برای پیداکردن نزدیک‌ترین همسایه‌ها در روش پیشنهادی نیز استفاده شده‌اند، پس می‌توان با استفاده از همان معیاری که در سامانهٔ توصیه‌گر شباهت میان کاربران را برای انتخاب همسایه‌ها محاسبه می‌کند، شبکه اطمینان را نیز تشکیل داد تا این شبکه حالتی یک‌پارچه پیدا کند؛ به معنی این که با استفاده از معیاری از جنس معیار انتخاب‌شده برای انتخاب همسایه‌ها برای تولید پیشنهادها، اطمینان بین کاربران نیز محاسبه می‌شود. در روش پیشنهادی نیز به همین صورت شبکهٔ اطمینان ایجاد می‌شود.

برای محاسبهٔ اطمینان بین کاربرانی که به‌طور مستقیم به یکدیگر اطمینان ندارند -آیتم‌های یکسانی را امتیازدهی نکرده‌اند- باید مقادیر اطمینان بین این کاربران را به‌ازای کاربران میانی مختلف محاسبه و مقادیر آن‌ها را تجمیع کرد تا مقدار اطمینان استنتاج‌شده بین آن‌ها به دست بیاید.

$$recall = \frac{\text{تعداد پیشنهادهای‌درست}}{\text{تعدادآیتم‌های مطلوب}} \qquad (۵)$$

$$\Gamma_{S\xrightarrow[N]{}T} = \frac{n(I_S \cap I_N)}{n(I_S \cap I_N)+n(I_N \cap I_T)}\left|T_{S\longrightarrow N}\right| + \frac{n(I_N \cap I_T)}{n(I_S \cap I_N)+n(I_N \cap I_T)}\left|T_{N\longrightarrow T}\right| \qquad (۶)$$

به کمک (رابطه-۶) که شکل بدون اعمال ضرایب بی‌اعتمادی[27] کاربران به یکدیگر در رابطه به کار رفته در [7] است، می‌توان اطمینان اطمینان استنتاج‌شده بین دو کاربرS و T با یک کاربر میانی N را به دست آورد.

مقدار $T_{S\xrightarrow[N]{}T}$ برابر میزان اطمینان بین دو کاربر S و T از مسیر کاربر میانی N است. $n(I_S \cap I_N)$ برابر تعداد آیتم‌هایی است که کاربران S,N به آن به‌طور مشترک امتیاز داده‌اند.

این رابطه اطمینان بین کاربر را S و T که توسط کاربر N به یکدیگر به‌صورت غیرمستقیم، ارتباط دارند، با میانگین وزنی مقادیر اطمینان مستقیم کاربر S به N و اطمینان مستقیم کاربر N به T با ضریب تعداد آیتم‌هایی که هر زوج کاربر با اطمینان مستقیم به‌طور مشترک، امتیازدهی کرده‌اند، محاسبه می‌کند.

روش‌های مدیریت چند مسیر اطمینان و به دست آوردن اطمینان کلی بین دو کاربر شامل دو دستهٔ اصلی (۱) ترکیب مسیرها[28] و (۲) انتخاب مسیر[29]است. در روش ترکیب مسیرها مقادیر چند مسیر اطمینان با یکدیگر ترکیب شده و یک مقدار برای اطمینان تولید می‌کنند؛ در حالی که در روش انتخاب مسیر مطمئن‌ترین مسیر از بین مسیرها انتخاب می‌شود [7].

برای حل مشکل پراکندگی داده‌ها نیاز به تولید اطمینانی داریم که بتواند نشانگر اطمینان ضمنی بین دو کاربر باشد و انتخاب یکی از مسیرها به‌عنوان نماینده از میان مسیرهای مختلف بین دو کاربر، دقّت لازم را ندارد. به همین دلیل از روش ترکیب مسیرها روش میانگین ترکیب[30] انتخاب شده است که با (رابطه- ۷) به دست می‌آید و در روش پیشنهادی از این رابطه برای به دست آوردن اطمینان استنتاج‌شده استفاده شده است.

[27] Distrust
[28] Path Composition
[29] Path Selection
[30] Composition

$$T_{S\to T}=\frac{\sum_{i=1}^{p} T_{S\xrightarrow[Ni]{}T}}{p} \quad (۷)$$

در این رابطه p تعداد مسیرهای جدا میان دو کاربر S و T است. با این روش میزان اطمینان استنتاج‌شده بین دو کاربر با میانگین گرفتن از مقادیر اطمینان که از تمام مسیرهای ممکن بین دو کاربر وجود دارد، محاسبه می‌شود [7].

۳-۴- انتخاب روش برای هر ژن

استفاده از همهٔ روش‌های محاسبهٔ شباهت در حالت‌های کاربرمحور و آیتم‌محور به‌عنوان یک روش مربوط به یک ژن در الگوریتم ژنتیک، سبب می‌شود که کروموزوم ایجادشده بسیار بزرگ شود و رسیدن به هم‌گرایی الگوریتم ژنتیک نیاز به ساعت‌ها اجرا داشته باشد که این زمان برای به‌روزرسانی یک سامانهٔ توصیه‌گر و مشخص‌کردن ژن برتر در سامانه‌های اینترنتی که در آن‌ها کاربران به‌صورت برخط با سامانه تعامل دارند و باید سرعت پاسخ‌دهی سامانه مناسب باشد، زمان زیادی است. به همین دلیل بر اساس توانایی هریک از معیارها به‌صورت مجزا، روش‌های برتر انتخاب شده و در الگوریتم ژنتیک به آنها یک ژن اختصاص داده می‌شود.

برای پیدا کردن بهترین روش‌ها، دقّت یک سامانهٔ توصیه‌گر فیلترینگ مشارکتی با استفاده از هریک از روش‌های محاسبهٔ شباهت، در حالت کاربرمحور و آیتم‌محور و همچنین، با اضافه کردن اطمینان استنتاج‌شده در حالت کاربرمحور محاسبه می‌شود؛ به این صورت که پیشنهادها در هر فهرست با روش مرتب‌سازی توضیح‌داده‌شده (بخش۲-۳) مرتب شده و دقّت در پیشنهاد دادن هر روش، بر اساس پیشنهاد دادن تعداد N/2 آیتم بالای فهرست به کاربران، محاسبه می‌شود.

انتخاب تعداد N/2 آیتم به این دلیل است که در فهرست نهایی Nتایی پیشنهادها که توسط الگوریتم ژنتیک مشخص می‌شود، به‌طور معمول حداکثر تعداد N/2 آیتم از هر روش انتخاب می‌شود و مقدار دقّت برای تعداد N/2 آیتم پیشنهاد می‌تواند معیار بهتری به نسبت N پیشنهاد، برای این کاربرد باشد.

برای بررسی تأثیر اضافه شدن اطمینان استنتاج‌شدهٔ کاربران به نتایج اطمینان استنتاج‌شده بین کاربرانی که با یک کاربر میانی با یکدیگر ارتباط دارند، به دست آورده‌شده‌است، شباهت بین دوکاربر با اطمینان مستقیم بر اساس معیار شباهت مشخص‌شدهٔ هر ژن محاسبه می‌شود و سپس، توسط (رابطه -۶) میزان اطمینان استنتاج‌شده بین دو کاربر به دست می‌آید و با (رابطه -۷) این مقادیر جمع می‌شود و میزان اطمینان استنتاج‌شدهٔ هر دو کاربر را مشخص می‌کند؛ که به‌عنوان شباهت در این حالت در سامانه استفاده می‌شود.

پس‌از محاسبهٔ دقّت برای هر حالت، برای ایجاد تنوّع کافی در پیشنهادهای تولیدشده، از هر روش بهترین حالت -که بالاترین دقّت را دارد- انتخاب می‌شود؛ زیرا روش‌های گوناگون محاسبهٔ شباهت می‌توانند پیشنهادهای متفاوتی تولید کرده و در ایجاد فهرست نهایی پیشنهادها، که از اجتماع نتایج به دست می‌آید، تنوع ایجاد کنند. هرچه این تنوع آیتم‌ها، که از متفاوت بودن فهرست تولیدشده از هر روش به وجود می‌آید، بیشتر باشد، اجتماع نتایج نتیجهٔ بهتری دارد؛ زیرا آیتم‌های تکراری در فهرست‌ها وجود ندراند که در اجتماع آن‌ها مجبور به حذف این آیتم‌ها و استفاده از آیتم‌هایی با کیفیت پایین‌تر که در پایان فهرست‌ها قرار گرفته‌اند، باشیم.

برای انتخاب روش‌های منتسب به هر ژن در الگوریتم ژنتیک با در نظر گرفتن N برابر ۱۰، میزان دقّت پیشنهاد پنج آیتم به هر کاربر در هر حالت برای حدود ۳۰۰ کاربر با ۱۰ امتیاز با استفاده از داده‌های MovieLens 100k اندازه‌گیری شده است و میزان دقّت با فرض این که آیتم‌هایی که امتیاز چهار و پنج دارند و موردعلاقهٔ کاربر هستند، محاسبه شده است، نتایج آن برای روش‌های محاسبهٔ شباهت در (جدول -۲) مشخص شده است.

(جدول-۲) : مقدار دقّت روش‌های مجزا در پیشنهاد پنج آیتم با اعمال الگوریتم مرتب‌سازی پیشنهادها

(Table-2): precision of Top-5 sorted Recommendations generated by separate collaborative filtering methods

روش محاسبهٔ شباهت	کاربر محور	آیتم-محور	کاربرمحور با اطمینان استنتاج‌شده
City Block	0.7016	0.523	0.4458
Euclidean Distance	0.6803	0.5448	0.6
Spearman Correlation	0.3156	-	0.6518
Log	0.6976	0.5415	0.64186

Likelihood		2	
Pearson Correlation	0.1946	0.3348	0.5667
Cosine	0.6823	0.5521	0.5176
Tanimoto	0.7043	0.5408	0.63455

بر اساس تعریف، روش spearman که از رتبه‌بندی امتیاز کاربر به آیتم برای محاسبه شباهت استفاده می‌کند، نمی‌توان آن را برای حالت آیتم‌محور استفاده کرد.

به‌دلیل‌این‌که دقّت هر روش همسایه‌محور به میزان متفاوتی به نسبت بین کاربران و آیتم‌ها در سامانه بستگی دارد، میزان دقّت هر روش با روش‌های دیگر متقاوت است.

در الگوریتم‌های همسایه‌محور در حالتی که تعداد همسایه‌های هرکاربر کم است، ولی این همسایه‌ها قابل-اعتمادند، در برابر حالتی که تعداد همسایه‌های هر کاربر زیاد است، ولی روابط آن‌ها با کاربر هدف غیرقابل اعتماد است، مطلوب‌تر است. پس در حالاتی مانند سامانه‌های بزرگ تجاری که تعداد کاربران خیلی بیشتر از تعداد آیتم‌ها است، روش‌های کاربرمحور می‌توانند نتایجی با دقّت بالاتر داشته باشند، زیرا همسایه‌های آیتم‌ها تعداد کمتری دارند و قابل‌اعتمادتر هستند. به همین شکل در حالتی که در سامانه نسبت تعداد کاربران به تعداد آیتم‌ها کمتر از حد مشخصی است، روش‌های کاربرمحور می-توانند بهتر کار کنند. همان‌طور که در (جدول-۲) مشاهده می‌شود، به‌دلیل‌این‌که در این آزمایش فقط تعدادی حدود ۳۰۰ کاربر بررسی شده است و تعداد آیتم-ها بیشتر از تعداد کاربران است، دقّت روش‌های کاربرمحور بیشتر از روش‌های آیتم محور است. همچنین، مشاهده می‌شود که روش‌های Pearson و Spearman با استفاده از اطّلاعات اطمینان استنتاج‌شده توانستند نتایج به‌مراتب بهتری از حالت معمول تولید کنند. این روش‌های برتر که برای استفاده در روش پیشنهادی انتخاب شده‌اند، در (جدول- ۲) با رنگ تیره مشخص شده است.

۴- آزمایش و ارزیابی

برای پیاده‌سازی این سامانه از فریم‌ورک‌های[۳۱] Apache Mahout برای ایجاد سامانه توصیه‌گر و Jenes[۳۲] برای پیاده‌سازی الگوریتم ژنتیک استفاده شده‌است.

[31] http://mahout.apache.org/
[32] http://jenes.intelligentia.it/

۴-۱- آماده‌سازی داده‌ها

برای آزمودن روش پیشنهادی این مقاله، از داده‌های 100k وMovieLens 1M وHetrec[۳۳] از GroupLens Research[۳۴] استفاده کرده‌ایم که مجموعه‌دادگان‌هایی هستند که به‌طور گسترده در زمینهٔ سامانه‌های توصیه‌گر استفاده می‌شوند. این مجموعه‌دادگان‌ها شامل امتیازدهی کاربران سایت Movielens به فیلم‌ها، که عددی از یک تا پنج است، می‌شود. مشخصات این داده‌ها در (جدول-۳) نشان داده شده است.

(جدول -۳): مشخصات مجموعه‌دادگان‌ها
(Table-3): Datasets Properties

مجموعه‌دادگان	مخفف اسم	تعداد کاربران	تعداد آیتم‌ها	تعداد امتیازدهی‌ها	ویژگی اطّلاعات
Movielens 100k	100K	943	682	100,000	امتیازدهی‌ها از Sep. 1997 تا April. 1998
Movielens 1M	1M	6040	900	1,000,209	کاربرانی که در سال ۲۰۰۰ عضو شده اند.
Hetrec2011 movielens -2k	HR	2113	10197	855,598	کاربرانی که در MovieLens10M دارای اطّلاعات امتیازدهی و tag هستند.

برای صحّت آماری آزمایش، از داده‌های هر مجموعه‌دادگان شش بار حدود ۳۰۰ کاربر به‌صورت تصادفی جدا شده است و از هرکدام به‌طور تقریبی به تعداد ۱۰ امتیازدهی از مجموعه‌دادگان، به‌صورت تصادفی انتخاب شده است و مجموعه داده‌های شماره یک تا شش را برای هر مجموعه‌دادگان ایجاد کرده‌اند. با استفاده از این داده‌ها، پیشنهادهایی برای هرکاربر در هر بخش تولید شده است و درصورتی‌که آیتم پیشنهادشده در مجموعه‌دادگان اصلی توسّط کاربر امتیازی برابر چهار یا پنج داشته باشد، آن را موردپسند کاربر در نظر می‌گیریم. در هر بخش آزمایش‌ها با استفاده از شش سری داده‌ها تکرار شده است و میانگین آن در جدول‌ها قرار داده شده است.

[33] the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011) http://ir.ii.uam.es/hetrec2011 at the 5th ACM Conference on Recommender Systems (RecSys 2011) http://recsys.acm.org/2011
[34] https://grouplens.org/datasets/movielens/

۴-۲- انتخاب شاخص‌های آزمایش

دقّت روش پیشنهادی به میزان سازگاری کروموزوم تولیدشده با الگوریتم ژنتیک بستگی دارد و پیداشدن سازگارترین کروموزوم وابسته به انتخاب مجموعه شاخصهای الگوریتم ژنتیک است؛ همچنین، تولید بهترین نتیجه در الگوریتم فیلترینگ مشارکتی به تعداد همسایه-های انتخاب‌شده برای تولید پیشنهادها برای کاربر بستگی دارد. به این دلایل برای پیدا کردن مناسب‌ترین شاخصها در این بخش، تأثیر تغییر شاخصها بر دقّت روش پیشنهادی بررسی شده است تا بهترین شاخصهای ممکن انتخاب شود.

۴-۲-۱- اندازهٔ جمعیّت

تعداد کروموزوم‌ها در جمعیت، نشان‌دهندهٔ تعداد راه‌حل‌هایی است که در هر نسل برای حل مسئله بررسی می‌شود. تعداد کروموزوم‌ها در جمعیت باید به اندازه‌ای باشد که بتواند پراکندگی فضای جستجو[۳۵] را پوشش دهد. از سوی دیگر هرچه اندازه جمعیت بیشتر باشد، زمان اجرای الگوریتم بیشتر می‌شود. در آزمایش نخست، ارتباط بین اندازهٔ جمعیت و میزان دقّت و زمان اجرای الگوریتم بررسی شده است و اندازهٔ جمعیت از تعداد ۲۰ کروموزوم تا ۲۰۰ کروموزم افزایش داده شده است و تغییرات میزان دقّت نهایی بر روی داده‌های تست از مجموعه‌دادگان MovieLens 100k بررسی شده است. تعداد نسل‌ها در این آزمایش برابر ۱۰۰ قرار داده شده، و نتایج آن در (جدول-۴) ثبت شده است.

(جدول -۴) : تأثیر افزایش تعداد جمعیت بر میزان دقّت و زمان اجرا

(Table-4): Impact of increasing population size on the purposed method's precision and running duration

تعداد جمعیت	میزان دقّت	زمان(میلی‌ثانیه)
20	0.6243	26721
40	0.6246	59359
60	0.6263	87508
80	0.6263	117592
100	0.6297	157722
120	0.6301	183116
140	0.6309	209213
160	0.6299	227409
180	0.6293	250578
200	0.6306	306855

مشاهده می‌شود که با افزایش تعداد کروموزوم‌ها در جمعیّت، زمان اجرا افزایش یافته و میزان دقّت بهبود می‌یابد؛ امّا افزایش جمعیت بیشتر از تعداد ۱۰۰ کروموزوم تأثیر به‌سزایی در افزایش دقّت ندارد؛ به همین دلیل تعداد بهینهٔ جمعیّت برابر ۱۰۰ کروموزم است.

۴-۲-۲- شرط توقّف

برای توقف الگوریتم ژنتیک از شرط رسیدن به تعداد بیشینهٔ نسل‌ها استفاده شده است. برای یافتن بهترین تعداد نسل‌ها برای توقف الگوریتم ژنتیک در (جدول-۵) ارتباط بین تعداد نسل‌ها و میزان سازگاری نهایی الگوریتم ژنتیک و زمان اجرای الگوریتم، با آزمایش بر روی داده‌ها با شرایط آزمایش پیشین و ثابت نگه‌داشتن تعداد جمعیت برابر ۱۰۰ کروموزوم، بررسی شده است.

مشاهده می‌شود که با افزایش تعداد نسل‌ها برای توقف الگوریتم ژنتیک، زمان اجرا افزایش یافته و میزان دقّت بهبود می‌یابد؛ امّا افزایش تعداد نسل‌ها بیشتر از تعداد ۸۰ نسل تأثیر به‌سزایی در افزایش دقّت ندارد؛ به همین دلیل تعداد بهینهٔ نسل‌ها برای توقف الگوریتم ژنتیک برابر ۸۰ نسل است.

(جدول-۵) : تأثیر افزایش تعداد نسل بر میزان دقّت و زمان اجرا

(Table-5): Impact of increasing number of the generations on the purposed method's precision and running duration

تعداد نسل	میزان دقّت	زمان(میلی ثانیه)
20	0.6286	143077
40	0.6269	143738
60	0.6266	142435
80	0.6307	148084
100	0.6309	146056
120	0.6303	149128
140	0.6306	149887
160	0.6309	153994
180	0.6299	151994
200	0.628	154270

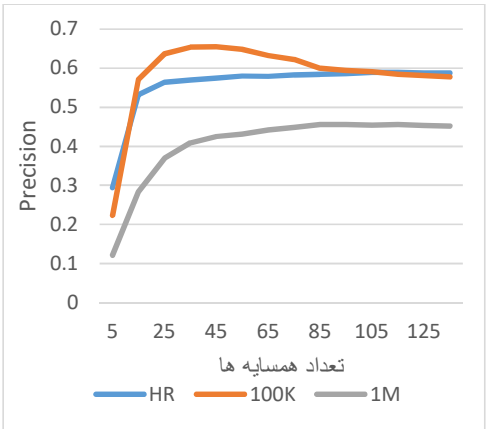
[35] Search space

۴-۲-۳- تعداد همسایه‌ها

در الگوریتم فیلترینگ مشارکتی همسایه‌محور، پیش‌بینی امتیازدهی کاربر به آیتم‌هایی که تاکنون امتیازدهی نکرده است، با الگوریتم" k نزدیک‌ترین همسایه"[1] انجام می‌شود شود که در آن به تعداد k همسایه برای هر کاربر - که اندازهٔ شباهت آن‌ها با کاربر بیشتر از دیگر کاربران است- برای فرایند پیش‌بینی امتیازدهی‌های کاربر انتخاب می‌شوند.

برای یافتن بهترین تعداد همسایه‌ها برای هر کاربر، با افزایش تعداد همسایه‌های ۳۰۰ کاربر به‌صورت تصادفی از مجموعه‌دادگان‌های HR100Kو MovieLens 1M، میزان دقّت الگوریتم در ۸۰ نسل و جمعیت ۱۰۰ کروموزوم، بررسی شده است و نتیجهٔ آن در (نمودار-۱) به نمایش در آمده است.

مشاهده می‌شود که با تعداد پنج همسایه میزان دقّت پایین است و با افزایش همسایه‌هایی که در تولید پیشنهادها مشارکت دارند، میزان دقّت افزایش می‌یابد و در نهایت زمانی‌که تعداد همسایه‌ها خیلی زیاد می‌شود، دقّت پایین می‌آید؛ که این پدیده به‌دلیل این است که تأثیر ارتباط‌های قوی اندکی که بین کاربر و همسایه‌های نزدیک او (مشابه او) وجود داشته است، از سوی تعداد زیادی ارتباط ضعیف با همسایه‌های دورتر از او، از بین رفته است. به‌دلیل تفاوت نسبت تعداد کل کاربران و آیتم‌ها در مجموعه‌دادگان‌ها، مقدار بهینهٔ همسایه‌ها برای آن‌ها متفاوت است. باتوجّه‌به نتایج به دست آمده که در (نمودار-۱) رسم شده است، این مقدار برای مجموعه‌دادگان 100K برابر ۴۴ و برای مجموعه‌دادگان HR برابر ۱۰۵ و برای مجموعه‌دادگان 1M برابر 80 است.



(نمودار-۱) : تأثیر افزایش تعداد همسایه‌ها بر دقّت الگوریتم
(Chart-1): Impact of increasing number of neighbors on the precision of the purposed method

[1] k-nearest-neighbors

۴-۳- نتایج آزمایش

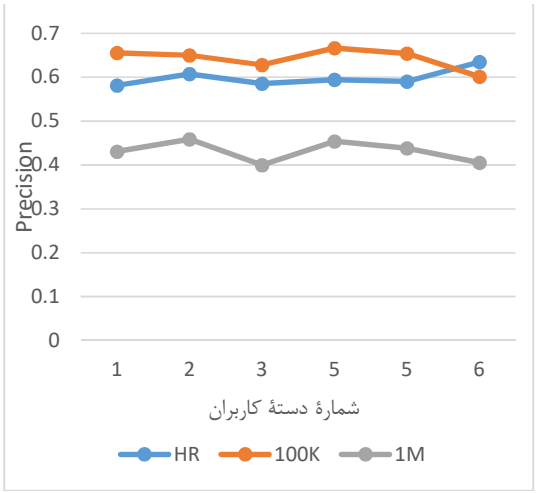
برای پیاده‌سازی الگوریتم ژنتیک، در مقاله [4] میزان Crossover برابر ۰,۸ است و این فرایند توسط one point Cross Over انجام شده و میزان Mutation برابر ۰,۰۲ است، از روش تغییر یک بیت استفاده شده و انتخاب والدها با استفاده از تورنمنت است. در آزمایش روش پیشنهادی نیز همین مقادیر در نظر گرفته شده است. همچنین، در مقالهٔ [4] به تعداد ۱۰ توصیه پیشنهاد شده است که ما برای مقایسه درست با این مقاله، همین تعداد پیشنهاد برای ارزیابی الگوریتم انتخاب کرده‌ایم.

بر اساس نتایج (بخش ۲-۴) اندازه جمعیت ۱۰۰ کروموزوم است و الگوریتم تا رسیدن به ۸۰ نسل تکرار شده‌است. همچنین، یک نخبه از هر نسل بدون تغییر وارد نسل بعدی می‌شود. با استفاده از هفت روش محاسبهٔ شباهت معرفی‌شده در (بخش۱-۱-۲) در الگوریتم ژنتیک، **Error! Reference source not found.** کروموزم‌ها با هفت ژن ایجاد می‌شوند و به این صورت، سامانهٔ پیشنهادی پیاده‌سازی شده است.

در ادامه میزان زمان اجرا، precision با (رابطه -۴) و recall با (رابطه-۵) برای هر روش به‌دست آمده و مقایسه‌ها بر اساس میانگین شش بار تکرار اجرای روش پیشنهادی و روش‌های مشابه برای مجموعه داده‌های ML1M و ML100K و HR نشان داده شده است.

۴-۳-۱- بررسی یکنواخت بودن نتایج روش پیشنهادی بر روی دسته‌های مختلف مجموعه‌دادگان‌ها

در (نمودار-۲) هر آزمایش با شش دسته کاربر از هر مجموعه‌دادگان تکرار و میزان دقّت هر اجرا برای روش پیشنهادی برای هر دسته از کاربران از مجموعه‌دادگان‌های 1M و100k و HR نشان داده شده - است.



(نمودار-۲) : میزان دقّت روش پیشنهادی با

دسته‌های کاربران مجموعه‌دادگان HR و100k و 1M

(Chart-2): precision changes of the purposed method on different user sets of MovieLens 100k, 1M and HR dataset

بر اساس این نمودار مشاهده می‌شود که مقدار دقّت روش پیشنهادی در مجموعه‌داده‌های 100k در بازه۰/۶ و ۰/۷ است و دقّت روش پیشنهادی در مجموعه-داده‌های 1M در بازه ۰/۴ و۰/۵ است و و دقّت روش پیشنهادی در مجموعه‌داده‌های HR در حدود ۰/۶ است دقّت تغیر قابل ملاحظه‌ای در دسته‌های مختلف کاربران یک مجموعه داده، ندارد.

۴-۳-۲- بررسی اثر هر مرحله از روش پیشنهادی

برای مشاهدهٔ تأثیر هریک از بخش‌های پیشنهادی الگوریتم، در آغاز، با پیاده‌سازی روش پیشنهادی [4] سامانهٔ توصیه‌گر Invenire که با الگوریتم‌های یک تا پنج از (جدول-۶) کار می‌کند، میزان دقّت به دست آمده است.

در مرحلهٔ بعد تابع ارزیابی این سامانه تغییر کرده و بر اساس روش پیشنهادی پیاده‌سازی شده است. در آزمایش بعدی اطمینان مطابق نتایج (بخش۴-۳) به روش‌های یک و پنج از (جدول-۶) اضافه شده است و مقدار دقّت سامانه به دست آمده است. در مرحلهٔ بعد فهرست پیشنهادها هریک از روش‌های پایه در الگوریتم، مرتب‌سازی و دقّت سامانه محاسبه شده است. در مرحلهٔ نهایی با اضافه کردن دو روش شش و هفت از جدول به الگوریتم‌های توصیه، دقّت روش پیشنهادی به دست آمده است. مراحل ذکرشده در این آزمایش‌ها در (جدول -۷) نشان داده شده است.

(جدول- ۶): روش‌های محاسبه شباهت در روش پیشنهادی

(Table-6): Similarity Calculating methods in proposed method

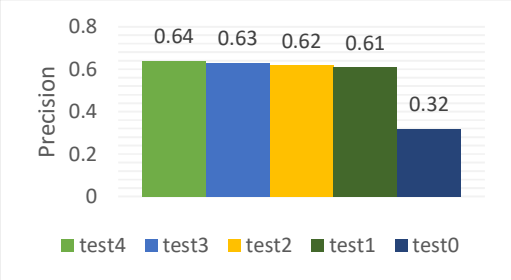
شماره	روش محاسبهٔ شباهت
۱	Spearman
۲	Euclidean
۳	Tanimoto
۴	LogLikelihood
۵	Pearson
۶	Cosine
۷	CityBlock

(جدول -۷) : تغییرات هر مرحله از روش پیشنهادی

(Table-7): changes in each level of the proposed method

آزمایش	توضیح
test0	Invenire
test1	test0 با تغییر تابع ارزیابی
test2	test1 با محاسبه اطمینان
test3	test2 با مرتب سازی فهرست پیشنهادها
test4	test3 با اضافه شدن دو الگوریتم توصیه (روش پیشنهادی)

میزان دقّت حاصل از هر مرحله که نشان‌دهندهٔ میزان تأثیر هر یک از مراحل در نتیجهٔ سامانه پیشنهادی است، در (نمودار-۳) مشاهده می‌شود.



(نمودار-۳) : تغییرات precision روش پیشنهادی در هر مرحله آزمایش

(chart-3): precision changes of proposed method at each step

بر اساس این نمودار دقّت هر مرحله بیشتر از مرحلهٔ پیشین به دست آمده است.

۴-۳-۳- مقایسه روش پیشنهادی و روش‌های سازنده آن

برای بررسی تأثیر روش پیشنهادی در اجتماع نتایج الگوریتم‌های توصیه، روش‌های فیلترینگ مشارکتی مجزا که بهترین نتیجه را داشته‌اند آزمایش شده‌اند و میزان precision و recall آن‌ها محاسبه شده‌است.

در (جدول- ۸) مقدار precision روش پیشنهادی در مقایسه با روش‌های مجزا که سازنده آن هستند برای مجموعه‌دادگان‌های 1M و 100K نشان داده شده‌است.

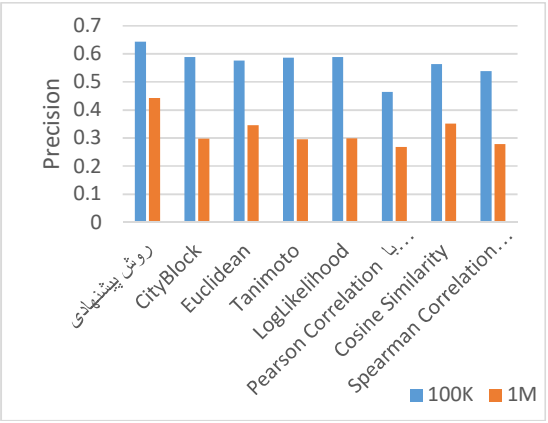
(جدول -۸) : میانگین دقّت روش پیشنهادی و روش‌های مجزای سازندهٔ آن

(Table-8): Average precision of purposed method and it's building methods

	روش	100K	1M
۱	روش پیشنهادی	0.642733	0.4425

۲	CityBlock	0.588013	0.29795
۳	Euclidean	0.576383	0.346217
۴	Tanimoto	0.586333	0.2962
۵	LogLikelihood	0.588333	0.298383
۶	Pearson Correlation با اطمینان	0.463633	0.267517
۷	Cosine Similarity	0.563233	0.351667
۸	Spearman Correlation با اطمینان	0.538267	0.2785

نتایج این جدول در (نمودار-۴) به نمایش در آمده است. بر اساس این نمودار میزان دقّت روش پیشنهادی بهتر از روش‌های سازندهٔ آن است.



(نمودار-۴) : مقدار دقّت روش پیشنهادی و روش‌های سازندهٔ آن

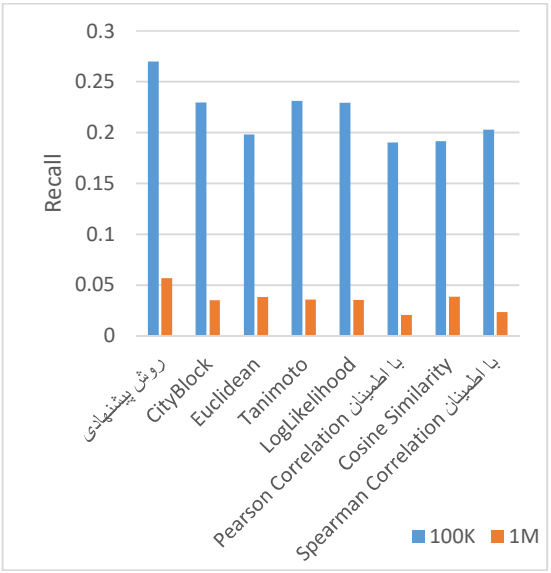
(chart-4): Precision of purposed method and it's building method

میزان recall روش پیشنهادی و روش‌های مجزّای سازندهٔ آن برای مجموعه‌دادگان‌های 1M و 100Kدر (جدول -۹) نشان داده شده است و (نمودار-۵) این نتایج را به نمایش در آورده است.

(جدول -۹) : میزان recall روش پیشنهادی و روش‌های مجزای سازندهٔ آن

(Table-9): Average recall of purposed method and it's building methods

	روش	100K	1M
۱	روش پیشنهادی	0.2698	0.0566
۲	CityBlock	0.2297	0.0351
۳	Euclidean	0.1984	0.038
۴	Tanimoto	0.2314	0.0356
۵	LogLikelihood	0.2293	0.0355
۶	Pearson Correlation با استفاده از اطمینان	0.1904	0.0207
۷	Cosine Similarity	0.1917	0.0385
۸	Spearman Correlation با استفاده از اطمینان	0.2031	0.0232



(نمودار-۵) : مقدار recall روش پیشنهادی و روش‌های سازندهٔ آن

(chart-5): Recall of purposed method and it's building method

بر اساس این نتایج میزان precision و recall روش پیشنهادی بهتر از روش‌های مجزّای تشکیل‌دهندهٔ آن است و اجتماع نتایج روش‌ها به نتیجه‌ای به‌مراتب بهتر از هریک از روش‌ها به‌صورت مجزا دست یافته است.

۴-۳-۴- مقایسه روش پیشنهادی با روش‌های مشابه

برای مقایسهٔ روش پیشنهادی با روش‌های مشابه آن، با استفاده از [4] سامانه Invenire برای تولید پیشنهادها طبق مشخصات ارائه‌شده پیاده‌سازی شده است و بر اساس [1] روش فیلترینگ مشارکتی الگومحور SVDPlusPlus و بر اساس [24] روش فیلترینگ مشارکتی الگو محور Parallel SGD برای یک سامانهٔ توصیه‌گر N آیتم برتر پیاده‌سازی شده‌اند و میزان زمان اجرا، precision و recall آن‌ها با روش پیشنهادی مقایسه شده است.

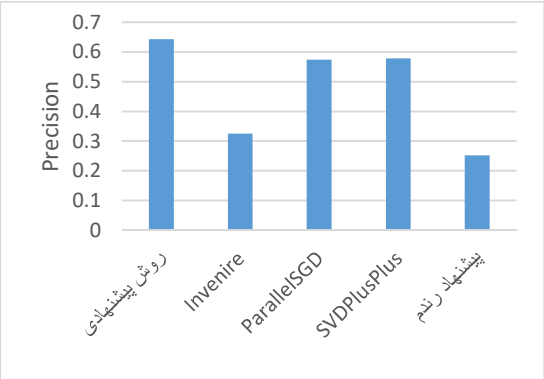
میزان زمان اجرا پس‌از تعیین ترکیب مناسب، precision و recall روش پیشنهادی و روش‌های مشابه برای مجموعه داده‌های ML1M و ML100K و HR در (جدول-۱۰) و (جدول-۱۱) و (جدول-۱۲) نشان داده شده است.

(جدول-۱۰): مقایسهٔ روش پیشنهادی و روش‌های مشابه(ML100K)

(Table-10): comparing purposed method and similar methods

	precision	recall	Duration(ms)
روش پیشنهادی	6427.0	2698.0	422
Invenire	3248.0	2314.0	328
ParallelSGD	5734.0	1605.0	8163
SVDPlusPlus	5780.0	2197.0	25719
پیشنهاد رندم	2514.0	0459.0	106

برای مقایسهٔ بهتر، میزان Precision نتایج این جدول در (نمودار-۶) نشان داده شده است.



(نمودار-۶): مقایسهٔ Precision روش پیشنهادی و روش‌های مشابه (ML100K)

(chart-6): Precision of purposed method and similar method(ML100K)

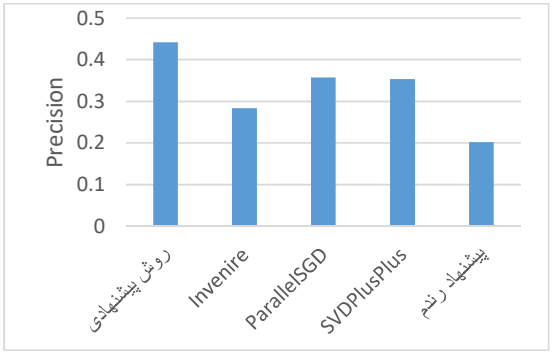
روش پیشنهادی و روش‌های مشابه آن با داده‌های ML1M در (جدول-۱۱) مقایسه شده‌است.

(جدول-۱۱): مقایسه ی روش پیشنهادی و روش‌های مشابه(ML1M)

(Table-11): comparing purposed method and (ML1M) similar method

	precision	recall	Duration(ms)
روش پیشنهادی	4425.0	0566.0	327
Invenire	2837.0	0355.0	443
ParallelSGD	3578.0	0452.0	7942
SVDPlusPlus	3534.0	0439.0	8059
پیشنهاد رندم	2023.0	0268.0	105

برای مقایسهٔ بهتر، میزان Precision نتایج این جدول در (نمودار-۷) نشان داده‌شده است.



(نمودار-۷): مقایسهٔ Precision روش پیشنهادی و روش‌های مشابه (ML1M)

(chart-7): Precision of purposed method and similar method(ML1M)

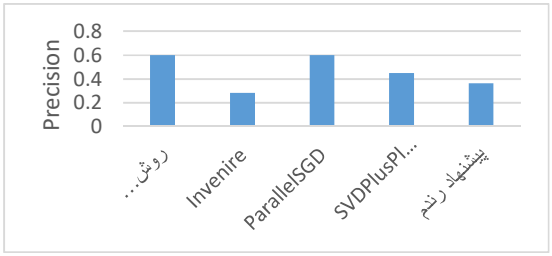
روش پیشنهادی و روش‌های مشابه آن با داده‌های HR در (جدول -۱۲) مقایسه شده است.

(جدول-۱۲): مقایسهٔ روش پیشنهادی و روش‌های مشابه (HR)

(Table-12): comparing purposed method and (HR) similar method

	precision	recall	Duration(ms)
روش پیشنهادی	0.5995	0.0357	373
Invenire	0.2839	0.0145	280
ParallelSGD	0.5993	0.0342	7665
SVDPlusPlus	0.4502	0.0521	24772
پیشنهاد رندم	0.3645	0.0190	122

برای مقایسهٔ بهتر، میزان Precision نتایج این جدول در (نمودار-۸) نشان داده‌شده است.



(نمودار-۸): مقایسهٔ Precision روش پیشنهادی و روش‌های مشابه(HR)

(Chart-8): Precision of purposed method and similar method (HR)

مشاهده می‌شود که روش پیشنهادی پس‌از تعیین ترکیب مناسب هر مجموعه داده، به نسبت روش‌های الگو محور زمان کمتری برای تولید پیشنهادها نیاز دارد. بر اساس نتایج به‌دست آمده، مشاهده می‌شود که روش پیشنهادی Precision بالاتری نسبت به روش‌های مشابه آن دارد.

علت پایین بودن کلی مقدار recall که بر اساس (رابطه-۵) به دست می‌آید، این است که در این آزمایش فقط ۱۰ آیتم به کاربر پیشنهاد شده، ولی تعداد آیتم‌ها در مجموعه‌دادگان‌ها زیاد است و در نتیجه، تعداد آیتم-های مطلوب کاربر بسیار زیادتر از تعداد آیتم‌هایی است که در بین فهرست پیشنهادها ممکن است موردپسند او باشد.

۵- نتیجه‌گیری

در این مقاله، نخست سامانه‌های توصیه‌گر فیلترینگ مشارکتی و کلیات روش‌های آن بررسی شد و سپس، روش استفاده از اطمینان که در این مقاله برای غلبه بر چالش شروع سرد استفاده شده است، شرح داده شد. در ادامه، اجتماع نتایج در سامانه‌های توصیه‌گر به‌عنوان روشی برای رسیدن به فهرست پیشنهادها با دقّتی مناسب بررسی و سپس، استفاده از الگوریتم ژنتیک در سامانه‌های توصیه‌گر مطالعه شد. در بخش بعد، یک روش پیشنهادی تشریح شد که در آن فرایند اجتماع نتایج توسط الگوریتم ژنتیک مدیریت می‌شود و در ادامه برای انتخاب روش ارزیابی کروموزم‌ها در الگوریتم ژنتیک، معیارهای ارزیابی سامانه‌های توصیه‌گر بررسی و در نهایت، روش پیشنهادی برای تولید و مرتب کردن پیشنهادها و انتخاب روش توصیه متناظر با هر ژن در کروموزوم‌های الگوریتم ژنتیک بیان شده است.

روش پشنهادی در آزمایش با داده‌های ML100k و ML1M میزان precision و recall بالاتری به نسبت الگوریتم‌های توصیه تشکیل‌دهندهٔ آن به دست آورد. این روش در آزمایش با داده‌های ML100k به precision با مقدار ۶۴ درصد رسید که از دقّت بهترین روش تکی ۵۴٪ بیشتر است و در آزمایش با داده‌های ML1M به precision با مقدار ۰/۴۳ درصد رسید که از بهترین روش تکی هشت درصد بیشتر است. و همچنین، در آزمایش با داده‌های ML100k و ML1M و HR میزان precision بالاتری به نسبت روش‌های مشابه خود به‌دست آورد.

روش پیشنهادی این پژوهش با اجتماع نتایج روش-های حافظه‌محور، می‌تواند ترکیبی ارائه دهد که در مقایسه با سامانه‌های مشابه و سامانه‌های فیلترینگ مشارکتی سازنده‌اش به‌صورت مجزا عملکرد بهتری دارد. این سامانه برای ایجاد تنوع در روش‌های محاسبهٔ شباهت در الگوریتم فیلترینگ مشارکتی از حالت‌های کاربرمحور و آیتم‌محور و روش‌های گوناگون محاسبهٔ شباهت، و برای حل مشکل پراکندگی داده‌ها از اطمینان کاربران به یکدیگر استفاده می‌کند. برای تولید پیشنهادها برای کاربر، با استفاده از الگوریتم ژنتیک بهترین ترکیب الگوریتم‌های توصیه را انتخاب و از فهرست‌های پیشنهادهای تولیدشده توسط این روش‌ها که بر اساس امتیاز و سپس، معروف بودن مرتب شده‌اند، مناسب‌ترین آیتم‌ها را به کاربر پیشنهاد می‌کند.

علت برتری این روش به الگوریتم‌های توصیه تشکیل‌دهندهٔ آن، این است که با مرتّب‌سازی آیتم‌ها در روش پیشنهادی آیتم‌هایی که با احتمال بیشتری موردعلاقهٔ کاربر هستند، در فهرست پیشنهادها بالاتر قرار می‌گیرند، با این کار هر الگوریتم توصیه بهترین آیتم‌های خود را در آغاز فهرست پیشنهادهایش قرار می‌دهد و با استفاده از روش پیشنهادی این آیتم‌های برتر از بالای فهرست هر روش انتخاب می‌شود و مواردی که اطمینان زیادی به آن‌ها وجود ندارد، به کاربر پیشنهاد نخواهد شد. به این دلیل دقّت روش پیشنهادی بهتر از دقّت تک‌تک روش‌های سازندهٔ آن است.

علت برتری این روش در تولید پیشنهادها نسبت به سامانهٔ Invenire [4] نوآوری‌ها و بهبودهایی است که در آن اعمال شده و که شامل موارد زیر است:

- محاسبهٔ تابع ارزیابی الگوریتم ژنتیک بر اساس دقّت فهرست پیشنهادها در الگوریتم ژنتیک برای افزایش دقّت سامانهٔ پیشنهادی
- افزایش تنوع روش‌های پایه برای تولید پیشنهادها با استفاده از روش‌های محاسبه شباهت متفاوت برای بهبود اجتماع نتایج
- رفع مشکل برخی روش‌های محاسبهٔ شباهت با پراکندگی داده‌ها توسط استفاده از اطمینان بین کاربران
- مرتب‌سازی پیشنهادهای روش‌های پایه‌ای برای افزایش دقّت

هدف از این پژوهش، ایجاد یک سامانهٔ توصیه‌گر برای استفاده در سامانه‌هایی است که نیاز به اجرای سریع

مجدد الگوریتم پس‌از تغییر اساسی داده‌ها دارند؛ به همین دلیل سامانه‌های الگومحور که نیاز به زمانی مجزا برای آموزش دارند، به‌عنوان روش‌های پایه در نظر گرفته نشده‌اند. می‌توان در ادامهٔ این پژوهش برای سامانه‌هایی که نیاز به پاسخ‌دهی آنی ندارند، مسئلهٔ بهبود کارایی سامانه با اجتماع نتایج روش‌های الگو محور را نیز بررسی کرد. این سامانه‌ها به‌دلیل محدود نبودن در زمان محاسبات، به‌طور معمول نتایج دقیق‌تری دارند و بر اساس قاعده، اجتماع نتایج آن‌ها باید بتواند پیشنهادهای بهتری برای کاربران تولید کند.

همچنین، می‌توان با تلاش برای بهبود هریک از توابع محاسبهٔ شباهت در روش پیشنهادی مقدار دقّت نهایی را افزایش داد. همان‌طورکه توانستیم با افزودن اطمینان استنتاج‌شده به روش‌های محاسبهٔ شباهت Spearman و Pearson به نتایج بهتری دست بیابیم، ممکن است بتوان راه‌هایی برای رفع نقاط ضعف سایر روش‌های محاسبهٔ شباهت نیز پیدا کرد و نتیجه روش پیشنهادی را بهبود داد.

در روش پیشنهادی برای اضافه‌کردن اطمینان استنتاج‌شده تنها یک روش مناسب مسئله از میان روش-های مختلف انتخاب و پیاده‌سازی شد. توابع مختلف محاسبهٔ اطمینان با یکدیگر تفاوت‌های فراوانی دارند و ممکن است با استفاده از توابع اطمینان دیگر بتوان به نتایج بهتری رسید.

سامانهٔ ارائه‌شده فقط با تعداد ثابتی آیتم پیشنهادی کار می‌کند. در ادامهٔ این پژوهش می‌توان با کاهش و افزایش این تعداد و با بررسی تأثیر تغییر تعداد آیتم‌های پیشنهادشده از طرف سامانه به هر کاربر، به نتایج جدیدی دست یافت.

همچنین، با بررسی معیارهای ارزیابی دیگر علاوه‌بر دقّت سامانه، مانند نوآوری در پیشنهادها، توانایی تطبیق با تغییرات، مقیاس‌پذیری، تنوع و جدید بودن پیشنهادها به‌عنوان هدف سامانه، می‌توان تابع ارزیابی الگوریتم ژنتیک را تغییر داد و تلاش کرد تا سامانهٔ پیشنهادی به این اهداف نیز برسد.

۶- مراجع 6- Refrence

[1] Y. Koren, "Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Mode,," Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008, 2008.

[2] F. Ricci, L. Rokach and B. Shapira, "Recommender Systems Handbook," Springer, 2011.

[3] F. Isinkaye, Y. Folajimi and B. Ojokoh, "Recommendation systems: Principles, methods and evaluation," Egyptian Informatics Journal 16, 261–273 ,2015.

[4] E. Q. Silva, C. G.Camilo-Junior, L. MarioL.Pascoal and C. Thierson, "An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering," IEEE Congress on Evolutionary Computation (CEC), Beijing, China, 2014, pp. 959-966, 2014.

[5] B. B. Sinha and R. Dhanalakshmi, "Evolution of recommender system over the time," Soft Comput 23, 12169–12188, 2019.

[6] S. Gupta and S. Nagpal, "Trust Aware Recommender Systems: A Survey on Implicit Trust Generation Techniques," (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (4) , 3594-3599, 2015.

[7] M. Papagelis, D. Plexousakis and T. K. Kutsuras, "Alleviating the Sparsity Problem of Collaborative Filtering Using Trust Inferences," iTrust 2005, LNCS 3477, pp. 224 – 239, 2005..

[8] J. Bobadilla, F. Ortega, A. Hernando and A. Gutiérrez, "Recommender systems survey," Knowledge-Based Systems 46 109–132,2013.

[9] T. Dunning, "Accurate methods for the statistics of surprise and coincidence," Computational Linguistics Volume 19, Number 1 pages 61-74,1993.

[10] T. K, Paradarami, NathanielD, Bastian, J. and Wightman, "A hybrid recommender system using artificial neural networks," Expert Systems With Applications 83 (2017).

[11] Z. Kang, C. Peng and Q. Cheng, "Top-N Recommender System via Matrix Completion," Association for the Advancement of Artificial 2016.

[12] U. Kużelewska, "Clustering Algorithms in Hybrid Recommender System on MovieLens Data," UDIES IN LOGIC, GRAMMAR AND RHETORIC 37 (50) 2014.

[۱۳] حسینی منیره، نصرالهی مقصود، بقائی علی. یک سامانه توصیه‌گر ترکیبی با استفاده از اعتماد و خوشه‌بندی دوجهته به‌منظور افزایش کارایی پالایش‌گروهی. پردازش علائم و داده‌ها. ۱۳۹۷; ۱۵ (۲) :۱۳۲-۱۱۹

[13] M. Hosseini, M. Nasrollahi and A. Baghaei, "A hybrid recommender system using trust and bi-clustering in order to increase the efficiency of

collaborative filtering,". JSDP 2018; 15 (2) :119-132.
[14] P. Massa and B. Bhattacharjee, "Using Trust in Recommender Systems:An Experimental Analysis," iTrust,Springer-Verlag Berlin Heidelberg , LNCS 2995, pp. 221–235,2004.
[15] W. Yuan, L. Shu, H. Chao, D. Guan, Y. Lee and S. Lee, "itars: trustaware recommender system using implicit trust networks,," Communications,IET, 4(14):17091721, 2010.
[16] SAMUEL, O. E. L, D. VICTOR, L. ANISIO, M. LUIZ and P. GISELE L, "Is Rank Aggregation Effective in Recommender Systems? An Experimental Analysis," ACM Transactions on Intelligent Systems and Technology 11(2), 2019.
[17] M. T. Ribeiro, Nivio Ziviani,, Edleno Silva De Moura and , Itamar Hata,, "Multiobjective Pareto-Efficient Approaches for Recommender Systems," ACM Trans. Intell. Syst. Technol. 5, 4, Article 53 , 2014.
[18] S. Oliveira, V. Diniz, A. Lacerda and G. L. Pappa., "Evolutionary rank aggregation for recommender systems.," IEEE Congress on Evolutionary Computation (CEC). 255–262, 2016.
[19] s. Mirjalili, "Evolutionary Algorithms and Neural Networks," Studies in Computational Intelligence 780,Springer International Publishing AG, part of Springer Nature, 2019.
[20] B. Alhijawi and Y. Kilani, "A collaborative filtering recommender system using genetic algorithm," Information Processing & Management 57(6):102310, 2020.
[21] M. Bhusal and A. Shakya, "Collaborative Filtering Recommender System Using Genetic Algorithm," Proceedings of IOE Graduate Conference, Volume: 6, 2019.
[22] J. Xiao, M. Luo, J.-M. Chen and J.-J. Li, "An Item Based Collaborative Filtering System Combined with Genetic Algorithms Using Rating Behavior," Springer International Publishing Switzerland , ICIC 2015, Part III, LNAI 9227, pp. 453–460,, 2015.
[23] F. H. d. Olmo and E. Gaudioso, "Evaluation of recommender systems: A new approach," Expert Systems with Applications 35 790–804 ,2008.
[24] G. Takacs, I. Pilaszy, B. Nemeth and D. Tikk, "Scalable Collaborative Filtering Approaches for Large Recommender Systems," Journal of Machine Learning Research 10 623-656,, 2009.
[25] S. Oliveira, V. Diniz, A. Lacerda, L. Merschmanm and G. L. Pappa., "Is Rank Aggregation Effective in Recommender Systems? An Experimental Analysis," ACM Trans. Intell.Syst. Technol. 1, 1, Article 1 ,2019.
[26] E. Q. d. Silva, C. G. Camilo, L. M. L. Pascoal and T. C. Rosa, "An evolutionary approach for combining results of recommender systems techniques based on Collaborative Filtering," Expert Systems With Applications 53 204–218,2016.

مژده رباطی انارکی مدرک کارشناسی خود را در رشته مهندسی‌کامپیوتر گرایش نرم‌افزار از دانشگاه علم و صنعت در سال ۱۳۹۷ و مدرک کارشناسی ارشد خود را در رشتهٔ هوش مصنوعی در سال ۱۳۹۹ از دانشگاه الزهرا دریافت کرد. بازی‌های رایانه‌ای، سامانه‌های توصیه‌گر و رایانش تکاملی زمینه‌های پژوهشی مورد علاقه ایشان است.
نشانی رایانامهٔ ایشان عبارت است از:
Robati.m7@gmail.com

نوشین ریاحی دانش‌آموختهٔ دانشگاه صنعتی شریف در رشتهٔ مهندسی برق است. ایشان در حال حاضر عضو هیأت علمی و دانشیار گروه مهندسی کامپیوتر دانشکده فنی مهندسی دانشگاه الزهرا است. بازشناسی و پردازش گفتار، پردازش هوشمند متن و پردازش سیگنال‌های پزشکی از زمینه‌های پژوهشی مورد علاقه ایشان است.
نشانی رایانامهٔ ایشان عبارت است از:
nriahi@alzahra.ac.ir