پینو: یک سامانه توصیه‌گر با استفاده از کاوش در سیاهه‌های وب

محمدرضا عباس‌نژاد[۱]، دانشجوی کارشناسی ارشد؛ امیر جهانگرد رفسنجانی[۲]، استادیار؛ محمدرضا پژوهان[۳]، استادیار

۱- گروه مهندسی کامپیوتر - دانشگاه یزد - یزد - ایران - abbasnezhad.m.r@stu.yazd.ac.ir
۲- گروه مهندسی کامپیوتر - دانشگاه یزد - یزد - ایران - jahangard@yazd.ac.ir
۳- گروه مهندسی کامپیوتر - دانشگاه یزد - یزد - ایران - pajoohan@yazd.ac.ir

چکیده: سامانه‌های توصیه‌گر در وبگاه‌ها شخصی‌سازی وب را به‌صورت هوشمند و برخط با ارائه پیشنهادهایی به کاربران انجام می‌دهند. این سامانه‌ها را می‌توان با مدل‌سازی شیوه‌های دسترسی کاربران با استفاده از کاوش در سیاهه‌های وب ایجاد نمود. به بیانی دیگر می‌توان الگوهای دسترسی کاربران را با روش‌های داده‌کاوی از سیاهه‌های وب استخراج کرد؛ سپس به کاربران بر پایه این الگوها پیشنهاد داد. سامانه‌های توصیه‌گر گوناگونی مبتنی بر کاوش در سیاهه‌های وب ایجاد شده‌اند اما هنوز بهبود کارایی و پیچیدگی آن‌ها موضوعی چالش‌برانگیز است. در این مقاله، سامانه توصیه‌گری بنام پینو مبتنی بر کاوش در سیاهه‌های وب به همراه رویکرد جدیدی برای استخراج الگوهای دسترسی در آن ارائه شده است. در این رویکرد ابتدا شیوه‌های دسترسی کاربران با گرافی جهت‌دار و وزن‌دار مدل می‌شوند. صفحات، رأس‌های این گراف و یال‌ها نشان‌دهنده ارتباط بین آن‌ها بر اساس تکرارهای با هم صفحات هستند. وزن یال‌ها بر اساس معکوس احتمال شرطی مشاهده صفحات با در نظر گرفتن ترتیب آن‌ها محاسبه می‌شود سپس صفحات با بخش‌بندی گراف شیوه‌های دسترسی کاربران بر پایه کوتاه‌ترین مسیرها خوشه‌بندی می‌شوند. پیشنهاددهی بر پایه این الگوها با پیچیدگی زمانی ثابت و سازگار با فراموش‌کاری پروتکل HTTP انجام می‌گیرد. سامانه پینو بر روی سیاهه‌های یک سرور مورد ارزیابی قرار گرفته است. اثربخشی پیشنهاددهی با معیارهای قابلیت پیشنهاددهی، پیشنهادهای درست، دقت و پوشش ارزیابی شده است. نتایج ارزیابی نشان‌دهنده توانایی سامانه پینو در بهبود کیفیت پیشنهادها است به‌گونه‌ای که میانگین همساز بین این معیارها در سامانه پینو به ۵۷٪ رسیده که نسبت به سامانه‌های پیشین ۱۲٪ بهبود یافته است.

واژه‌های کلیدی: سامانه توصیه‌گر، کاوش در سیاهه‌های وب، استخراج الگوهای دسترسی، کوتاه‌ترین مسیرها در گراف.

Pino: A Recommender System using Web Usage Mining

M. R. Abbasnezhad[1], MSc Student; A. Jahangard Rafsanjani[2], Assistant Professor; M. R. Pajoohan[3], Assistant Professor

1- Department of Computer Engineering, Yazd University, Yazd, Iran, Email: abbasnezhad.m.r@stu.yazd.ac.ir
2- Department of Computer Engineering, Yazd University, Yazd, Iran, Email: jahangard@yazd.ac.ir
3- Department of Computer Engineering, Yazd University, Yazd, Iran, Email: pajoohan@yazd.ac.ir

Abstract: Recommender systems for websites do web personalization online and intelligently using recommendations to the users. These systems can be developed by web usage mining techniques which model user navigation patterns. In other words, data mining methods can be used to discover user access patterns from web logs. Then, recommendations to the users are provided based on these patterns. A variety of recommender systems based on web usage mining have been proposed, although improving the efficiency and complexity of them is still a challenging issue. In this paper, a recommender system called Pino has been proposed. A new approach for mining access patterns is proposed in Pino. In this approach, users' navigation patterns are modeled with a directed and weighted graph. Its vertices are webpages and the edges indicate their correlation based on co-occurrence frequencies between webpages. The weight of edges is calculated on the basis of the inverse conditional probability of viewing the webpages by considering their order. Then, webpages are clustered by partitioning this graph based on shortest paths. Recommendations will be generated based on discovered patterns with constant time complexity and with consistency to the statelessness property of HTTP protocol. Pino has been evaluated on web server logs. The effectiveness of recommendations has been evaluated by criteria applicability recommendation, correct recommendations, accuracy and coverage. Evaluation results indicate the ability of the system to improve quality of recommendations so that the harmony mean of these criteria in Pino system has reached 57% improved by 12% compared to previous systems.

Keywords: Recommender system, Web usage mining, Access pattern mining, Graph shortest path.

تاریخ ارسال مقاله: ۱۳۹۵/۱۱/۰۹
تاریخ اصلاح مقاله: ۱۳۹۶/۰۲/۳۱، ۱۳۹۶/۰۶/۱۷ و ۱۳۹۶/۰۷/۱۳
تاریخ پذیرش مقاله: ۱۳۹۶/۱۱/۰۳
نام نویسنده مسئول: امیر جهانگرد رفسنجانی
نشانی نویسنده مسئول: ایران - یزد - بلوار دانشگاه - دانشگاه یزد – پردیس فنی و مهندسی – گروه مهندسی کامپیوتر.

۱- مقدمه

استفاده از وب راحتی، سرعت، صرفه‌جویی در زمان و هزینه‌ها را به دنبال دارد. به همین دلیل وب کاربردهای فراوانی مانند تجارت الکترونیک، دولت الکترونیک و سامانه‌های اطلاعاتی دارد. فراگیر شدن وب در همه زمینه‌ها، حجم اطلاعات موجود در وب را افزایش داده است. حجم انبوه اطلاعات وب، مشکلاتی را در استفاده از آن به وجود می‌آورد؛ از جمله اینکه دستیابی کاربران به اطلاعات مفید و مرتبط با نیازها و اولویت‌هایشان را دشوار می‌کند. شخصی‌سازی وب روشی برای از بین بردن این مشکل است.

هر عملی که وب را به نیازها، علایق و اولویت‌های کاربران نزدیک می‌کند، شخصی‌سازی وب نام دارد [۱]. سامانه‌های توصیه‌گر ابزاری برای شخصی‌سازی وب هستند. هدف سامانه‌های توصیه‌گر در وبگاه‌ها، پیشنهاد اقلام مرتبط با علاقه و اولویت‌های کاربران است. در واقع این سامانه‌ها به دنبال کاهش تلاش و زحمت کاربران برای دستیابی به اقلام مورد نیازشان هستند.

کاوش در نحوه استفاده از وب یا کاوش در سیاهه‌های وب (WUM)[1] به معنای کشف و تحلیل الگوهای موجود در سیاهه‌های وب است [۲]. پیشنهاددهی به کاربران می‌تواند با کاوش در سیاهه‌های وب صورت بگیرد. برای این منظور الگوهایی از داده‌های مربوط به نحوه استفاده کاربران از وب یعنی سیاهه‌های[2] وب سرورها یا مرورگرهای وب به دست می‌آید؛ سپس رفتارهای کاربران جاری با الگوهای کشف شده مقایسه و مجموعه پیشنهادها مشخص می‌شوند.

کاوش در سیاهه‌های وب شامل سه مرحله پیش‌پردازش، کشف الگوها و تحلیل الگوها است [۲]. پیش‌پردازش به معنای تبدیل سیاهه‌ها به شکلی مناسب برای مرحله کشف الگوها است. به بیانی دیگر شیوه‌های دسترسی یا نشست‌های کاربران را مشخص می‌کند. مرحله پیش‌پردازش شامل عملیات‌های پاک‌سازی سیاهه‌ها، شناسایی کاربران، شناسایی نشست‌ها و تکمیل مسیر است [۳]. در مرحله کشف الگوها با استفاده از الگوریتم‌های داده‌کاوی الگوهای دسترسی کاربران استخراج می‌شوند. الگوهای دسترسی رابطه‌هایی را بین عامل‌ها و اشیای وب مانند کاربران یا صفحات وبگاه بیان می‌کنند. در نهایت نیز این الگوها با توجه به کاربرد نهایی یعنی پیشنهاددهی، تحلیل و استفاده می‌شوند.

در این مقاله سامانه توصیه‌گری بنام پینو مبتنی بر کاوش در سیاهه‌های وب ارائه شده است. سامانه توصیه‌گر پینو به دنبال آن است که دستیابی به صفحات مرتبط با یکدیگر در وبگاه‌ها را با اثربخشی بیشتری فراهم کند. در این سامانه، رویکرد جدیدی برای استخراج الگوهای دسترسی از سیاهه‌های وب ارائه شده است. در این رویکرد، شیوه‌های دسترسی کاربران با گرافی جهت‌دار و وزن‌دار بنام گراف شیوه‌های دسترسی مدل می‌شوند. این گراف، صفحات و ارتباط بین آن‌ها را بر اساس تکرارهای با هم صفحات در نشست‌ها نشان می‌دهد. در واقع صفحات، رأس‌های این گراف هستند و یال‌ها ارتباط بین آن‌ها را نشان می‌دهند که معکوس احتمال شرطی مشاهده صفحات برای تخصیص وزن به یال‌های این گراف بکار برده شده است. این گراف جهت‌دار است زیرا ترتیب مشاهده صفحات را در محاسبه وزن یال‌ها در نظر می‌گیرد.

در این رویکرد برخلاف روش‌های مشابه مبتنی بر مدل گراف، ترتیب شیوه‌های دسترسی کاربران در نظر گرفته می‌شود که برای پیش‌بینی درخواست‌های آتی کاربران و پیشنهاددهی بسیار مفید است.گراف شیوه‌های دسترسی کاربران بر پایه کوتاه‌ترین مسیرها بخش‌بندی می‌شود و خوشه‌هایی از صفحات مرتبط با یکدیگر به دست می‌آیند. با این روش، پیشنهاددهی در بخش برخط برخلاف روش‌های موجود به‌صورت سازگار با فراموش‌کاری پروتکل HTTP است؛ یعنی برای پیشنهاددهی در بخش برخط نیازی به نگهداری و دنبال نشست‌های کاربری نیست. همچنین پیشنهاددهی در بخش برخط با پیچیدگی زمانی ثابت انجام خواهد شد. همچنین عملکرد سامانه پینو با سیاهه‌های یک سرور مورد ارزیابی قرار گرفته است. اثربخشی پیشنهاددهی با معیارهای قابلیت پیشنهاددهی، پیشنهادهای درست، دقت و پوشش ارزیابی شده است. نتایج ارزیابی نشان می‌دهد که این سامانه می‌تواند کیفیت پیشنهادها را بهبود دهد.

ادامه مقاله بدین‌صورت است که در بخش دوم مروری بر روی کارهای مرتبط انجام شده و چندین سامانه توصیه‌گر مورد بررسی قرار گرفته است. بخش سوم جزئیات سامانه توصیه‌گر پیشنهادی یعنی سامانه پینو را بیان می‌کند. بخش چهارم به بیان جزئیات مربوط به ارزیابی سامانه پینو می‌پردازد. در نهایت این مقاله با بیان نتیجه‌گیری و کارهای پیش‌رو در بخش پنجم به پایان می‌رسد..

۲- کارهای مرتبط

سامانه Analog یکی از اولین سامانه‌های کاوش در سیاهه‌های وب است [۴]. این سامانه نشست‌های کاربران را به بردارهایی نگاشت، و با الگوریتم Leader [۵] آن‌ها را خوشه‌بندی می‌کند. این الگوریتم از نظر پیچیدگی زمانی و مکانی بسیار کارا است اما دارای مشکلاتی است. از جمله اینکه، نتیجه خوشه‌بندی به ترتیب ورودی بردارها وابسته است. سامانه Analog نشان داد که می‌توان شیوه‌های دسترسی کاربران را تحلیل کرد و به الگوهای دسترسی دست یافت. همچنین در این مقاله معماری دوبخشی برون‌خط-برخط برای سامانه‌های توصیه‌گر معرفی شد.

بخش برون‌خط الگوهای دسترسی را با استفاده از سامانه Analog استخراج می‌کند. بخش برخط صفحاتی را به کاربران برای مشاهده پیشنهاد می‌دهد. برای این منظور نشست‌های کاربرانی که اکنون وبگاه را مشاهده می‌کنند، با توجه به خوشه‌های بخش برون‌خط دسته‌بندی می‌کند. یک نشست در یک خوشه قرار می‌گیرد درصورتی‌که تعداد صفحات مشترک بین نشست و خوشه از تعدادی معین بیشتر باشد. در نهایت نیز صفحاتی از خوشه‌هایی که نشست به آن‌ها تعلق دارد، به کاربر پیشنهاد می‌شود.

سامانه توصیه‌گر WebPersonlizer دارای معماری دوبخشی برون‌خط-برخط است [۱]. این سامانه الگوها را با خوشه‌بندی صفحات

به‌طور مستقیم و بدون نیاز به خوشه‌بندی نشست‌ها استخراج می‌کند. در این سامانه ابتدا مجموعه‌های پرتکرار از صفحات با روش در [۶] به دست می‌آیند. بر اساس این مجموعه‌ها یک فراگراف[۳] ایجاد می‌شود. در نهایت خوشه‌بندی صفحات با استفاده از الگوریتم بخش‌بندی فراگراف حاصل از قواعد وابستگی انجام می‌شود [۷]. این الگوریتم با حذف فرایال‌های[۴] با وزن کمینه به‌صورت بازگشتی فراگراف را بخش‌بندی می‌کند.

در سامانه WebPersonlizer دسته‌بندی کاربران در الگوهای استخراج شده بر پایه بردارها و عملیات‌های برداری انجام می‌شود [۸]. بنابراین با افزایش تعداد صفحات، مولفه‌های بسیاری در بردارها صفر می‌شوند؛ تعیین تطابق بین نشست‌ها و خوشه‌ها دشوار می‌شود؛ درنتیجه کارایی و دقت کاهش می‌یابد.

آقای پرکویتز و همکارش یک سامانه WUM بنام PageGather برای ایجاد خودکار صفحات نمایه ارائه کرده‌اند [۹]. صفحات نمایه دارای پیوندهایی به مجموعه‌ای از صفحات مرتبط با یکدیگر هستند. الگوریتم PageGather خوشه‌هایی از صفحات مرتبط با یکدیگر را پیدا می‌کند. هر خوشه یک صفحه نمایه است. این الگوریتم گرافی غیرجهت‌دار و وزن‌دار از صفحات بر اساس معیار شباهت تکرارهای با هم صفحات در نشست‌ها ایجاد می‌کند. این الگوریتم برای خوشه‌بندی صفحات، مولفه‌های همبند و گروهک‌های بیشینه را در این گراف پیدا می‌کند. گروهک‌ها، زیرگراف‌هایی هستند که بین هر دو رأس آن‌ها یک یال وجود دارد. از طرف دیگر گروهکی بیشینه است که زیرمجموعه گروهک دیگری نباشد. هزینه‌بر بودن شناسایی گروهک‌ها در گراف یکی از مشکلات این روش است.

سامانه Yoda یک سامانه توصیه‌گر دیگر است [۱۰]. در سامانه Yoda خوشه‌بندی کاربران برای دستیابی به علائقشان بر اساس مدل ماتریس‌های ویژگی انجام می‌شود [۱۱]. در این مدل، نشست‌ها به ماتریس‌های چندبعدی تبدیل می‌شوند که هر ماتریس بیانگر یک ویژگی برای زیرمجموعه‌ای از صفحات در نشست است. سپس نشست‌ها بر اساس معیار فاصله اقلیدسی محض بهبود یافته خوشه‌بندی می‌شوند.

در سامانه Yoda خصوصیاتی برای توصیف اقلام در نظر گرفته می‌شود که دارای مقادیری غیرقطعی و کیفی هستند. برای پیشنهاددهی، نشست کاربر به‌صورت غیرقطعی دسته‌بندی می‌شود. میزان شباهت نشست کاربر با هر خوشه بر اساس معیار فاصله اقلیدسی محض بهبود یافته به دست می‌آید. سپس میزان اولویت یا اهمیت اقلام برای کاربر بر اساس میزان شباهت کاربر به خوشه‌ها و اولویت اقلام در آن‌ها محاسبه می‌شود؛ و مجموعه اقلام پیشنهادی به کاربر مشخص می‌شود. مشکل استفاده از این سامانه در برنامه‌های کاربردی تحت وب با مقیاس بزرگ و با اقلام بسیار زیاد این است که این سامانه، روشی خودکار برای مقداردهی به خصوصیات اقلام ندارد.

آقای ناکاگاوا و همکارش یک سامانه توصیه‌گر ترکیبی با توجه به تاثیرگذاری ساختار یا توپولوژی وبگاه در عملکرد روش‌های پیشنهاددهی ارائه کرده‌اند [۱۲]. ساختار یا توپولوژی وبگاه، صفحات و پیوندهای بین آن‌ها را نشان می‌دهد. پیشنهاددهی می‌تواند بر اساس قواعد وابستگی، الگوهای ترتیبی و الگوهای ترتیبی به‌هم‌پیوسته صورت بگیرد. به بیانی دیگر می‌توان این الگوها را بر پایه الگوریتم Apriori [۱۳] استخراج و آن‌ها را در داده‌ساختار درخت ترای [۱۴] ذخیره کرد. در نهایت پیشنهاددهی را با جستجوی نشست‌های کاربران در این درخت انجام داد.

ویژگی‌های ساختاری وبگاه مانند درجه اتصال پیوندها در عملکرد روش‌های پیشنهاددهی مبتنی بر الگوریتم‌های قواعد وابستگی، الگوهای ترتیبی و ترتیبی به‌هم‌پیوسته مؤثر است [۱۵]. این درجه، میزان پیوند بین صفحات را در گراف توپولوژی وبگاه مشخص می‌کند. این سامانه ترکیبی، به‌صورت خودکار بر اساس مکان قرارگیری کاربر در گراف توپولوژی وبگاه بین این روش‌های پیشنهاددهی تغییر می‌کند. برای تغییر خودکار بین روش‌های پیشنهاددهی از معادله رگرسیون لجستیک استفاده می‌کند. این معادله احتمال موفقیت هر یک از روش‌های پیشنهاددهی را با توجه به درجه اتصال پیوندها در مکان قرارگیری کاربر مشخص می‌کند. یکی از مشکلات این سامانه هزینه‌بر بودن الگوریتم Apriori برای استخراج الگوها است. از طرف دیگر استفاده از داده‌ساختار درخت ترای برای پیشنهاددهی زمانی که تعداد صفحات بسیار زیاد است اتلاف حافظه را در پی دارد.

سامانه توصیه‌گر SWARS از الگوهای دسترسی ترتیبی برای پیشنهاددهی استفاده می‌کند [۱۶]. این سامانه الگوهای دسترسی ترتیبی موجود در سیاهه‌ها را استخراج [۱۷] و آن‌ها را با استفاده از داده‌ساختار درخت ترای ذخیره می‌کند. برای پیشنهاددهی، نشست کاربر با الگوهای ترتیبی این درخت مقایسه می‌شود و مجموعه صفحات پیشنهادی به کاربر مشخص می‌شوند. برای استخراج الگوهای ترتیبی ابتدا، دنباله‌های دسترسی کاربران یا نشست‌ها در داده‌ساختاری بنام درخت شیوه‌های دسترسی ذخیره می‌شوند. این درخت مبتنی بر درخت ترای است. الگوهای ترتیبی با استفاده از این درخت استخراج می‌شوند و دیگر نیازی به نشست‌ها نیست. الگوهای ترتیبی به‌صورت بازگشتی با ایجاد درخت‌های شیوه‌های دسترسی میانی برای دنباله‌های دسترسی با پسوند یکسان به دست می‌آیند.

استخراج الگوهای ترتیبی به شیوه توضیح داده شده نسبت به الگوریتم Apriori دارای مزیت‌هایی است. از جمله اینکه این روش به دلیل استفاده از داده‌ساختار درخت شیوه‌های دسترسی، برای تعیین تعداد دفعات تکرار دنباله‌ها، نیازی به تشخیص تطابق بین آن‌ها ندارد. از طرف دیگر تنها دنباله‌های با پسوند یکسان را در نظر می‌گیرد، بنابراین مجموعه‌های بسیاری را به‌عنوان نامزد پرتکرار بودن تولید نمی‌کند. اما ایجاد درخت‌های شیوه‌های دسترسی میانی در این روش بازگشتی، هزینه‌بر است و یکی از مشکلات مهم این روش محسوب می‌شود. از طرف دیگر استفاده از داده‌ساختار درخت ترای برای پیشنهاددهی زمانی که تعداد صفحات بسیار زیاد است اتلاف حافظه را در پی دارد.

سامانه توصیه‌گر SUGGEST یک‌بخشی است و تنها بخش برخط دارد [۱۸]. سامانه SUGGEST شیوه‌های دسترسی موجود در سیاهه‌ها را به‌صورت گرافی غیرجهت‌دار و وزن‌دار نشان می‌دهد. این گراف صفحات و ارتباط بین آن‌ها را بر اساس تکرارهای با هم صفحات در نشست‌ها مدل می‌کند. این سامانه صفحات مرتبط به یکدیگر را با بخش‌بندی این گراف به مولفه‌های همبند پیدا می‌کند. هر مؤلفه همبند، خوشه‌ای از صفحات مرتبط با یکدیگر را نشان می‌دهد.

سامانه SUGGEST مولفه‌های همبند گراف را به‌گونه‌ای افزایشی بر پایه الگوریتم جستجوی در عمق گراف به دست می‌آورد. در این سامانه برای حل محدودیت حافظه در بخش برخط از روشی مبتنی بر تکنیک اخیراً کمتر استفاده شده [۱۹] استفاده شده است. برای پیشنهاددهی ابتدا نشست کاربر با توجه به خوشه‌ها دسته‌بندی می‌شود. نشست کاربر در خوشه‌ای قرار می‌گیرد که تعداد صفحات مشترک بین آن‌ها بیشینه باشد. سپس صفحاتی از آن خوشه به کاربر پیشنهاد داده می‌شود. این سامانه برای وبگاه‌هایی با تعداد و تغییرات زیاد صفحات باعث کاهش کارایی سرور خواهد شد زیرا تعداد جایگزین کردن صفحات افزایش می‌یابد.

سامانه WebPUM بر اساس کاوش در سیاهه‌های وب رفتار آتی کاربران را پیش‌بینی می‌کند و صفحاتی را به آن‌ها پیشنهاد می‌دهد [۲۰]. مدل‌سازی شیوه‌های دسترسی کاربران به‌صورت برون‌خط با گرافی غیرجهت‌دار و وزن‌دار انجام می‌شود. برای تخصیص وزن به یال‌ها از تکرارهای با هم صفحات در نشست‌ها، فاصله زمانی بین آن‌ها و جایگاه آن‌ها در نشست‌ها استفاده می‌شود. الگوهای دسترسی کاربران با استفاده از این گراف استخراج می‌شود. مولفه‌های همبند این گراف، الگوهای دسترسی کاربران در نظر گرفته می‌شوند که مجموعه‌ای از صفحات مرتبط با یکدیگر را نشان می‌دهند. برای پیشنهاددهی در بخش برخط، نشست‌های کاربران با توجه به الگوهای دسترسی بخش برون‌خط دسته‌بندی می‌شوند. برای این منظور از الگوریتم بزرگ‌ترین زیر دنباله مشترک [۲۱] استفاده می‌شود. در این سامانه در بخش برخط برای پیشنهاددهی به دلیل نیاز به دسته‌بندی نشست‌ها و ردیابی آن‌ها، سربار به وجود می‌آید.

در سامانه توصیه‌گر [۲۲] صفحات وبگاه به بردارهایی از کلیدواژه‌ها تبدیل می‌شوند. وزن کلیدواژه‌ها بر اساس تعداد تکرارشان در آن صفحه و تعداد صفحات دارای آن‌ها محاسبه می‌شود. نشست‌ها نیز به بردارهایی از صفحات تبدیل می‌شوند. وزن صفحات بر اساس مدت‌زمان مشاهده صفحه در نشست و تعداد کل دستیابی‌های به آن صفحه در تمامی نشست‌ها محاسبه می‌شود. سپس صفحات و نشست‌ها به‌صورت جداگانه با استفاده از روشی فازی مبتنی بر الگوریتم ژنتیک خوشه‌بندی می‌شوند. برداری برای تمامی صفحه‌ها و نشست‌ها به دست می‌آید که میزان عضویت آن صفحه و نشست را در هر خوشه از صفحات و نشست‌ها مشخص می‌کند. صفحات و کاربران مشابه با محاسبه فاصله اقلیدسی بین این بردارها تعیین می‌شوند. این سامانه بر اساس این شباهت‌ها به کاربران صفحاتی را برای مشاهده پیشنهاد می‌دهد.

در [۲۳] برای پیش‌بینی صفحات آتی کاربران از ترکیب خوشه‌بندی و مدل مارکوف استفاده شده است. به بیانی دیگر نشست‌ها در این روش به بردارهایی از صفحات تبدیل می‌شوند که وزن صفحات در این بردارها بر اساس مدت‌زمان مشاهده و تعداد دفعات تکرار صفحه در نشست محاسبه می‌شود. نشست‌ها بر اساس روشی بر پایه الگوریتم k-means خوشه‌بندی می‌شوند که شباهت بین نشست‌ها بر اساس شباهت کسینوسی محاسبه می‌شود. در نهایت یک مدل مارکوف برای هر خوشه ایجاد و بر اساس آن صفحات آتی کاربران پیش‌بینی می‌شود.

در سامانه توصیه‌گر [۲۴] از نظریه مجموعه‌های فازی ناهموار برای پیشنهاددهی استفاده شده است. این نظریه روش‌هایی را با استفاده از ابزار ریاضی ارائه می‌دهد که با تحلیل اطلاعات نهان در داده‌ها، اطلاعات غیرمرتبط را از مجموعه داده‌ها حذف می‌کند. بنابراین این سامانه با استفاده از مجموعه‌های فازی ناهموار، اطلاعات کاربرانی را می‌یابد که بر مبنای آن‌ها می‌توان پیش‌بینی بهتری از علائق کاربران انجام داد. سپس شباهت بین این کاربران را بر اساس فاصله اقلیدسی محاسبه می‌کند تا بتواند بر اساس آن به کاربران پیشنهادهایی را ارائه دهد. زمان و حجم زیاد محاسبات یکی از معایب این روش است.

جستجوی همساز[۵] یک الگوریتم بهینه‌سازی و مبتنی بر جمعیت شبیه به الگوریتم ژنتیک است. جستجوی همساز با تعدادی جواب اولیه کار خود را آغاز می‌کند، جواب جدیدی را مرحله‌به‌مرحله تولید می‌کند تا اینکه شرایط پایان الگوریتم برآورده شود. الگوریتم جستجوی همساز در هر مرحله بر اساس احتمالی از پیش تعیین شده تصمیم می‌گیرد که جواب جدید بر پایه جواب‌های قبلی به همراه تغییری احتمالی در آن ایجاد شود یا اینکه به‌صورت تصادفی از مجموعه مقادیر ممکن انتخاب شود.

دو سامانه توصیه‌گر HSCR و IHKSCR در [۲۵] معرفی شده‌اند. نشست‌ها در این سامانه‌ها به بردارهایی دودویی از صفحات تبدیل می‌شوند که ۱ و ۰ به ترتیب بیانگر مشاهده و عدم مشاهده صفحه در نشست است. این نشست‌ها در سامانه توصیه‌گر HSCR با روشی بر پایه الگوریتم جستجوی همساز خوشه‌بندی می‌شوند. سامانه توصیه‌گر IHKSCR با توجه به ماهیت الگوریتم جستجوی همساز یعنی جستجوی کلی فضای مسئله برای تعیین محدوده جواب بهینه، و همچنین بر اساس ماهیت الگوریتم k-means یعنی جستجو در نزدیکی جواب‌های اولیه برای بهبود آن‌ها، نشست‌ها را با روشی ترکیبی بر پایه روش خوشه‌بندی سامانه HSCR و الگوریتم k-means خوشه‌بندی می‌کند. در نهایت پیشنهاددهی به کاربران بر پایه خوشه‌های استخراجی صورت می‌گیرد.

همه سامانه‌های توصیه‌گر بررسی شده دارای هدف مشترکی بودند. در همه این سامانه‌ها سعی بر آن بود که معماری و الگوریتمی مناسب برای پیشنهاددهی ایجاد شود. به‌گونه‌ای که پیچیدگی، دقت، کارایی

سامانه توصیه‌گر و کیفیت پیشنهادها در حد مطلوب باشد. اما همچنان این موضوع یک مسئله چالش‌برانگیز است.

۳- سامانه توصیه‌گر پینو

هر وبگاه را می‌توان به‌صورت مجموعه‌ای از اقلام در نظر گرفت. محصولات، خدمات، صفحات یا هر چیزی که توسط وبگاه ارائه می‌شود، جزء اقلام آن هستند. هدف سامانه‌های توصیه‌گر ارائه مجموعه‌ای از اقلام به کاربر بر اساس اولویت‌ها و نیازهای اوست. در این بخش سامانه توصیه‌گری بنام پینو معرفی می‌شود. این سامانه توصیه‌گر با استفاده از WUM، مجموعه‌ای از صفحات را برای مشاهده به کاربران پیشنهاد می‌دهد. در ادامه معماری و نحوه عملکرد سامانه پینو مورد بررسی قرار گرفته است.

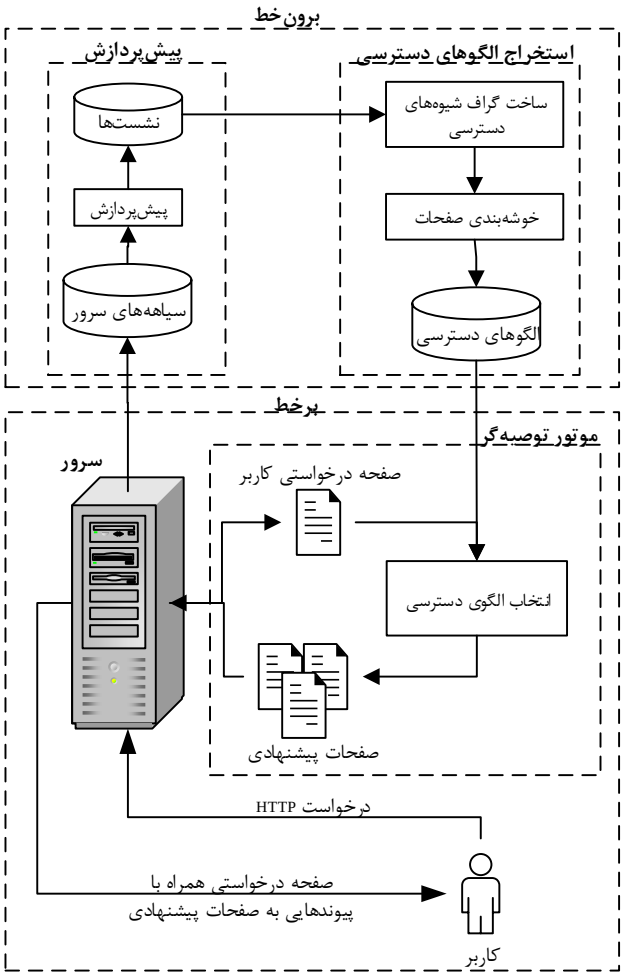
۳-۱- معماری سامانه

سامانه پینو دارای معماری دوبخشی برون‌خط-برخط است. شکل ۱ معماری این سامانه را نشان می‌دهد. اگرچه بخش‌های برون‌خط و برخط از یکدیگر جدا هستند اما وابستگی زیادی میان آن‌ها وجود دارد. بخش برون‌خط شامل پیش‌پردازش و کشف الگوهای دسترسی است. موتور توصیه‌گر در بخش برخط قرار دارد.

پیش‌پردازش، شیوه‌های دسترسی یا نشست‌های کاربران را از سیاهه‌های وب استخراج می‌کند. نشست‌ها ورودی قسمت کشف الگوهای دسترسی هستند. در قسمت کشف الگوهای دسترسی، رویکرد جدیدی برای استخراج الگوهای دسترسی ارائه شده است. ابتدا گراف شیوه‌های دسترسی از نشست‌ها ایجاد می‌شود. صفحات، رأس‌های این گراف و یال‌ها نشان‌دهنده ارتباط بین آن‌ها هستند. وزن یال‌ها بر اساس معکوس احتمال شرطی مشاهده صفحات با در نظرگرفتن ترتیب آن‌ها محاسبه می‌شود. خوشه‌بندی صفحات بر پایه کوتاه‌ترین مسیرها در این گراف انجام می‌شود تا الگوهای دسترسی به دست آیند. در بخش برخط از الگوهای دسترسی برای ارائه پیشنهادهای پویا به کاربران استفاده می‌شود. منظور از پیشنهادهای پویا، پیشنهادهایی است که به مرور زمان تغییر می‌کنند و ثابت نیستند. موتور توصیه‌گر صفحات پیشنهادی به کاربر را مشخص می‌کند. برای این منظور هنگامی‌که کاربر یک صفحه را درخواست می‌کند؛ موتور توصیه‌گر الگوی دسترسی مربوط به درخواست کاربر را بازیابی می‌کند؛ آن‌ها را به‌عنوان مجموعه صفحات پیشنهادی به کاربر ارائه می‌دهد. در ادامه بخش‌های سامانه پینو با جزییات بیشتر توضیح داده شده‌اند.

۳-۲- پیش‌پردازش

پیش‌پردازش یکی از مراحل کلیدی در کشف الگوهای دسترسی از سیاهه‌های وب است. پیش‌پردازش به دنبال تبدیل سیاهه‌ها به نشست‌های کاربران است. مراحل پیش‌پردازش شامل پاک‌سازی سیاهه‌ها، شناسایی کاربران، شناسایی نشست‌ها و تکمیل مسیر پیش از استخراج الگوهای دسترسی انجام می‌شود [۳].



شکل ۱: معماری سامانه پینو

وب سرورها به‌صورت خودکار دسترسی‌ها، حرکت‌ها و فعالیت‌های کاربران را در سیاهه‌ها ثبت می‌کنند. با توجه به تنظیمات وب سرورها، انواع گوناگونی از سیاهه‌ها وجود دارد. اما معمولاً همه آن‌ها قسمت‌های کلیدی مانند آدرس IP، زمان درخواست، URL درخواستی، مشخصات سیستمی و قسمت ارجاع‌دهنده را دارند. همه اطلاعات یا سطرهای سیاهه‌ها برای کشف الگوها لازم نیستند. در پاک‌سازی سیاهه‌ها تنها سطرهایی از سیاهه‌ها نگه داشته می‌شوند که اطلاعات لازم در مرحله کشف الگوهای دسترسی را دارند. عملیات پاک‌سازی سیاهه‌ها، سطرهای حاصل از موارد اضافی مانند خزشگرها یا ارجاع به تصاویر را حذف می‌کند تا به ازای هر بازدید صفحه تنها یک سطر در سیاهه وجود داشته باشد.

بعد از پاک‌سازی، سیاهه‌ها به‌صورت مجموعه‌ای از دسترسی‌های کاربران به صفحات به ترتیب صعودی زمان هستند که مشخص نیست هر دسترسی مربوط به کدام کاربر یا نشست است. منظور از نشست کاربر کلیه فعالیت‌هایی است که کاربر از زمان ورود به وبگاه تا زمان خروج از آن انجام می‌دهد. بنابراین باید کاربران و نشست‌های آن‌ها از سیاهه‌ها استخراج شوند.

شناسایی کاربران و نشست‌ها از سیاهه‌ها با دشواری‌هایی همراه است. از جمله اینکه معمولاً کاربران در وبگاه‌ها ناشناس هستند. از طرفی

دیگر پروتکل HTTP فراموش‌کار است یعنی رابطه‌ای بین درخواست‌های دریافتی تشخیص نمی‌دهد و به هر درخواست، مستقل از درخواست‌های دیگر پاسخ می‌دهد. درنتیجه سرورها در حالت عادی نمی‌توانند رفتار کاربران را به‌صورت انفرادی دنبال کنند. همچنین تمامی درخواست‌ها، از سوی یک سرور پیشکار دارای IP یکسان هستند اگرچه کاربرانی متفاوت مبدأ واقعی درخواست‌ها هستند. همین‌طور مشاهده صفحات موجود در حافظه‌های نهان میان کاربر و سرور در سیاهه‌ها ثبت نمی‌شود.

شناسایی کاربران و نشست‌ها می‌تواند با استفاده از کوکی‌ها صورت بگیرد. کوکی‌ها داده‌هایی هستند که وب سرورها به کاربران در ابتدای بازدیدشان ارسال می‌کنند. در این روش، هم‌زمان با مشاهده وبگاه توسط کاربران مبدأ هر درخواست مشخص می‌شود. اما استفاده از کوکی‌ها به دلیل عدم تمایل کاربر یا محدودیت‌های وب سرورها همیشه امکان‌پذیر نیست.

شناسایی کاربران و نشست‌ها می‌تواند بعد از پایان بازدید کاربران از وبگاه انجام شود. برای این منظور می‌توان از روش‌های تقریبی استفاده کرد. در واقع شناسایی کاربران می‌تواند به‌صورت تقریبی با استفاده از قسمت‌های آدرس IP، نوع سیستم عامل و مرورگر انجام گیرد [۳]. به بیانی دیگر دسترسی‌هایی که آدرس IP، سیستم عامل و مرورگر یکسانی دارند، به یک کاربر نسبت داده می‌شوند. سپس می‌توان براساس قسمت ارجاع‌دهنده و توپولوژی وبگاه دسترسی‌هایی را تقریب زد که مربوط به کاربرانی متفاوت با آدرس IP یکسان هستند [۳]. پس از شناسایی تقریبی تمامی دسترسی‌های مربوط به یک کاربر، باید آن‌ها را به نشست‌هایی تقسیم کرد.

شناسایی نشست‌ها می‌تواند به‌صورت تقریبی با قرار دادن آستانه بر روی مدت‌زمان نشست یا مشاهده صفحه انجام گیرد [۳]. به‌عبارتی‌دیگر می‌توان به‌صورت تجربی متناسب با شرایط و ویژگی‌های وبگاه برای مدت‌زمان نشست حدی مشخص کرد، یعنی مدت‌زمانی را که کاربران در یک نشست می‌گذارند محدود فرض کرد. سپس بر اساس این محدودیت، دسترسی‌های کاربران را به نشست‌ها تقسیم کرد. همچنین می‌توان بر روی مدت‌زمان مشاهده یک صفحه آستانه‌ای مشخص کرد، یعنی اختلاف بین دو درخواست متوالی از سوی یک کاربر را محدود فرض کرد. اگر این اختلاف از مقدار مشخص شده بیشتر باشد آنگاه می‌توان نشست جدیدی را برای کاربر در نظر گرفت. بعد از شناسایی نشست‌ها می‌توان براساس قسمت ارجاع‌دهنده و توپولوژی وبگاه دسترسی‌هایی را در نشست‌ها تقریب زد که از حافظه نهان مرورگر صورت گرفته‌اند [۳].

در پیش‌پردازش می‌توان محدودیت‌هایی را به‌منظور افزایش کارایی و کاهش نویز با توجه به ماهیت کاربرد اعمال کرد [۱]. از جمله اینکه می‌توان برخی از نشست‌ها را با اعمال محدودیت بر روی تعداد صفحات موجود در نشست‌ها حذف کرد. به‌عنوان مثال نشست‌های با طول کوتاه ضرورتی ندارند و می‌توان آن‌ها را حذف کرد. علاوه بر این می‌توان صفحات با فراوانی بسیار کم را حذف کرد؛ یعنی صفحاتی که بسیار کم در نشست‌ها وجود دارند. همچنین صفحات تکراری در نشست‌ها نیز قابل حذف است. زیرا همواره کاربران در یک نشست برای پیدا کردن اطلاعات جدید صفحات میانی بسیاری مشاهده و چندین مسیر را طی می‌کنند.

۳-۳- استخراج الگوهای دسترسی

در پینو رویکرد جدیدی برای کشف الگوهای دسترسی از سیاهه‌های وب ارائه شده است. به بیانی دیگر برای خوشه‌بندی شیوه‌های دسترسی کاربران، رویکرد جدیدی معرفی شده است که مبتنی بر بخش‌بندی گراف شیوه‌های دسترسی کاربران است. مجموعه نشست‌های حاصل از پیش‌پردازش مشخص می‌کند که کاربران چه صفحاتی را با چه ترتیب و مدت‌زمانی مشاهده کرده‌اند. بنابراین شیوه‌های دسترسی کاربران در این مجموعه قرار دارند.

نشست‌ها یا شیوه‌های دسترسی، ورودی مرحله استخراج الگوهای دسترسی هستند. الگوهای دسترسی، خروجی این مرحله هستند که رابطه‌هایی بین صفحات وبگاه را بیان می‌کنند. این الگوها در بخش برخط برای پیش‌بینی درخواست‌های آتی کاربران و ارائه پیشنهادهای پویا به آن‌ها مورد استفاده قرار می‌گیرند.

ویژگی‌های مشترک بین شیوه‌های دسترسی گروهی از کاربران، اولویت‌های آن‌ها را مشخص می‌کند. زیرا کاربران اولویت‌ها، نیازها و علائق مشترکی در طول زمان دارند. الگوهای دسترسی را می‌توان این ویژگی‌های مشترک در نظر گرفت [۲۶]. همچنین این الگوها را می‌توان با استفاده از روش‌های خوشه‌بندی استخراج نمود. زیرا این روش‌ها مجموعه‌ای از اشیا را به زیرمجموعه‌ها یا خوشه‌هایی تقسیم می‌کنند که شباهت بین اشیای در یک خوشه نسبت به خوشه‌های دیگر بیشتر است. خوشه‌ها را می‌توان الگوهای دسترسی در نظر گرفت.

در اینجا هدف بر این است که با خوشه‌بندی، صفحات مرتبط با یکدیگر مشخص شوند. البته مانعی ندارد که صفحه‌ای در بیش از یک خوشه قرار بگیرد. همان‌گونه که می‌توان اسناد را به بردارهایی از کلمات نگاشت کرد و با محاسبه زاویه بین بردارهایشان آن‌ها را خوشه‌بندی کرد. در اینجا نیز می‌توان صفحات و رابطه بین آن‌ها را بر اساس معیار تکرار با هم آن‌ها در نشست‌ها به یک گراف نگاشت کرد [۹]. سپس از الگوریتم‌های گراف برای دستیابی به خوشه‌ها یا مجموعه صفحات مرتبط به هم استفاده کرد. البته صفحات بر پایه فرض بازدید زنجیره‌وار[۶] خوشه‌بندی می‌شوند. بر اساس این فرض یک ارتباط مفهومی وجود دارد، بین صفحاتی که کاربر در یک نشست مشاهده می‌کند [۹].

الگوهای دسترسی رفتار گروهی از کاربران را بر اساس اولویت‌ها و نیازهای مشترکشان بیان می‌کنند. به‌عبارتی‌دیگر رابطه‌ای بین صفحات p_i به ازای $1\le i\le n$ در وبگاه بیان می‌کنند که n تعداد صفحات وب‌گاه است. الگوهای دسترسی توسط مجموعه $Clusters=\{cluster(p_1),cluster(p_2),cluster(p_3),\ldots,cluster(p_n)\}$ نشان داده می‌شوند. هر $cluster(p_i)$ به ازای $1\le i\le n$ مجموعه $\langle p_{i_1},p_{i_2},p_{i_3},\ldots,p_{i_k}\rangle$ است که صفحات مرتبط با صفحه p_i را بر

اساس میزان ارتباطشان با آن مرتب نگه می‌دارد. در این مجموعه مرتب، اولین صفحه یعنی p_{i_1} بیشترین ارتباط را با صفحه p_i دارد. البته ممکن است که برای صفحاتی مثل p_i هیچ صفحه مرتبطی پیدا نشود یعنی $cluster(p_i)=\varnothing$.

برای خوشه‌بندی صفحات در مرحله اول، گرافی جهت‌دار و وزن‌دار از صفحات بر اساس میزان ارتباط بین آن‌ها ایجاد می‌شود. به بیانی دیگر درجه ارتباط بین صفحات بر اساس معیار تعداد تکرارهای با هم صفحات در نشست‌ها محاسبه می‌شود. درجه‌های ارتباط دوبه‌دوی صفحات را می‌توان در یک ماتریس ذخیره کرد. این ماتریس می‌تواند بیانگر ماتریس هم‌جواری گرافی جهت‌دار و وزن‌دار باشد. معکوس احتمال شرطی مشاهده صفحات برای تخصیص وزن به یال‌های این گراف بکار گرفته شده است. در مرحله دوم، گراف بر پایه کوتاه‌ترین مسیرها بخش‌بندی می‌شود. با این عمل صفحات خوشه‌بندی و مجموعه صفحات مرتبط با یکدیگر مشخص می‌شوند. در ادامه این دو مرحله با جزئیات بیشتر توضیح داده شده‌اند.

۳-۳-۱- ایجاد گراف جهت‌دار و وزن‌دار

گراف جهت‌دار و وزن‌دار $G=(V,E)$ گراف شیوه‌های دسترسی کاربران است که شیوه‌های دسترسی یا نشست‌های کاربران را مدل می‌کند به بیانی دیگر صفحات وبگاه رأس‌های این گراف هستند یعنی $V(G)=\{p_1,p_2,p_3,\ldots,p_n\}$ که p_i به ازای $1\le i\le n$ صفحات وبگاه هستند و n تعداد آن‌ها است. $E(G)$ مجموعه یال‌های بین رأس‌ها است که رابطه بین صفحات را نشان می‌دهد. ترتیب مشاهده صفحات در نشست‌ها مهم است بنابراین این یال‌ها جهت‌دار هستند. در حقیقت برای هر جفت از صفحات می‌توان درجه ارتباط بین آن‌ها را بر اساس تعداد تکرارهای با هم صفحات در نشست‌ها محاسبه کرد [۹]. درجه‌های ارتباط بین صفحات را می‌توان در ماتریس M ذخیره کرد به‌گونه‌ای که هر درایه M_{ij} در این ماتریس بیانگر درجه ارتباط بین جفت صفحات $\langle p_i,p_j\rangle$ است. در این حالت ماتریس M را می‌توان به‌عنوان ماتریس هم‌جواری گراف جهت‌دار و وزن‌دار $G=(V,E)$ یعنی گراف شیوه‌های دسترسی کاربران در نظر گرفت. که از رابطه زیر برای تخصیص وزن به یال‌ها استفاده شده است:

$$W\langle p_i,p_j\rangle=\frac{1}{Prob(p_j|p_i)} \qquad (۱)$$

در رابطه (۱)، $W\langle p_i,p_j\rangle$ بیانگر وزن یال بین صفحات p_i و p_j است. در حقیقت یالی جهت‌دار از رأس p_i به رأس p_j با وزن $W\langle p_i,p_j\rangle$ را نشان می‌دهد. در این رابطه $Prob(p_j|p_i)$ احتمال شرطی بین صفحات p_i و p_j را مشخص می‌کند. احتمال صفحه p_j به‌شرط صفحه p_i ، احتمال بازدید صفحه p_j را بعد از بازدید صفحه p_i نشان می‌دهد. به‌عبارتی‌دیگر $Prob(p_j|p_i)$ احتمال آن است که کاربر صفحه p_j را مشاهده کند به‌شرط آنکه پیش از آن صفحه p_i را مشاهده کرده باشد. این مقدار به‌صورت نسبت تعداد دفعاتی که صفحه p_j بعد از صفحه p_i وجود دارد به‌کل تعداد دفعات وجود صفحه p_i در نشست‌ها محاسبه می‌شود. ازآنجایی‌که $Prob(p_j|p_i)\in[0,1]$ بنابراین طبق رابطه (۱) داریم $W\langle p_i,p_j\rangle\in[1,\infty)$ یعنی وزن یال‌ها عددی بزرگ‌تر مساوی ۱ است. طبق رابطه (۱) هر چه که $W\langle p_i,p_j\rangle$ به عدد ۱ نزدیک‌تر باشد، احتمال بازدید صفحه p_j بعد از بازدید صفحه p_i بیش‌تر است و ارتباط بین صفحات p_i و p_j بیشتر است.

۳-۳-۲- بخش‌بندی گراف و خوشه‌بندی صفحات

روش‌های بخش‌بندی گراف مانند کشف مولفه‌های همبند یا گروهک‌ها، یک گراف را به رأس‌هایی تقسیم می‌کنند. در سامانه پینو، خوشه‌بندی صفحات با بخش‌بندی گراف شیوه‌های دسترسی انجام می‌شود. برای این منظور الگوریتم خوشه‌بندی مبتنی بر کوتاه‌ترین مسیرهای گراف ارائه شده است. الگوریتم ۱، الگوریتم خوشه‌بندی پیشنهادی را نشان می‌دهد.

الگوریتم خوشه‌بندی ابتدا گراف شیوه‌های دسترسی کاربران را ایجاد می‌کند؛ یعنی صفحات وبگاه را به‌عنوان رأس‌های گراف جهت‌دار و وزن‌دار $G=(V,E)$ در نظر می‌گیرد. $V(G)=\{p_1,p_2,p_3,\ldots,p_n\}$ که p_i به ازای $1\le i\le n$ صفحات وبگاه هستند و n تعداد آن‌ها است. در سطرهای ۱ و ۲ الگوریتم، با استفاده از رابطه (۱) یال‌های جهت‌دار و وزن‌دار بین صفحات مشخص می‌شود. این مقادیر در ماتریس هم‌جواری گراف $G=(V,E)$ یعنی ماتریس M ذخیره می‌شوند. رابطه (۱) با استفاده از مجموعه نشست‌های کاربران محاسبه می‌شود؛ بنابراین مجموعه نشست‌ها یا شیوه‌های دسترسی کاربران ورودی این الگوریتم هستند.

یال‌ها و وزن‌هایشان طبق رابطه (۱)، ارتباط بین صفحات را نشان می‌دهند. برای کاهش تأثیر صفحاتی که زیاد به یکدیگر مرتبط نیستند، بر روی وزن یال‌ها یک آستانه قرار داده می‌شود. محدودیت بر روی وزن یال‌ها با استفاده از متغیری بنام *MaxDist* اعمال می‌شود. مقدار عددی وزن یال با ارتباط بین صفحات آن یال طبق رابطه (۱)، نسبت معکوس دارد. بنابراین مقدار عددی وزن یال‌ها نباید از مقدار *MaxDist* بیشتر باشد. سطرهای ۳ تا ۵ این الگوریتم، پالایش یال‌ها را انجام می‌دهد. و یال‌هایی که وزن آن‌ها از مقدار *MaxDist* بیشتر باشد، از گراف $G=(V,E)$ حذف خواهند شد.

در مرحله بعد الگوریتم خوشه‌بندی یعنی سطر ۶، از الگوریتم فلوید-وارشال [۲۷] برای تعیین کوتاه‌ترین مسیرها بین تمامی رأس‌های گراف $G=(V,E)$ استفاده می‌شود. با استفاده از الگوریتم فلوید-وارشال دو ماتریس به دست می‌آید. ماتریس اول وزن کوتاه‌ترین مسیرها بین تمامی رأس‌های گراف را نشان می‌دهد. کوتاه‌ترین مسیرها بین رأس‌های گراف از ماتریس دوم به دست می‌آید. در این الگوریتم ماتریس اول و دوم به ترتیب *APSD* و *APSP* نام‌گذاری شده‌اند.

هدف نهایی، خوشه‌بندی صفحات با استفاده از بخش‌بندی گراف است.

الگوریتم ۱: الگوریتم خوشه‌بندی

Input:
Cleaned, Filtered and Sessionized Log File
MaxDist
MaxClsSize
Output:
Clusters – A List of Clusters
Method:
1. **for all** $p_i, p_j \in V(G)$
2. **do** $\mathrm{M}[p_i, p_j] \leftarrow W\langle p_i, p_j \rangle$
3. **for all** $e = \langle p_i, p_j \rangle$, $e \in E(G)$
4. **do if** $W\langle p_i, p_j \rangle > MaxDist$
5. **then** $E(G) \leftarrow E(G) - \{e\}$
6. *APSD and APSP matrices computed by the Floyd – Warshall algorithm for* $G = (V, E)$ *with adjacency matrix* M
7. **for all** $p_i, p_j \in V(G)$
8. **do if** $APSD[p_i, p_j] \neq \infty$
9. **then** Add p_j with $\mathrm{DM}(p_i, p_j)$ to $Clusters(p_i)$
10. **for all** $p_i \in V(G)$
11. **do** $Clusters(p_i) \leftarrow$ *Sorting ascending all* $p_j \in Clusters(p_i)$ *based on* $\mathrm{DM}(p_i, p_j)$
12. $Clusters(p_i) \leftarrow$ *Removing all* $p_j \in Clusters(p_i)$ *after MaxClsSize index*
13. **return** *Clusters*

به بیانی دیگر هدف آن است صفحات مرتبط با یکدیگر مشخص شوند. در سطرهای ۷ تا ۹ الگوریتم خوشه‌بندی، به ازای هر صفحه مانند $p_i \in V(G)$ خوشه‌ای با نماد $Clusters(p_i)$ در نظر گرفته می‌شود؛ در واقع *Clusters* حاوی همه خوشه‌های استخراجی است. در خوشه صفحه p_i یعنی $Clusters(p_i)$ همه صفحات $p_j \in V(G)$ وجود دارند که طبق الگوریتم فلوید-وارشال کوتاه‌ترین مسیری از رأس متناظر با صفحه p_i به رأس متناظر با صفحه p_j در گراف شیوه‌های دسترسی وجود دارد. واضح است که این امکان وجود دارد، صفحه p_j در بیش از یک خوشه قرار بگیرد. یک درجه عضویت به هر صفحه در $Clusters(p_i)$ نسبت داده می‌شود که میزان عضویت صفحه در خوشه را مشخص می‌کند. درجه عضویت صفحه p_j به خوشه صفحه p_i از رابطه زیر به دست می‌آید:

$$DM(p_i, p_j) = \frac{APSD[p_i, p_j]}{Pathlength(p_i, p_j)} \qquad (۲)$$

در رابطه (۲) نماد $\mathrm{DM}(p_i, p_j)$ درجه عضویت صفحه p_j به خوشه صفحه p_i را مشخص می‌کند. نماد $APSD[p_i, p_j]$ بیانگر درایه (i, j) ماتریس *APSD* است. این درایه از ماتریس *APSD* وزن کوتاه‌ترین مسیر از رأس متناظر با صفحه p_i به رأس متناظر با صفحه p_j یعنی مجموع وزن یال‌های این مسیر را در گراف شیوه‌های دسترسی بیان می‌کند. نماد $PathLength(p_i, p_j)$ طول کوتاه‌ترین مسیر از رأس متناظر با صفحه p_i به رأس متناظر با صفحه p_j یعنی تعداد یال‌های این مسیر است. مقدار $PathLength(p_i, p_j)$ با استفاده از ماتریس *APSP* محاسبه می‌شود. در حقیقت مقدار $\mathrm{DM}(p_i, p_j)$ با میانگین وزن یال‌های موجود در کوتاه‌ترین مسیر از رأس متناظر با صفحه p_i به رأس متناظر با صفحه p_j در گراف شیوه‌های دسترسی برابر است.

تعیین مجموعه‌ای کوچک از صفحات مرتبط با یکدیگر هدف است؛ از طرف دیگر خوشه‌ها برای پیشنهاددهی بکار می‌روند و نمی‌توان به کاربران تعداد زیادی پیشنهاد ارائه داد. بنابراین بر روی تعداد اعضای خوشه، محدودیت قرار داده می‌شود. این محدودیت با استفاده از متغیر *MaxClsSize* اعمال می‌شود. در سطرهای ۱۰ تا ۱۲ الگوریتم خوشه‌بندی، خوشه‌ها مورد بررسی قرار می‌گیرند. به بیانی دیگر صفحات هر خوشه بر اساس درجه عضویتشان در آن خوشه به‌صورت صعودی مرتب می‌شوند. و تنها چند صفحه ابتدایی در هر خوشه با توجه به مقدار متغیر *MaxClsSize* نگه داشته می‌شود و بقیه صفحات از آن خوشه حذف می‌شوند. در سطر ۱۳ الگوریتم خوشه‌بندی، خوشه‌های استخراجی یعنی مجموعه *Clusters* به‌عنوان نتیجه نهایی برگردانده می‌شوند.

خوشه‌هایی از صفحات مرتبط با یکدیگر بر پایه کوتاه‌ترین مسیرها در گراف شیوه‌های دسترسی ایجاد شد. این خوشه‌ها، همان الگوهای دسترسی هستند و می‌توان از آن‌ها برای پیشنهاددهی استفاده کرد. لازم به ذکر است که سیاهه‌ها ثابت نیستند و در طول زمان تغییر می‌کنند. به‌عبارتی‌دیگر همواره سطرها یا دسترسی‌های جدیدی به آن‌ها اضافه می‌شود. در این مدل فرض می‌شود که سیاهه‌ها برای یک دوره ثابت هستند. بنابراین الگوهای دسترسی باید به‌صورت دوره‌ای به‌روزرسانی شوند تا دسترسی‌های جدید را در نظر بگیرند.

۳-۴- موتور توصیه‌گر

الگوهای دسترسی در بخش برون‌خط مشخص می‌شوند. موتور توصیه‌گر در بخش برخط به دنبال پیش‌بینی درخواست‌های آتی کاربران و پیشنهاددهی به آن‌ها بر پایه الگوهای دسترسی است. الگوریتم ۲، الگوریتم پیشنهاددهی ارائه شده را نشان می‌دهد.

فرض می‌شود که S_j نشست شماره j و شیوه دسترسی کاربری دلخواه در بخش برخط است. این کاربر تاکنون صفحات $p_{j_1}, p_{j_2}, p_{j_3}, \ldots, p_{j_{k-1}}$ را به ترتیب دیده است. به‌عبارتی‌دیگر $S_j = \langle p_{j_1}, p_{j_2}, p_{j_3}, \ldots, p_{j_{k-1}} \rangle$ است. از طرف دیگر این کاربر در ادامه بازدید خود از وبگاه، صفحه p_{j_k} را درخواست کرده است. در این لحظه می‌توان مجموعه صفحاتی را به کاربر پیشنهاد داد. برای پیشنهاددهی به کاربر و پیش‌بینی آینده او می‌توان اندکی از گذشته او را در نظر گرفت. بنابراین صفحات پیشنهادی با استفاده از صفحه p_{j_k} و تعداد کمی از صفحات قبل از آن ایجاد می‌شود. به‌عبارتی‌دیگر پیشنهاددهی بر اساس

$\left\langle p_{j_{k-m+1}}, \ldots, p_{j_{k-2}}, p_{j_{k-1}}, p_{j_k} \right\rangle$ تعیین می‌شود. پنجره نشست S_j با نماد W_j نام این مجموعه است و تعداد اعضای آن یعنی $|W_j|$ معمولاً ۱، ۲ یا ۳ است.

الگوریتم ۲: الگوریتم پیشنهاددهی

Input:
RecNum – The number of recommendations
ReqPage – The requsted page by user
Clusters – A list of clusters mined in offline phase
Output:
RecList – A list of pages recommended
Method:
1. $Create(RecList)$
2. **if** $isNotEmpty(Clusters(ReqPage))$
3. **then while** $size(RecList) \le RecNum$ **and** $hasNext(Clusters(ReqPage))$
4. **do** $Add(RecList, Next(Clusters(ReqPage)))$
5. **return** *RecList*

در الگوریتم پیشنهاددهی فرض می‌شود که کاربر صفحه *ReqPage* را برای مشاهده درخواست کرده است. در بخش برون‌خط برای هر صفحه، مجموعه صفحات مرتبط با آن در مجموعه‌ای بنام *Clusters* مشخص شد. در سطر ۱ الگوریتم پیشنهاددهی مجموعه‌ای خالی بنام *RecList* ایجاد می‌شود تا در ادامه صفحات پیشنهادی در آن قرار بگیرند. در سطر ۲ این الگوریتم تهی بودن خوشه صفحه *ReqPage* بررسی می‌شود. درصورتی‌که $isNotEmpty(Clusters(ReqPage))$ درست نباشد، الگوریتم پیشنهاددهی قادر به پیشنهاددهی به کاربر نیست. مجموعه *RecList* تهی برگردانده می‌شود زیرا صفحه‌ای مرتبط با صفحه *ReqPage* در بخش برون‌خط پیدا نشده است.

اگر $isNotEmpty(Clusters(ReqPage))$ درست باشد آنگاه الگوریتم پیشنهاددهی می‌تواند برای کاربر مجموعه پیشنهادهایی ایجاد کند. مجموعه $Clusters(ReqPage)$ صفحات مرتبط با صفحه درخواستی کاربر یعنی *ReqPage* را دارد. بنابراین مجموعه صفحات پیشنهادی یعنی *RecList* را می‌توان با صفحات موجود در $Clusters(ReqPage)$ پر کرد. البته متغیر *RecNum* تعداد صفحات *RecList* یعنی تعداد صفحات پیشنهادی را مشخص می‌کند. بر روی تعداد صفحات پیشنهادی به کاربر محدودیت وجود دارد. زیرا پیشنهاددهی تعداد زیادی صفحه به کاربر با علت پیدایش سامانه‌های توصیه‌گر تناقض دارد. هدف بر این است که سامانه‌های توصیه‌گر به کاربر در دستیابی به اولویت‌هایشان در میان انبوه اطلاعات کمک کنند. اگر این سامانه‌ها نیز اطلاعات زیادی را به کاربر ارائه دهند آنگاه به کاربر کمکی نخواهد شد. پیشنهادها در سطرهای ۳ و ۴ این الگوریتم مشخص می‌شوند. به‌عبارتی‌دیگر تا زمانی که صفحه‌ای در $Clusters(ReqPage)$ موجود باشد و تعداد صفحاتی پیشنهادی از مقدار مجاز عبور نکرده باشد صفحات $Clusters(ReqPage)$ در *RecList* قرار می‌گیرند.

در انتهای الگوریتم پیشنهاددهی، مجموعه *RecList* به‌عنوان صفحات پیشنهادی به کاربر برمی‌گردانده می‌شود. در صفحه درخواستی کاربر، پیوندهایی به این صفحات پیشنهادی قرار داده می‌شود و به او ارسال می‌شود. در الگوریتم پیشنهاددهی طول پنجره نشست ۱ است؛ به بیانی دیگر برای تعیین پیشنهادها تنها از صفحه درخواستی کاربر استفاده می‌شود؛ بنابراین در بخش برخط نیازی به نگهداری و دنبال نشست‌های کاربری نیست؛ درنتیجه موتور توصیه‌گر با فراموش‌کاری پروتکل HTTP سازگار است.

در الگوریتم پیشنهاددهی برای ایجاد مجموعه پیشنهادها به ازای هر صفحه درخواستی تنها به خوشه آن صفحه مراجعه می‌شود؛ یعنی الگوریتم پیشنهاددهی به تعداد خوشه‌ها، صفحات یا تعداد صفحات مشاهده شده توسط کاربر وابستگی ندارد؛ در حقیقت الگوریتم پیشنهاددهی با پیچیدگی زمانی ثابت برای هر صفحه درخواستی مجموعه پیشنهادها را آماده می‌کند. البته لازم به ذکر است که باید نگاشتی بین صفحات و شماره یا اندیس حاوی خوشه آن‌ها در لیست خوشه‌ها یعنی *Clusters* ایجاد کرد. برای این منظور می‌توان از روش‌هایی مانند درهم‌سازی یا ذخیره صفحات در پایگاه داده و انتساب یک عدد به آن‌ها استفاده کرد.

۳-۵- پیچیدگی سامانه

در الگوریتم خوشه‌بندی پیشنهادی مرحله غالب استفاده از الگوریتم فلوید-وارشال است. پیچیدگی زمانی الگوریتم فلوید-وارشال و درنتیجه الگوریتم خوشه‌بندی مرتبه درجه ۳ از تعداد صفحات N یعنی $\theta(N^3)$ است. به همین ترتیب پیچیدگی مکانی الگوریتم خوشه‌بندی مرتبه درجه ۲ از تعداد صفحات یعنی $\theta(N^2)$ است. لازم به یادآوری و تأکید است که الگوریتم خوشه‌بندی به‌صورت برون‌خط انجام می‌شود.

موتور توصیه‌گر در بخش برخط قرار دارد. عملیات پیشنهاددهی نباید باعث ایجاد تأخیر شود، نسبت به حالتی که به کاربر پیشنهادی داده نمی‌شود. بنابراین پیچیدگی زمانی الگوریتم پیشنهاددهی بسیار مهم است. در سامانه پینو مجموعه صفحات پیشنهادی به ازای هر صفحه به‌صورت برون‌خط مشخص می‌شود. این صفحات در زمان برخط به کاربران پیشنهاد می‌شوند. طبق توضیحات الگوریتم پیشنهاددهی، پیچیدگی زمانی این الگوریتم و موتور توصیه‌گر در زمان برخط ثابت و $O(1)$ است؛ در حقیقت الگوریتم پیشنهاددهی وابستگی به تعداد خوشه‌ها، صفحات یا تعداد صفحات بازدید شده توسط کاربر ندارد. پیچیدگی مکانی الگوریتم پیشنهاددهی مرتبه خطی از تعداد صفحات یعنی $O(N)$ است. زیرا برای هر صفحه تعداد محدودی صفحه در خوشه آن قرار می‌گیرد.

۴- ارزیابی سامانه توصیه‌گر پینو

ارزیابی دقیق سامانه‌های توصیه‌گر به دلیل نیاز به همکاری مستقیم کاربران بسیار دشوار است. بنابراین سامانه‌های توصیه‌گر به‌صورت تقریبی

براساس سیاهه‌ها ارزیابی می‌شوند. اثربخشی سامانه‌های توصیه‌گر را می‌توان از نظر کیفیت پیشنهادها ارزیابی کرد. برای این منظور پارامترهای گوناگونی وجود دارد که در ادامه معرفی می‌شوند.

از سیاهه‌های سرور دانشکده علوم کامپیوتر، مخابرات و سامانه‌های اطلاعاتی در دانشگاه DePaul استفاده شد. دادگان CTI[۷] از تحلیل سیاهه‌های وب سرور این دانشکده در طول ۲ هفته از ماه آوریل سال ۲۰۰۲ به‌دست‌آمده‌اند. این دادگان دارای ۲۰۹۵۰ نشست و ۹۱۵۹ صفحه متمایز است. در این دادگان میانگین طول نشست‌ها ۳ است. به‌عبارتی‌دیگر در هر نشست به‌طور میانگین ۳ صفحه وجود دارد.

دو بخش اصلی سامانه پینو یعنی استخراج الگوهای دسترسی (الگوریتم خوشه‌بندی) و موتور توصیه‌گر (الگوریتم پیشنهاددهی) با زبان برنامه‌نویسی جاوا پیاده‌سازی شد. همه آزمایش‌ها بر روی یک دستگاه رایانه کیفی دارای ۶ گیگابایت حافظه، پردازنده Intel Core i7, 2.4GHz و سیستم عامل ویندوز انجام شد. در آزمایشات، دادگان به دو بخش بنام‌های آموزش و تست تقسیم شد. بخش آموزش برای ساخت مدل و بخش تست برای ارزیابی آن بکار رفت. برای انجام آزمایشات از روش اعتبارسنجی ۱۰-بخشی استفاده شد. در این روش اعتبارسنجی، بخش‌های آموزش و تست در ۱۰ بار با نسبت ۹ به ۱ به ترتیب از دادگان ایجاد می‌شوند. تمامی آزمایش‌ها علاوه بر سامانه پینو بر روی سامانه‌های WebPUM [۲۰]، SUGGEST [۱۸]، HSCR [۲۵] و IHKSCR [۲۵] نیز انجام شد. سپس نتایج حاصل از انجام آزمایش‌ها بر روی این سامانه‌ها با یکدیگر مقایسه شدند.

دادگان CTI قبل از انجام آزمایش‌ها برای کاهش نویز پالایش شد. در این پالایش صفحات تکراری در نشست‌ها، نشست‌های با طول ۱ تا ۳ و صفحات با تکرار بسیار کم در نشست‌ها حذف شد. در حقیقت صفحاتی که نسبت تعداد تکرارهای آن‌ها در نشست‌ها به تعداد نشست‌ها، کمتر از ۰/۰۰۱ است، حذف شدند. در نهایت، دادگان پالایش شده CTI به ۷۹۷۹ نشست و ۴۲۴ صفحه متمایز رسید. میانگین طول نشست‌ها نیز در این دادگان به عدد ۶ رسید.

۴-۱- معیارهای ارزیابی

فرض می‌شود S_j نشست شماره j برای $1 \le j \le \mathrm{N}$ و شیوه دسترسی کاربری دلخواه در بخش تست است. نماد N تعداد نشست‌های بخش تست است. کاربر صفحات $p_{j_1}, p_{j_2}, p_{j_3}, \ldots, p_{j_k}$ را به ترتیب دیده است. به‌عبارتی‌دیگر $S_j = \langle p_{j_1}, p_{j_2}, p_{j_3}, \ldots, p_{j_k} \rangle$ است. برای ارزیابی پیشنهاددهی، نشست S_j به ۲ قسمت تقسیم می‌شود. فرض می‌شود که از صفحه‌ای مانند p_{j_i} که $1 \le i \le k$ است به $S_{j_1} = \langle p_{j_1}, p_{j_2}, p_{j_3}, \ldots, p_{j_i} \rangle$ و $S_{j_2} = \langle p_{j_{i+1}}, p_{j_{i+2}}, p_{j_{i+3}}, \ldots, p_{j_k} \rangle$ تقسیم می‌شود. مجموعه S_{j_2} باقیمانده نشست نامیده می‌شود. الگوریتم پیشنهاددهی مجموعه پیشنهادهایی را با استفاده از تعدادی کمی صفحه از بخش اول ایجاد می‌کند. پنجره نشست S_j، مجموعه شامل این صفحات است که به‌صورت $W_j = \langle p_{j_{i-m+1}}, \ldots, p_{j_{i-2}}, p_{j_{i-1}}, p_{j_i} \rangle$ نشان داده می‌شود. تعداد اعضای پنجره نشست S_j یعنی $|W_j|$ معمولاً ۱، ۲ یا ۳ صفحه است. در سامانه پینو طول این پنجره ۱ است به بیانی دیگر پیشنهاددهی تنها بر اساس ۱ صفحه صورت می‌گیرد.

فرض می‌شود که R_j مجموعه صفحات پیشنهادی برای نشست S_j است. مجموعه R_j شامل صفحات $p_{R_{j_1}}, p_{R_{j_2}}, p_{R_{j_3}}, \ldots, p_{R_{j_r}}$ است. به‌عبارتی‌دیگر $R_j = \langle p_{R_{j_1}}, p_{R_{j_2}}, p_{R_{j_3}}, \ldots, p_{R_{j_r}} \rangle$ است. لازم به ذکر است که تعداد صفحات مجموعه پیشنهادی یعنی $|R_j|$ دارای محدودیت است. با مقایسه مجموعه صفحات پیشنهادی برای نشست S_j یعنی R_j و باقیمانده نشست S_j یعنی S_{j_2} می‌توان معیارهایی را برای ارزیابی کیفیت مجموعه پیشنهادها در نظر گرفت.

یکی از مسائل مهم در اثربخشی سامانه‌های توصیه‌گر، قابلیت پیشنهاددهی[۸] آن‌ها یا میزان توانایی آن‌ها در ارائه مجموعه پیشنهادها است [۱۶]. معیار قابلیت پیشنهاددهی مشخص می‌کند که سامانه می‌تواند به چند درصد از کاربران، مجموعه‌ای را پیشنهاد دهد. عدم توانایی سامانه در پیشنهاددهی برای یک درخواست می‌تواند ناشی از عدم وجود الگوی دسترسی برای آن درخواست در سامانه باشد. قابلیت پیشنهاددهی از رابطه زیر به دست می‌آید:

$$App = \frac{M}{N} \qquad (۳)$$

در رابطه (۳) نماد N بیانگر تعداد نشست‌های بخش تست است. نماد M بیانگر تعداد نشست‌هایی مانند S_j در بخش تست است که مجموعه پیشنهادهایی را دریافت کرده‌اند یعنی مجموعه R_j آن‌ها تهی نیست. نماد App بیانگر قابلیت پیشنهاددهی سامانه است. به بیانی دیگر معیار قابلیت پیشنهاددهی نسبت تعداد نشست‌هایی از بخش تست که پیشنهاد دریافت کرده‌اند، به تعداد کل نشست‌های بخش تست است.

در ارزیابی پیشنهادها به کاربران حقیقی دسترسی وجود ندارد تا به آن‌ها مجموعه‌ای از صفحات پیشنهاد داده شود و بر اساس کلیک‌های آن‌ها درستی مجموعه پیشنهادها بررسی شود. بنابراین درستی مجموعه پیشنهادها با در نظر گرفتن نشست‌هایی مشخص می‌شود که حداقل یک صفحه از مجموعه پیشنهادها را در ادامه خود دارند. به بیانی دیگر یک مجموعه از صفحات پیشنهادی مانند R_j زمانی درست نامیده می‌شود که صفحه‌ای از صفحات آتی مشاهده شده توسط کاربر را دارا باشد. در حقیقت مجموعه پیشنهادهای R_j زمانی درست است که اشتراک مجموعه R_j و مجموعه S_{j_2} یعنی $R_j \cap S_{j_2}$ تهی نباشد. درصد پیشنهادهای درست[۹] مشخص می‌کند که کاربر با چه احتمالی بر روی مجموعه پیشنهادها کلیک می‌کند [۱۶]. این درصد از رابطه زیر به دست می‌آید:

$$Cor = \frac{C}{M} \qquad (۴)$$

در رابطه (۴) نماد M تعداد نشست‌هایی در بخش تست است که مجموعه پیشنهادهایی را دریافت کرده‌اند. نماد C بیانگر تعداد

نشست‌هایی مانند S_j در بخش تست است که اشتراک بین مجموعه پیشنهادی آن‌ها یعنی مجموعه R_j و مجموعه S_{j_2} تهی نیست. نماد Cor بیانگر درصد پیشنهادهای درست سامانه است. به بیانی دیگر درصد پیشنهادهای درست نسبت تعداد نشست‌هایی از بخش تست که پیشنهاد درست دریافت کرده‌اند به تعداد کل نشست‌های بخش تست است.

بین معیارهای App و Cor مصالحه وجود دارد. حالت مطلوب زمانی است که هر دو معیار بیشینه باشند. بنابراین میانگین همساز بین معیارهای App و Cor محاسبه می‌شود. میانگین همساز معیارهای App و Cor در حقیقت همان f-measure برای این معیارها است. میانگین همساز اعداد حقیقی و مثبت $x_1,x_2,x_3,\dots,x_n$ از رابطه زیر به دست می‌آید:

$$HM\left(x_1,x_2,x_3,\dots,x_n\right)=\frac{n}{\sum_{i=1}^{n}\frac{1}{x_i}} \qquad (۵)$$

در رابطه (۵) نماد $HM\left(x_1,x_2,x_3,\dots,x_n\right)$ میانگین همساز بین اعداد گفته شده را نشان می‌دهد.

معیارهای دقت[۱۰] و پوشش[۱۱] در سامانه‌های توصیه‌گر برای بررسی بیشتر مجموعه پیشنهادهای درست تعریف می‌شوند. دقت سامانه توصیه‌گر در ارائه یک مجموعه پیشنهاد را می‌توان از رابطه زیر به دست آورد:

$$Acc\left(R_j,S_j\right)=\frac{\left|R_j\cap S_{j_2}\right|}{\left|R_j\right|} \qquad (۶)$$

در رابطه (۶)، $Acc\left(R_j,S_j\right)$ دقت سامانه توصیه‌گر را در ارائه مجموعه پیشنهاد R_j برای نشست S_j در بخش تست نشان می‌دهد. در این رابطه، $\left|R_j\cap S_{j_2}\right|$ تعداد صفحات مشترک بین مجموعه صفحات پیشنهادی R_j و باقیمانده نشست S_j یعنی S_{j_2} را نشان می‌دهد. $\left|R_j\right|$ تعداد صفحات مجموعه پیشنهادی R_j را بیان می‌کند. میانگین $Acc\left(R_j,S_j\right)$ برای همه نشست‌های بخش تست یعنی به ازای $1\le j\le \mathrm{N}$ دقت سامانه توصیه‌گر را نشان می‌دهد. این مقدار مشخص می‌کند که سامانه توصیه‌گر به چه میزان پیشنهادهای دقیق می‌آورد [۲۰]. پوشش سامانه توصیه‌گر در ارائه یک مجموعه پیشنهاد را می‌توان از رابطه زیر به دست آورد:

$$Cov\left(R_j,S_j\right)=\frac{\left|R_j\cap S_{j_2}\right|}{\left|S_{j_2}\right|} \qquad (۷)$$

در رابطه (۷)، $Cov\left(R_j,S_j\right)$ پوشش سامانه توصیه‌گر را در ارائه مجموعه پیشنهاد R_j برای نشست S_j در بخش تست نشان می‌دهد. در این رابطه، $\left|R_j\cap S_{j_2}\right|$ تعداد صفحات مشترک بین مجموعه صفحات پیشنهادی R_j و باقیمانده نشست S_j یعنی S_{j_2} را نشان می‌دهد. $\left|S_{j_2}\right|$ تعداد صفحات باقیمانده نشست S_j است. میانگین $Cov\left(R_j,S_j\right)$ برای همه نشست‌های بخش تست یعنی به ازای $1\le j\le \mathrm{N}$ پوشش سامانه توصیه‌گر را نشان می‌دهد. این مقدار مشخص می‌کند که سامانه توصیه‌گر به چه میزان در ارائه صفحات مشاهده شده توسط کاربر موفق است [۲۰].

بین معیارهای Acc و Cov مصالحه وجود دارد [۱۲]. حالت مطلوب زمانی است که هر دو معیار بیشینه باشند. بنابراین میانگین همساز بین معیارهای Acc و Cov طبق رابطه (۵) محاسبه می‌شود. میانگین همساز معیارهای Acc و Cov یعنی $HM\left(Acc,Cov\right)$ در حقیقت همان f-measure برای این معیارها است.

عملکرد سامانه‌های توصیه‌گر با معیارهای قابلیت پیشنهاددهی، پیشنهادهای درست، دقت و پوشش پیشنهاددهی توصیف شد. حالت مطلوب زمانی است که همه این معیارها بیشینه باشند. علاوه بر این زمانی که یک پدیده با چند معیار عددی بیان می‌شود، برای بیان آن پدیده به‌صورت یک عدد می‌توان از میانگین همساز بین معیارها استفاده کرد. بنابراین اثربخشی سامانه توصیه‌گر را می‌توان میانگین همساز معیارهای تعریف شده در نظر گرفت یعنی اثربخشی سامانه توصیه‌گر $HM\left(App,Cor,Acc,Cov\right)$ است.

۴-۲- نتایج ارزیابی

در ادامه نمودارهایی آورده شده‌اند که سامانه توصیه‌گر پینو را با سامانه‌های WebPUM [۲۰]، SUGGEST [۱۸]، HSCR [۲۵] و IHKSCR [۲۵] بر اساس معیارهای بیان شده مقایسه می‌کنند. در این نمودارها، محور افقی بیانگر محدودیتی است که در این سامانه‌ها اعمال می‌شود. در سامانه‌های پینو، WebPUM و SUGGEST این محدودیت بر روی وزن یال‌ها در گراف شیوه‌های دسترسی اعمال می‌شود. در سامانه پینو متغیر *MaxDist* و در سامانه WebPUM و SUGGEST متغیر *MinFreq* محدودیت بر روی وزن یال‌ها را نشان می‌دهد. در سامانه‌های HSCR و IHKSCR محدودیتی بر روی ابعاد بردارها هنگام محاسبه فاصله بین آن‌ها قرار داده می‌شود که این محدودیت در این دو سامانه با متغیر *MinFreq* نشان داده می‌شود. در تمامی این نمودارها مقدار *MaxDist* و *MinFreq* بین ۰ و ۱ تغییر می‌کند. نرمال‌سازی بر روی وزن یال‌ها در سامانه پینو صورت گرفته است تا بخشی از آن‌ها در بازه بین ۰ تا ۱ قرار بگیرند.

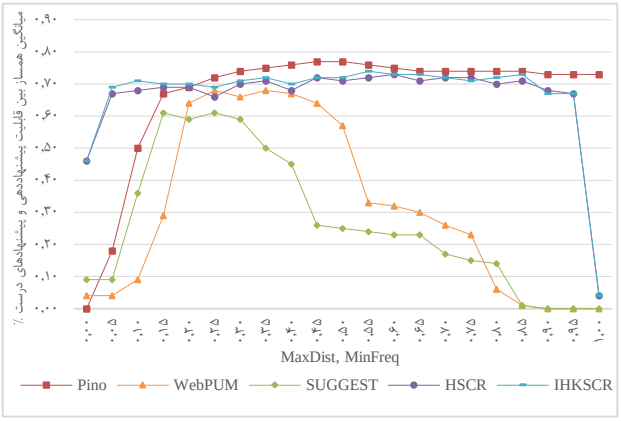
همان‌گونه که بیان شد در مجموعه داده‌ها میانگین طول نشست‌ها ۶ است. بنابراین برای پیشنهاددهی نقطه میانی نشست‌ها در نظر گرفته شده است تا تعداد صفحه کافی برای تعیین مجموعه پیشنهادها و ارزیابی آن‌ها در سامانه‌های مورد نظر وجود داشته باشد. به بیانی دیگر بر اساس تنظیمات هر یک از سامانه‌ها به‌طور میانگین می‌توان بر اساس ۳ صفحه ابتدایی هر نشست مجموعه پیشنهادها را تعیین و بر اساس ۳ صفحه انتهایی نشست آن‌ها را ارزیابی کرد. بنابراین در این سامانه‌ها حداکثر ۵ صفحه به کاربر برای برآورده کردن نیازهای احتمالی او پیشنهاد داده می‌شود.

در سامانه پینو با توجه به محدودیت بر روی تعداد صفحات پیشنهادی به کاربر یعنی حداکثر ۵ صفحه، تعداد اعضای هر خوشه نیز حداکثر به این تعداد محدود شده است. همچنین یادآوری می‌شود که طول پنجره نشست در سامانه پینو ۱ است. متغیرهای لازم برای سامانه‌های دیگر متناسب با توضیحات مقاله‌های مربوط به خود به‌گونه‌ای مقداردهی شدند تا به بالاترین کارایی ممکن خود برسند.

نمودار شکل ۲ میانگین همساز یا f-measure بین معیارهای قابلیت پیشنهاددهی و درصد پیشنهادهای درست را به ازای مقادیر مختلف محدودیت در سامانه‌های توصیه‌گر مورد بررسی نشان می‌دهد.

نمودار شکل ۳ میانگین همساز یا f-measure بین معیارهای دقت و پوشش پیشنهاددهی را به ازای مقادیر مختلف محدودیت در سامانه‌های توصیه‌گر مورد نظر نشان می‌دهد.

نمودار شکل ۴ اثربخشی پیشنهاددهی یا همان میانگین همساز بین معیارهای قابلیت پیشنهاددهی، پیشنهادهای درست، دقت و پوشش را به ازای مقادیر مختلف محدودیت در سامانه‌های توصیه‌گر مورد ارزیابی نشان می‌دهد.

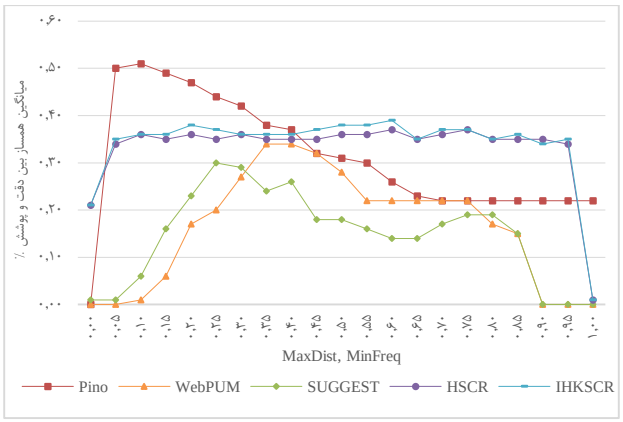


شکل ۲: نمودار میانگین همساز بین قابلیت پیشنهاددهی و پیشنهادهای درست

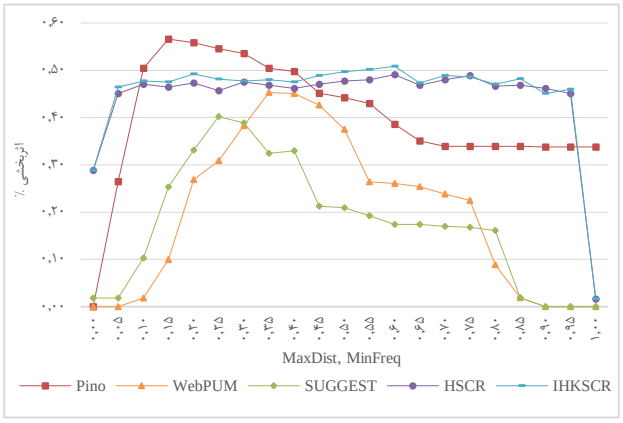
جدول ۱ بهترین عملکرد سامانه‌های توصیه‌گر مورد بررسی را در میانگین‌های همساز بین معیارهای ارزیابی نشان می‌دهد. طبق این جدول مشخص است که سامانه پینو از نظر معیارهای قابلیت پیشنهاددهی و پیشنهادهای درست، معیارهای دقت و پوشش و معیار اثربخشی پیشنهاددهی بهترین عملکرد را در بین این سامانه‌های توصیه‌گر دارد.

۵- نتیجه

در این مقاله، یک سامانه توصیه‌گر بنام پینو مبتنی بر کاوش در سیاهه‌های وب ارائه شده است. رویکرد جدیدی برای استخراج الگوهای دسترسی در این سامانه معرفی شده است. برای این منظور شیوه‌های دسترسی کاربران در یک گراف جهت‌دار و وزن‌دار مدل می‌شوند.



شکل ۳: نمودار میانگین همساز بین دقت و پوشش پیشنهاددهی



شکل ۴: نمودار اثربخشی پیشنهاددهی

این گراف بیانگر صفحات و رابطه بین آن‌ها است که وزن یال‌های این گراف با استفاده از معکوس احتمال شرطی مشاهده صفحات تخصیص می‌یابد. استخراج الگوهای دسترسی با خوشه‌بندی صفحات یعنی بخش‌بندی این گراف بر پایه کوتاه‌ترین مسیرها صورت می‌گیرد.

الگوهای دسترسی کشف شده برای پیشنهاددهی به کاربران با توجه به درخواست‌های آن‌ها در زمان برخط مورد استفاده قرار می‌گیرند. در حقیقت هنگام بازدید کاربران این سامانه به‌صورت سازگار با فراموش‌کاری پروتکل HTTP، مجموعه پیشنهادهایی را به آن‌ها با پیچیدگی زمانی ثابت ارائه می‌دهد. این سامانه با استفاده از سیاهه‌های یک سرور مورد ارزیابی قرار گرفته است. اثربخشی سامانه توصیه‌گر با معیارهای قابلیت پیشنهاددهی، پیشنهادهای درست، دقت و پوشش پیشنهاددهی ارزیابی شده است. نتایج ارزیابی نشان می‌دهد که این سامانه می‌تواند کیفیت پیشنهاددهی را بهبود دهد.

مواردی به‌عنوان کارهای پیش‌رو پیشنهاد می‌شود. نخست، در خوشه‌بندی تأثیر راهکارهای دیگری بررسی شود. این امکان وجود دارد که درجه عضویت صفحات در خوشه‌ها بر اساس راه‌های دیگری محاسبه شود. به‌عنوان مثال میانگین وزن‌دار مسیر برای آن در نظر گرفته شود.

جدول ۱: بهترین عملکرد سامانه‌های توصیه‌گر در میانگین‌های همساز بین معیارهای ارزیابی

	$HM(App, Cor)$	$HM(Acc, Cov)$	$HM(App, Cor, Acc, Cov)$
Pino	٪۷۷	٪۵۱	٪۵۷
IHKSCR	٪۷۴	٪۳۹	٪۵۱
HSCR	٪۷۳	٪۳۷	٪۴۹
WebPUM	٪۶۸	٪۳۴	٪۴۵
SUGGEST	٪۶۱	٪۳۰	٪۴۰

همچنین امکان‌پذیر است که محدودیت به‌جای تعداد اعضای خوشه بر روی درجه عضویت آن‌ها قرار داده شود. دوم، رویکردهای دیگری برای پیش‌بینی درخواست‌های آتی کاربران در نظر گرفته شود. به‌عنوان مثال قابلیت به‌کارگیری توزیع نمایی یا دیگر الگوریتم‌های گراف برای این منظور بررسی شود. سوم، برای ترکیب بخش‌های برون‌خط و برخط در این سامانه تلاش شود تا همواره مدل پیشنهاددهی با دسترسی‌های جاری موجود در وب سرور هماهنگ باشد. البته نباید ترکیب این دو بخش بر روی عملکرد سامانه توصیه‌گر مانند زمان ارائه پیشنهاد تأثیر منفی بسیار زیادی بگذارد.

مراجع

[1] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic Personalization Based on Web Usage Mining," *Commun. ACM*, vol. 43, no. 8, pp. 142–151, Aug. 2000.

[2] J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," *ACM SIGKDD Explorations Newsletter*, vol. 1, no. 2, pp. 12–23, Jan. 2000.

[3] R. Cooley, B. Mobasher, and J. Srivastava, "Data Preparation for Mining World Wide Web Browsing Patterns," *Knowledge and Information Systems*, vol. 1, no. 1, pp. 5–32, 1999.

[4] T. W. Yan, M. Jacobsen, H. Garcia-Molina, and U. Dayal, "From user access patterns to dynamic hypertext linking," *Computer Networks and ISDN Systems*, vol. 28, no. 7, pp. 1007–1014, 1996.

[5] J. A. Hartigan, *Clustering Algorithms*, 99th ed. New York, NY, USA: John Wiley & Sons, Inc., 1975.

[6] R. C. Agarwal, C. C. Aggarwal, and V. V. V Prasad, "A Tree Projection Algorithm for Generation of Frequent Item Sets," *Journal of Parallel and Distributed Computing*, vol. 61, no. 3, pp. 350–371, 2001.

[7] E.-H. Han, G. Karypis, V. Kumar, and B. Mobasher, "Clustering based on association rule hypergraphs," in *Proccedings of SIGMOD'97 Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD'97)*, 1997.

[8] B. Mobasher, R. Cooley, and J. Srivastava, "Creating adaptive Web sites through usage-based clustering of URLs," in *Proceedings of Workshop on Knowledge and Data Engineering Exchange*, 1999, pp. 19–25.

[9] M. Perkowitz and O. Etzioni, "Towards adaptive Web sites: Conceptual framework and case study," *Artificial Intelligence*, vol. 118, no. 1, pp. 245–275, 2000.

[10] C. Shahabi, F. Banaei-Kashani, Y.-S. Chen, and D. McLeod, "Yoda: An Accurate and Scalable Web-Based Recommendation System," in *Proceedings of the 9th International Conference on Cooperative Information Systems*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 418–432.

[11] C. Shahabi, F. Banaei-Kashani, J. Faruque, and A. Faisal, "Feature Matrices: A Model for Efficient and Anonymous Web Usage Mining," in *Proceedings of the Second International Conference on Electronic Commerce and Web Technologies*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 280–294.

[12] M. Nakagawa and B. Mobasher, "A hybrid web personalization model based on site connectivity," in *Proceedings of WebKDD*, 2003, pp. 59–70.

[13] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, 1994, pp. 487–499.

[14] D. E. Knuth, *The Art of Computer Programming : Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1997.

[15] M. Nakagawa and B. Mobasher, "Impact of Site Characteristics on Recommendation Models Based On Association Rules and Sequential Patterns," in *Proceedings of the IJCAI'03 Workshop on Intelligent Techniques for Web Personalization*, 2003.

[16] B. Zhou, S. C. Hui, and K. Chang, "An intelligent recommender system using sequential Web access patterns," in *IEEE Conference on Cybernetics and Intelligent Systems*, 2004, vol. 1, pp. 393–398.

[17] B. Zhou, S. C. Hui, and A. C. M. Fong, "CS-Mine: An Efficient WAP-Tree Mining for Web Access Patterns," in *Proceedings of the 6th Asia-Pacific Web Conference*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 523–532.

[18] R. Baraglia and F. Silvestri, "Dynamic Personalization of Web Sites Without User Intervention," *Commun. ACM*, vol. 50, no. 2, pp. 63–67, Feb. 2007.

[19] A. Silberschatz, P. B. Galvin, and G. Gagne, *Operating System Concepts*, 8th ed. Wiley Publishing, 2008.

[20] M. Jalali, N. Mustapha, M. N. Sulaiman, and A. Mamat, "WebPUM: A Web-based recommendation system to predict user future movements," *Expert Systems with Applications*, vol. 37, no. 9, pp. 6201–6212, 2010.

[21] D. S. Hirschberg, "Algorithms for the Longest Common Subsequence Problem," *J. ACM*, vol. 24, no. 4, pp. 664–675, Oct. 1977.

[22] R. Forsati, H. M. Doustdar, M. Shamsfard, A. Keikha, and M. R. Meybodi, "A fuzzy co-clustering approach for hybrid recommender systems," *International Journal of Hybrid Intelligent Systems*, vol. 10, no. 2, pp. 71–81, 2013.

[۲۳] سیامک عبداله‌زاده، محمدعلی بالافر و لیلی محمدخانی، «استفاده از خوشه‌بندی و مدل مارکوف جهت پیش‌بینی درخواست آتی کاربر در وب»، مجله مهندسی برق دانشگاه تبریز، جلد ۴۵، شماره ۳، صفحات ۹۶-۸۹، ۱۳۹۴.

[۲۴] جواد حمیدزاده و مریم صادق‌زاده، «فیلترکننده مشارکتی فازی ناهموار مبتنی بر کاربر در سیستم‌های پیشنهاددهنده»، *مجله مهندسی برق دانشگاه تبریز، جلد ۴۷، شماره ۲، صفحات ۵۰۰-۴۹۱، ۱۳۹۶.*

[25] R. Forsati, A. Moayedikia, and M. Shamsfard, "An effective Web page recommender using binary data clustering," *Information Retrieval Journal*, vol. 18, no. 3, pp. 167–214, 2015.
[26] H. Liu and V. Kešelj, "Combined mining of Web server logs and web contents for classifying user navigation patterns and predicting users' future requests," *Data & Knowledge Engineering*, vol. 61, no. 2, pp. 304–330, 2007.
[27] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*, 2nd ed. McGraw-Hill Higher Education, 2001.

زیرنویس‌ها

۱ Web usage mining
۲ Logs
۳ Hypergraph
۴ Hyperedge
۵ Harmony search
۶ visit-coherence assumption
۷ http://facweb.cs.depaul.edu/mobasher/classes/ect584/data/cti-data.zip
۸ Applicability Recommendation
۹ Correct Recommendations
۱۰ Accuracy
۱۱ Coverage