ارائه روشی جدید برای شخصی سازی صفحات وب با استفاده از آتاماتای یادگیر و آیتم ست های تکرار شونده وزن دار

رعنا فرصتی
گروه مهندسی کامپیوتر، دانشکده فنی، دانشگاه آزاد اسلامی، واحد کرج
کرج، ایران
forsati@kiau.ac.ir

محمد رضا میبدی
دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر
تهران، ایران
mmeybodi@aut.ac.ir

چکیده

شخصی سازی وب مجموعه‌ای از عملیات است که تجربه وب را برای یک کاربر خاص یا مجموعه‌ای از کاربران سازمان‌دهی می‌کند و پیشنهادات پویا بر اساس الگوهای رفتاری کاربران ارائه می‌دهد. در این مقاله الگوریتم جدیدی معرفی شده است که با انتساب وزن به آیتم‌های موجود در تراکنش‌ها با استفاده از اطلاعات پیمایش کاربران گونه جدیدی از آیتمهای تکرار شونده تحت عنوان "آیتم ست های تکرار شونده وزن دار "را استخراج می‌کند. الگوریتم پیشنهاد شده است از آتاماتای یادگیر توزیع شده، پیوند بین صفحات سایت، و آیتم ست های وزن‌دار استخراج شده به منظور پیشنهاد صفحات به کاربران استفاده می‌کند. الگوریتم ارائه شده مشکلات موجود در روش‌های پیشنهادی بر اساس آتاماتای یادگیر و قوانین انجمنی، مشکل صفحات جدیدی که اخیرا به سایت اضافه شده اند و کاهش دقت الگوریتمها با افزایش تعداد صفحات پیشنهادی را به نحو مطلوبی حل می‌کند.. نحوه کار الگوریتم به این صورت است که اولین صفحه را با استفاده از آیتم ست‌های تکرارشونده وزن‌دار معرفی شده و آتاماتای یادگیر پیشنهاد می‌کند. سپس این صفحه با استفاده از الگوریتم HITS و صفحاتی که با آن در یک دسته بندی هستند بسط داده می شود تا صفحاتی که اخیرا به سایت اضافه شده اند نیز فرصت حضور در مجموعه صفحات پیشنهادی را داشته باشند. نتایج شبیه‌سازی الگوریتم در داده‌های واقعی نشان داده است که کارایی الگوریتم پیشنهادی بالا می‌باشد و دانش بدست آمده از سیستم مذکور به طور قابل ملاحظه ای کیفیت پیشنهادات را بهبود داده است و مشکلات ذکر شده را در حد قابل توجهی کاهش داده است.

واژه های کلیدی: شخصی سازی صفحات وب، آتاماتای یادگیر

۱- مقدمه

وب به مجموعه بزرگی از داده های ساخت یافته و نیمه ساخت یافته تبدیل شده است که کاربران آن از همپوشانی داده‌ها رنج می‌برند. بنابراین تحلیل رفتارهای کاوشی کاربران وب و بررسی واقعی علایق کاربران اهمیت خاصی پیدا کرده است. بررسی رفتارهای کاربران در وب، به عنوان روشی جهت کشف دانش نهفته در نحوه تعامل کاربران با وب، یکی از ابزارهای مهم در حوزه کاوش در وب شناخته می‌شود. کارهای تحقیقاتی بسیاری در این حوزه انجام شده است که عمدتا بر مبنای اطلاعات موجود از رفتار کاربر در تعامل با وب به استخراج این دانش و استفاده از آن در کاربردهای مختلف در وب نظیر شخصی‌سازی صفحات وب و پیشنهاد صفحات[۶,۷]، تعیین ارتباط بین اسناد [۸]، خود سازمانده کردن وب[۹] می‌پردازند.

در ارزیابی به عمل آمده از روشهای مبتنی بر اتاماتای یادگیر [۳] و قوانین انجمنی [۱۱] ، مواردی مشاهده می‌شود که متناظر با دنباله صفحات بازدید شده، سیستم در حالتی قرار گرفته که قبلا هرگز مشاهده نشده است. این مسئله، حتی در حالت استفاده از پنجره هایی با طول مختلف نیز وجود دارد. در این شرایط، سیستم در ارائه پیشنهاد با مشکل روبرو خواهد بود. در واقع، این مسئله معادل این نکته است که کاربر دنباله یا زیر دنباله ای از صفحات را بازدید کند که پیش از آن، در مرحله آموزش، هر گز توسط سیستم مشاهده نشده بوده است. این مسئله می تواند بر اثر مشکل صفحه جدید که مشکل شناخته شده‌ای می باشد روی دهد. به این معنی که با اضافه شدن صفحات جدید به دامنهٔ اقلام، به علت در دست نبودن اطلاعات کاربرد، سیستم پیشنهاد دهنده اطلاعاتی در مورد این اقلام نخواهد داشت. در نتیجه در این حالت، سیستم، قادر به ارائه پیشنهاد اقلام جدید نیست. با بررسی نتایج آزمایشات انجام شده، مشخص می شود که پوشش پیشنهادهای سیستم بر روی مجموعه صفحات وب سایت (درصد صفحاتی که حداقل یک بار پیشنهاد شده اند)، بطور متوسط در حدی پایین است. دلیل این مسئله نیز، مشابه مسئله قبلی،" عدم رخداد "یا" رخداد با تناوب کم" مجموعه ای از صفحات در مجموعه اطلاعات کاربرد است. در این شرایط، سیستم شواهدی برصحت پیشنهاد صفحات بازدید شده ندارد. در این مقاله ابتدا الگوریتمی برای استخراج آیتم ست های تکرار شونده وزن دار پیشنهاد می کنیم. سپس با استفاده از آتاماتای یادگیر توزیع شده و آیتم ست های تکرار شونده وزن دار معرفی شده، الگوریتم جدیدی برای پیشنهاد صفحات وب به کاربران پیشنهاد کرده ایم. در نهایت از اطلاعات ساختار، جهت غنی کردن اطلاعات کاربرد و الگوهای استخراج شده از این اطلاعات استفاده کرده ایم. الگوریتم ارائه شده با بسط و توسعه الگوها با توجه به ساختار سایت، مشکلات اشاره شده در بالا را در حد قابل توجهی حل می‌کند. به دلیل اینکه طراحان صفحات وب از یک صفحه به صفحات دیگر زمانی پیوند قرار می‌دهند که عنوان و محتوای صفحات مذکور در راستای محتوای آن صفحه وب باشند، بنابراین

داده‌های ساختار حاوی اطلاعات ضمنی با ارزشی هستند و استفاده از این اطلاعات دانش زیادی راجع به این صفحات و ارتباطشان به دست می‌دهد که دقت الگوریتم ارائه‌شده را تا حد زیادی بالا می‌برد.

الگوریتم ارائه شده بر روی داده‌های واقعی شبیه‌سازی شده و نتایج نشان می‌دهد که دانش بدست آمده از سیستم مذکور به طور قابل ملاحظه ای کیفیت پیشنهادات را بهبود داده است. در ادامه در بخش ۲ الگوریتم پیشنهادی ارائه می‌گردد. در بخش۳ پس از معرفی مدل استفاده شده برای شبیه‌سازی، نتایج شبیه‌سازی ارائه می‌شود. در پایان نیز نتیجه‌گیری آورده شده‌است.

۲- روش پیشنهادی

هدف از سیستم های شخصی سازی محاسبه یک مجموعه پیشنهادی، rs، برای نشست کاربر جاری می‌باشد که بیشترین تطابق را با علایق کاربر داشته باشد. این جز تنها جز برخط سیستم بوده و باید از کارایی و دقت بالایی برخوردار باشد. یکی از مشکلات اصلی الگوریتم پیشنهاد صفحات بر اساس اتاماتای یادگیر [۳] محاسبه امتیاز برای همه مجموعه صفحات مشاهده نشده توسط کاربر است که زمان‌بر بوده و کارایی الگوریتم را محدود می‌کند لذا با استفاده از الگوریتم معرفی شده مبتنی بر الگوی آیتم‌های تکرار شونده، مجموعه صفحات بالقوه پیشنهادی را به صورت هوشمندانه‌تری انتخاب می‌کنیم.

در الگوریتم پیشنهاد صفحات با استفاده از اتاماتای یادگیر برای پیشنهاد صفحه به کاربر جاری که بصورت برخط انجام می‌شود پس از دریافت نشست جاری کاربر امتیاز پیشنهاد صفحه P_{k+1} به کاربر از رابطه زیر محاسبه می‌شود [۳]:

$$\Pr(p_1 \to p_2 \to p_3 \ldots \to p_k) = \Pr(p_1) \times \prod_{i=2}^{k} \Pr(p_i |\ p_{i-m} \ldots p_{i-1}) \qquad (۱)$$

به عنوان مثال، احتمال مسیر $p_1 \to p_2 \to p_3$ برابراست با:

$$\Pr(p_1 \to p_2 \to p_3) = \Pr(p_1)\Pr(p_2 \mid p_1)\Pr(p_3 \mid p_2) = \Pr(p_1)\frac{\Pr(p_1 \to p_2)\Pr(p_2 \to p_3)}{\Pr(p_1)\Pr(p_2)} \qquad (۲)$$

که در آن $Pr(\bullet \to \bullet)$ برابر با احتمال گذار بین دو صفحه است و $Pr(\bullet)$ احتمال حالت پایدار صفحه متناظر می‌باشدکه با استفاده از آتاماتای یادگیر محاسبه شده‌اند. در این الگوریتم برای پیشنهاد صفحه به کاربر، به ازای صفحات مختلف P_{k+1} که در مسیر $p_1 \to p_2 \to p_3 \to \ldots \to p_k$ ملافات نشده‌اند، احتمال مسیر $p_1 \to p_2 \to p_3 \to \ldots \to p_k \to p_{k+1}$ محاسبه می‌شود [۳]. احتمال هر مسیر امتیاز صفحه P_{k+1} را برای پیشنهاد به کاربر نشان می‌دهد. اساسا این رهیافت برای هریک از صفحات p_i ، احتمال دسترسی به آن صفحه را در مرحله بعد مشخص می‌کند و پس از آن صفحه‌ای که بالاترین احتمال را دارد، انتخاب می‌کند. مرحله اصلی در تعیین p_{i+1} در رابطه بالا توانایی محاسبه احتمالات شرطی مختلفی است که توسط نشست وب کاربر مشخص شده است. در حالت کلی، محاسبه و تعیین این احتمالات شرطی در مواردی ممکن است مقدور نباشد، زیرا: اولا: نشست‌های وب می‌توانند به اندازه دلخواه بزرگ باشند.. ثانیا: اندازه مجموعه آموزشی، معمولا خیلی کوچکتر از اندازه مورد نیاز برای تخمین احتمالات شرطی مختلفی است که در نشست‌های طولانی وب وجود دارند.

در این بخش با در نظر گرفتن پارامتر وزن در کنار هر آیتم، الگوریتم جدیدی برای استخراج آیتم‌ست‌های تکرار شونده وزن‌دارمعرفی کرده ایم. در این مدل، وزن‌های بزرگتر بیانگر آیتم های مهمتری می باشند که این مساله استخراج آیتم های مهمتر با تعداد تکرار کمتر را فراهم می کند. در روش پیشنهادی به منظور تسهیل فرایند جستجوی مجموعه صفحات کاندید برای پیشنهاد و بهبود زمان لازم برای ارائه پیشنهاد با استفاده از آیتم ست های تکرارشونده وزن‌دار، از ساختمان داده جدیدی برای ذخیره اقلام مکرر وزن دار استفاده می‌کنیم. ساختمان داده معرفی شده، *گراف اقلام مکرر وزن دار* نامیده شده است. این ساختمان داده مجموعه صفحات کاندیدی که برای ارائه پیشنهاد باید بررسی شوند و امتیاز پیشنهاد برای آنها باید محاسبه شود را محدود می کند و به این ترتیب زمان تولید یک مجموعه پیشنهادی که تنها جر بر خط سیستم است را تا حد زیادی کاهش می دهد.

الگوریتم پیشنهاد شده از مراحل زیر تشکیل شده است:

۱- تولید مجموعه پایه بر اساس اتاماتای یادگیر و آیتم ست های تکرار شونده وزن دار معرفی شده

۲- خوشه بندی صفحات وب بر اساس کاوش کاربرد وب

۳- توسعه مجموعه پایه با استفاده از نتایج خوشه بندی و تولید مجموعه کاندید

۴- استفاده از الگوریتم HITS به منظور رتبه بندی صفحات و تولید پیشنهادات نهایی

۲-۱ ایجاد مجموعه پایه پیشنهاد صفحات

روند ایجاد مجموعه پایه پیشنهاد صفحات به شرح زیر است: فرض کنیم که P مجموعه صفحات قابل دسترسی توسط کاربران یک سایت باشد $P = \{p_1, p_2, \ldots, p_m\}$، همچنین T مجموعه تراکنش‌های کاربران در فایل پیش پردازش شده ثبت وقایع باشد $T = \{t_1, t_2, \ldots, t_n\}$ که در آن تراکنش $t_i \in T$ زیر مجموعه ای از صفحات P می باشد. هر تراکنش t_i را بصورت بردار m تایی از صفحات مدل می‌کنیم $t_i = \{(p_1, w_1), (p_2, w_2), \ldots, (p_m, w_m)\}$ که w_i وزن صفحه p_i درتراکنش t_i می باشد که بر اساس معیار معرفی شده در[۱۰] و بر اساس دو پارامتر مدت زمان مشاهده صفحه و فرکانس مشاهده محاسبه

می شود. برای توصیف الگوریتم پیشنهاد شده برای استخراج آیتم ست های تکرار شونده وزن دار تعاریف زیر ارائه می کنیم.

تعریف ۱: وزن اقلام[1] در هر تراکنش

وزن اقلام بر اساس وزن اعضایی(صفحاتی) $w(p_i)$ که شامل آنها است تعیین می گردد و با $w(X,t)$ نمایش داده می شود. ساده ترین راه برای بدست آوردن وزن اقلام در نظر گرفتن مینیمم وزن عضو هایی که شامل آنهاست می باشد که در رابطه (۳) نشان داده شده است.

$$\mathbf{w(X,t)} = \begin{cases} \min(w(p_1, p_2, ..., p_k) & X \subseteq t \\ 0 & X \not\subset t \end{cases} \qquad (۳)$$

که در آن K تعداد عضو های اقلام می باشد.

تعریف ۲: وزن تراکنش

با اختصاص وزن به اقلام، ما می توانیم به هر تراکنش نیز وزنی نسبت دهیم. نسبت دادن وزن به هر تراکنش به ما اجازه می دهد تفاوت تراکنش های مختلف را بهتر تشخیص بدهیم. بدین ترتیب تراکنشی که وزن و ارزش بالاتری دارد در کاوش نتایج حاصله نقش بیشتری ایفا کرده است. ساده ترین راه برای محاسبه وزن هر تراکنش، بدست آوردن میانگین وزن اقلامی می باشد که تراکنش شامل آنها بوده است. بدین ترتیب وزن هر تراکنش مطابق رابطه (۴) محاسبه می شود.

$$\mathbf{w(t_k)} = \frac{\sum_{i=1}^{|t_k|} w(p_i)}{|t_k|} \qquad (۴)$$

تعریف ۳ : ضریب پشتیبانی وزن دار در میان همه تراکنش ها

ما ضریب پشتیبانی قوانین انجمنی را مطابق روبرو تغییر داده ایم. ضریب پشتیبانی وزن دار آیتم ست X در میان همه تراکنش ها مطابق رابطه زیر تعریف می شود.

$$wsp(X) = \frac{\sum_{t_i \in T} w(t_i) * w(X, t_i)}{\overline{w} * \sum_{k=1}^{|T|} w(t_k)} \qquad (۵)$$

که در آن $\overline{w}$ میانگین وزن همه آیتم ها در کل تراکنش ها می باشد و T نیز مجموعه ای از همه تراکنش ها می باشد.

تعریف ۴ : آیتم ست های تکرار شونده وزن دار

مسئله کشف الگوهای تکرار شونده در کاوش قوانین انجمنی معمول، یافتن مجموعه کاملی از اقلام می باشد که حد آستانه مینیمم ضریب پشتیبانی را در پایگاه داده داشته باشند. در مدل پیشنهاد شده در این بخش، گروه هایی از اقلام را مجموعه اقلام مکرر می نامیم که آستانه‌ی ضریب پشتیبانی وزن دار تعیین شده توسط کاربر را ارضا می‌کنند. در مدل ارائه شده نیز کاوش اقلام تکرار شونده بر اساس الگوریتم Apriori انجام می شود. برای هرس الگوهای غیر تکرار شونده از ویژگی بستاری رو به پایین در الگوریتم Apriori استفاده می کنیم. ویژگی بستاری رو به پایین در الگوریتم Apriori به این شرح است که هر زیر مجموعه ای از اقلام مکرر خود به تنهایی نیز جز اقلام مکرر خواهد بود. به عبارت دیگر اگر یک مجموعه اقلام غیر تکرار شونده باشد، معیار حداقل پشتیبانی را برآورده نسازد، آن‌گاه هیچ یک از ابرمجموعه‌های آن نیز این معیار را برآورده نخواهند کرد. از این ویژگی برای هرس کردن فضای حالت در حین هر تکراراستفاده می شود. در مدلی که در این بخش ارائه دادیم و با توجه به تعاریف ضریب پشتیبانی وزن دار و اقلام مکرر وزن دار، مفهوم جدید ویژگی "بستاری رو به پایین وزن دار "را نیز معرفی می کنیم. این ویژگی همانند ویژگی بستاری رو به پایین در قوانین انجمنی معمول بیانگر آن است که هر زیر مجموعه ای از اقلام مکرر وزن دار نیز اقلام مکرر وزن دار خواهد بود. قضیه بعدی اثبات این ویژگی را برای اقلام مکرر وزن دار نشان می دهد.

قضیه: شمای وزن دهی پیشنهاد شده ویژگی بستاری رو به پایین را برآورده می کند و برای اقلام مکررکاندید همه زیر مجموعه های آن نیز کاندید خواهند بود.

اثبات: I_1 و I_2 مجموعه ای از اقلام در یک مجموعه از تراکنش می باشند به طوریکه $I_1 \subset I_2$ می باشد به عبارت دیگر I_2 بالا مجموعه I_1 است. برای اثبات ویژگی "بستاری رو به پایین" در الگوریتم ارائه شده فرض می کنیم I_1 عضو مجموعه اقلام مکرر نیست در حالیکه I_2 عضو مجموعه اقلام مکرر می باشد. T_1 را مجموعه ای از تراکنش ها که شامل I_1 هستند تعریف کرده و مشابه آن T_2 رانیز مجموعه ای از تراکنش هایی که شامل I_2 هستند قرار می دهیم. از انجا که I_2 بالا مجموعه I_1 است بنابراین $T_2 \subset T_1$ و $\sum_{t \in T_1} w(t) \geq \sum_{t \in T_2} w(t)$.

مطابق تعریفی که برای ضریب پشتیبانی وزن دار اقلام ارئه دادیم، ضریب پشتیبانی وزن دار I_1 و I_2 به ترتیب $wsp(I_1) = \frac{\sum_{t \in T_1} w(t) * w(I_1, t)}{\overline{w} * \sum_{t \in T_1} w(t)}$ و $wsp(I_2) = \frac{\sum_{t \in T_2} w(t) * w(I_1, t)}{\overline{w} * \sum_{t \in T_2} w(t)}$ است. با مقایسه $wsp(I_1)$ و $wsp(I_2)$ و در نظر گرفتن این حقیقت که $\sum_{t \in T_1} w(t) \geq \sum_{t \in T_2} w(t)$ به $wsp(I_1) \geq wsp(I_2)$ می رسیم. از آنجا که I_1 عضو مجموعه اقلام مکرر نیست ضریب پشتیبانی وزن دار آن از حد آستانه مینیمم ضریب پشتیبانی کمتر است و با توجه به $wsp(I_1) \geq wsp(I_2)$؛ $wsp(I_2)$ نیز از حد آستانه مینیمم ضریب پشتیبانی کمتر خواهد بود بنابراین I_2 نیز عضو مجموعه اقلام مکرر نخواهد بود. بنابراین برای اقلام مکررکاندید همه زیر مجموعه های آن نیز عضو مجموعه اقلام مکرر خواهند بود. ■

این قضیه صحت ویژگی "بستاری رو به پایین وزن دار" را در الگوریتم ارائه شده اثبات می کند.

[1] Itemset

پس از استخراج آیتم ست های تکرار شونده وزن دار آنها در یک گـراف مستقیم بدون دور[2] ذخیره می شوند. از انجا که تنها اقلام وزن دار تکرار شونده در این ساختمان داده ذخیره مـی شـوند ایـن سـاختمان داده را گراف اقلام مکرر وزن دار می نامیم. هر نود این گراف شامل اقلام مکـرر در کنار ضریب پشتیبانی وزن دار آنها می باشد. این گراف از سطح ۰ تـا K که K ماکزیمم سایز اقلام وزن دار تکرار شـونده اسـت سـازماندهی می شود. هر نود در عمق d این گراف متناظر با آیتم ست I با سـایز d است که به آیتم ست های سطح d+1ای که شامل آیتمهای I هستند لینک دارد. برای نود N این گراف، که شامل اقلام I است همه فرزندان N معرف اقلام مکرر $I \cup \{p\}$ در سطح d+1 هستند. ریشه این گـراف نیز در سطح صفر شامل آیتم ست خالی است. برای آنکه بتوانیم ترتیـب های متفاوتی از نشست جاری کاربر را بـا اقـلام وزن دار مکـرر منطبـق کنیم همه اقلام قبل از آنکه در گراف چـیده شوند بر اساس حروف الفبا مرتب شده سپس وارد گراف می شوند. به همین ترتیب نشسـت جـاری کاربر پیش از تطبیق با الگوهای استخراج شده نیز به همین روش مرتب می شود. با استفاده از این ساختمان داده کـافی اسـت امتیـاز پیشـنهاد تنها برای مجموعه محدودی از صفحات محاسبه شـود. ایـن امـر زمـان فرایند تولید پیشنهاد را تا حد زیادی بهبود می‌دهد. اگر نشسـت کـاربر با پنجره ای به طول w و آیتمهای تکرار شـونده بـه عنـوان ورودی بـه الگوریتم داده شوند، الگوریتم همه آیتم ست های تکـرار شـونده سـطح w را با جستجوی اول عمق برسی می کند. اگر تطابقی پیدا شود تمام فرزندان نود N به طول $|w|+1$ که شامل نشست جاری هسـتند(w) هستند در مجموعه پیشنهادی کاندید قرار می گیرند. هر یک از فرزندان نود N معرف اقلام مکرر $w' \cup \{p\}$ می باشند. سپس به ازای صـفحات موجود در مجموعه پیشنهادی کاندید ارزش پیشنهاد هر صفحه مطـابق رابطه زیر محابسه می شود.

$$score(p) = \frac{wsp(w' \cup \{p\})}{wsp(w')} \prod_{<u,v>\in w'} p(u,v) \qquad (۶)$$

در آن: $p(u,v)$: احتمال انتقال بین صفحات است که با استفاده از الگوریتم آتاماتای یادگیر معرفی شده در [۳] محاسبه می شود. $\frac{wsp(w' \cup \{p\})}{wsp(w')}$ نیز برابر نسبت ضریب پشتیبانی وزن دار $w' \cup \{p\}$ به ضریب پشتیبانی وزن دار w' می باشد که در بخش قبل معرفی شدند.

۲-۲ خوشه بندی صفحات وب بر اساس الگوی رفتار کاربران

پس از محاسبه امتیاز پیشنهاد برای همه صفحات متعلق به مجموعه کاندید، با مرتب کردن صفحات بر اساس امتیاز آنها، برای حل مشکل کیفیت پایین پیشنهاد بیش از یک صفحه و مشکل صفحات جدیدی که ممکن است در فایل ثبت وقایع ظاهر نشوند، صفحه ای را که بیشترین امتیاز را دارد انتخاب می‌کنیم. سپس این صفحه را بر اساس صفحات مشابه که با آن صفحه در یک خوشه قرار دارند، بسط می‌دهیم و N صفحه (اندازه پنجره پیشنهاد) را از آن خوشه به همراه صفحه اصلی انتخاب می کنیم. در گام بعدی $N+1$ صفحه انتخاب شده در این مرحله را با استفاده از الگوریتم HITS بسط می دهیم. تا صفحات مهم با تکرار کم که ارزش پیشنهاد را دارند نیز در مجموعه صفحات پیشنهاد شده حضور داشته باشند.. برای بسط صفحه، در این قسمت الگوریتم خوشه‌بندی جدیدی مبتنی بر اتوماتای یادگیر و افراز گراف ارائه می‌کنیم که بدون استفاده از هیچگونه اطلاعاتی در باره محتوای اسناد و صرفا با استفاده از الگوی رفتار کاربران، بدون نیاز به استفاده از روشهای شباهت فاصله و با استفاده از روش‌های مبتنی برگراف صفحات را دسته‌بندی می‌کند.

در الگوریتم پیشنهاد شده برای دسته بندی صفحات وب، از یک اتوماتای یادگیر توزیع‌شده با n اتوماتای یادگیر با تعداد اقدامهای متغیر][2] [که هر یک n-1 اقدام دارند، استفاده می‌شود تا در نهایت ماتریس انتقال P بر اساس رفتار همه کاربران تولید شود. تعداد اقدامهای اتوماتای یادگیر متناظر با هر صفحه مانند i برابر است با تعداد صفحاتی که ممکن است کاربر بعد از آن صفحه مشاهده کند. هنگامیکه یک کاربر پس از مشاهده صفحه i، صفحه j را مشاهده می‌کند اقدام jام در اتوماتای iام پاداش می‌گیرد. جزییات بیشتر این الگوریتم و نحوه تولید ماتریس P مبتنی بر آتوماتای یادگیر و استفاده از اطلاعات پیمایش کاربران در[۳] ارائه شده است. در این روش بعد از اتمام یادگیری از اطلاعات پیمایش تمام کاربران، احتمال اقدام jام در اتوماتای iام در درایه p_{ij} قرار می‌گیرد که بیانگر احتمال مشاهده دو صفحه iام و jام به طور متوالی است. ماتریس نامتقارن تولید شده مبتنی بر آتوماتای یادگیر P $(p_{ij} \neq p_{ji})$را ماتریس انتقال صفحات می نامیم. از حاصلضرب ماتریس نامتقارن P با ماتریس وارون[3] آن P^T ماتریس متقارن جدیدی به نام ماتریس شباهت S حاصل می شود. درایه s_{ij} در این ماتریس، درجه شباهت دو صفحه iام و jام را نشان می دهد.

$$S = P \cdot P^T$$
$$s_{ij} = \sum_k p_{ik} p_{kj} \qquad (۷)$$

در مرحله بعدی یک گراف به روش زیر از روی ماتریس شباهت S ایجاد می شود. صفحات را به عنوان مجموعه راس های گراف و درایه های غیر صفر ماتریس یال های این گراف را تشکیل می دهند. سپس با استفاده از ابزارهای موجود افراز گراف، مانند $MeTis$[4]، گراف تولید شده را افراز می‌کنیم. معیار افراز، کمینه سازی برش بین افرازها

[2] acyclic

[3] Transpose

[4] www.cs.umn.edu/~karypis/metis

می‌باشد. این معیار باعث می شود که شباهت بین بخش‌های ایجاد شده از افراز کمینه شود و همچنین صفحاتی که در یک بخش از افراز قرار می‌گیرند، صفحاتی باشند که بین کاربران زیادی مشترک بوده‌اند و مجموعه صفحات موجود در آن بخش یک دسته بندی از صفحات را تشکیل می‌دهند.

۲-۳ گسترش مجموعه اولیه با استفاده از الگوریتم HITS

همان طور که اشاره شده برای بهبود دقت پایین الگوریتم با افزایش تعداد صفحات پیشنهادی و مشکل صفحات جدیدی که ممکن است در فایل ثبت وقایع ظاهر نشوند از ساختار پیوند سایت به صورت زیر استفاده می‌کنیم. صفحات مشابه با نشست کاربر ($N+1$ صفحه)، که در بخش قبل انتخاب شدند را به عنوان ریشه گراف همسایگی در الگوریتم HITS انتخاب می‌کنیم. در ادامه این ریشه به وسیله همسایگانش تکمیل می گردد. همسایه ها، مجموعه‌ای از صفحاتی هستند که یا از ریشه به آنها پیوند داده شده است و یا به ریشه پیوند داده‌اند. سپس با استفاده از الگوریتم HITS برای هر گره در گراف همسایگی، به طور تناوبی دو امتیاز Authority و Hub را محاسبه می کند. صفحات جدید و صفحات با فرکانس کم، به علت اینکه به آنها پیوندهای زیادی وجود نداشته است، در فایل ثبت وقایع کاربران کمتر مشاهده شده اند. بنابراین درجه Authority بالایی هم نخواهند داشت، پس گره ها را تنها بر اساس امتیاز Hub آنها مرتب می‌کنیم و فقط صفحاتی با بیشترین امتیاز مرکزیت به کاربر پیشنهاد می‌شوند.

مراحل اعمال الگوریتم HITS برای توسعه مجموعه اولیه بر روی صفحات وب، به شرح زیر است:

ایجاد مجموعه ریشه[5] : در مرحله اول، ورودی الگوریتم صفحه ای است که بالا ترین امتیاز را در بخش بالا بدست آورد. سپس مجموعه‌ای از صفحات مرتبط با این صفحه که طبق الگوریتم خوشه بندی معرفی شده در یک خوشه قرار دارند انتخاب شده و مجموعه ریشه شامل $N+1$ صفحه، ساخته می‌شود.

ایجاد مجموعه پایه[6]: در این مرحله مجموعه ریشه که در مرحله قبل ایجاد شد، با استفاده از صفحات موجود در همسایگی این صفحات که اعضای مجموعه با آنها در مدل ساختاری پیوند دارند، گسترش می یابد و مجموعه پایه را می‌سازند. برای این منظور ابتدا صفحاتی که صفحات مجموعه ریشه به آنها اشاره می کنند، به مجموعه پایه اضافه می شوند. سپس صفحاتی که به صفحات مجموعه ریشه اشاره می کنند، به این مجموعه اضافه می شوند. البته از آنجا که ممکن است، تعداد این صفحات زیاد باشد، حدی برای تعداد آنها در نظر گرفته می شود.

محاسبه امتیاز Hub و Authority : امتیاز Hub و Authority هر یک از صفحات مجموعه پایه محاسبه می شود[۵].

۲-۴ پیـــشنهاد صفحات وب

یکی از مشکلات الگوریتم‌های پیشنهاد صفحه، حل مشکل صفحات جدید می‌باشد. صفحات جدید و صفحات با فرکانس کم از آنجا که به آنها پیوندهای زیادی وجود نداشته است، در فایل ثبت وقایع کاربران کمتر مشاهده شده اند و بنابراین درجه Authority بالایی هم نخواهند داشت. بنابراین صفحات تنها بر اساس امتیاز Hub آنها رتبه بندی می‌شوند و هنگام ارائه پیشنهادها، با در نظر گرفتن صفحات بازدید شده توسط کاربر، فقط صفحات با بیشترین امتیاز Hub (بسته به تعداد صفحات پیشنهادی از ۱ تا m صفحه) به عنوان صفحات پیشنهادی در نظر گرفته می شوند و به کاربر پیشنهاد می شوند[۱].

۳- ارزیابی الگوریتم پیشنهادی

در این مقاله ما از داده‌های اسـتاندارد سـایت CTI DePaul [7] اسـتفاده می‌کنیم. برای بررسی دقت الگوریتم ارائه شـده رویـه زیـر اتخـاذ شـده است. ابتدا با استفاده از مجموعه یادگیری الگوریتم را اجرا می‌کنیم. بر اساس مقدار w ، از هر نشست در مجموعه تست که اندازه آن حـداقل $w+rt$ می‌باشـد، w صـفحه متـوالی را انتخـاب کـرده و بـه الگـوریتم می‌دهیم. فرض کنیم مجموعه $rp=\{x_{w+1},x_{w+2},...,x_{w+|rs|}\}$ صفحات مشاهده شده توسط کاربر در ادامه نشست واقعی باشد. درجه شـباهت مجموعـه پیشنهادی و مجموعه صفحات واقعی و درصد پوشش آنها از روابـط زیـر به دست می‌آیند:

$$\Pr ecision(rs,rp)=\frac{|rs\cap rp|}{|rs|} \qquad (۸)$$

$$Coverage(rs,rp)=\frac{|rs\cap rp|}{|rp|} \qquad (۹)$$

معیار $Pr\,ecision$ همپوشـانی دو مجموعـه یعنـی نسـبت پیشـنهادات مناسب را به تعداد کل پیشنهادات نشـان مـی دهـد. معیـار $Coverage$ درصدی از کل صفحاتی که سیستم قادر است پیشنهاد بدهـد را نشـان می دهد .

۲-۳ مقایسه الگوریتم ارائه شده با دیگر روشها

در این بخش، ابتدا عملکرد الگوریتم پیشـنهاد شـده بـا الگـوریتم هـای پیشنهاد براساس آتاماتای یادگیر(DLA Based)[۳] و پیشنهاد بر پایـه قوانین انجمنی وزن دار(WAR Based)[۱۰] مقایسه شده است.

جدول(۱): پارامترهای استفاده شده در الگوریتم پیشنهاد شده

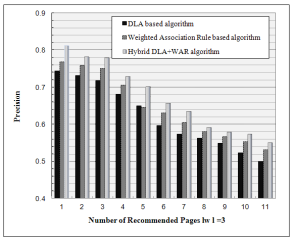
تعداد اعضای مجموعه پایه	تعداد اعضای مجموعه ریشه	w اندازه پنجره لغزان
۳۰	۱۰	۳

[5] Root Set

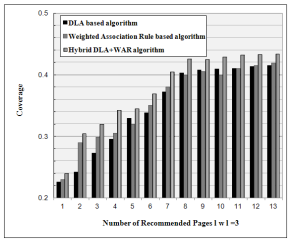
[6] Base Set

[7] http://maya.cs.depaul.edu/classes/ect584/data/cti-data.zip.

نتایج این مقایسه ها در شکل های (۱) و (۲) نشان داده شده است. در شکل (۱) می توان خلاصه عملکرد این روشها را بر مبنای دقت پیشنهادها مشاهده کرد. همان طور که انتظار می رود با افزایش تعداد صفحات پیشنهادی، بطور طبیعی و همانطور که انتظار می رود، دقت پیشنهاد ها در هر ۳ روش کاهش پیدا می کند. در این بین، روش پیشنهادی ما نتایج بهتری از دو روش دیگر کسب می کند. مقایسه درصد پوشش الگوریتم ها در شکل (۲) نشان داده شده است. با افزایش تعداد صفحات پیشنهادی الگوریتم به طور صعودی نتایج بهتری نسبت به DLA Based [۳] و WAR Based [۱۰] از خود نشان می دهد.



شکل(۱): مقایسه دقت الگوریتم پیشنهادی با الگوریتمهای[۱۰, ۳]

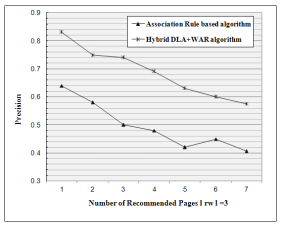


شکل(۲): مقایسه پوشش الگوریتم پیشنهادی با الگوریتمهای[۱۰, ۳]

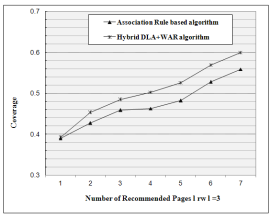
در ادمه کارایی الگوریتم ترکیبی با یکی از الگوریتم های رایج AR [۱۱] مقایسه شده است. شکل های (۳) و (۴) مقایسه این دو الگوریتم را در معیار دقت و پوشش بر روی مجموعه داده CTI نشان می دهند. با افزایش تعداد صفحات پیشنهادی دقت الگوریتم ها کاهش می یابد اما روند کاهش دقت الگوریتم ترکیبی پیشنهاد شده در مقایسه با AR شیب کمتری داشته و نتایج بهتری کسب کرده است. در نتیجه، روش پیشنهاد شده ، قادر به ارائه عملکردی بهتر(در قالب دقت و پوشش) در مقایسه با روش مبنای استفاده شده است.

۴- نتیجه گیری

در این مقاله الگوریتم جدیدی معرفی شده است که "آیتم ست های تکرار شونده وزن دار" را استخراج می کند و با استفاده از آیتم های وزن دار تولید شده و آتاماتای یادگیر توزیع شده صفحه اول را یافته و با استفاده از الگوریتم تحلیل پیوند HITS و ساختار سایت صفحات وب موردنیاز کاربر جاری را پیشنهاد می دهد. الگوریتم ارائه شده همچنین مشکل صفحات جدید و صفحاتی که به دلیل ساختار بد سایت فرکانس مشاهده کمتری دارند ولی ارزش مشاهده شدن را دارند حل می کند. این الگوریتم فرصت حضور صفحات مهم که ارزش پیشنهادی آنها بالا می باشد در مجموعه صفحات پیشنهادی را فراهم می کند. همچنین الگوریتم ارائه شده زمان ارائه پیشنهاد را در حد قابل توجهی کاهش می دهد. الگوریتم ارائه شده، از دقت و درصد پوشش بالایی برخوردار بوده و مناسب برای شخصی سازی وب برخط می‌باشد.



شکل(۳): مقایسه دقت الگوریتم پیشنهادی با AR



شکل(۴): مقایسه درصد پوشش الگوریتم پیشنهادی با الگوریتم AR

۵- مراجع

[1] O. R. Za¨ıane, j. Li, R. Hayward, Mission-Based Navigational Behaviour Modeling for Web Recommender Systems, Springer-Verlag Berlin Heidelberg, 2006.

[2] M. A. L. Thathachar, R. Harita Bhaskar, Learning Automata with Changing Number of Actions, IEEE Transactions on Systems Man and Cybernetics, vol. 17, no. 6, 1987.

[3] R. Forsati, M. Meybodi, M. Mahdavi, Web Personalization based on Distributed Learning Automata and PageRank Alogorithm, in Information and Knowledge Technology Discovery, Mashahd, Iran, 2008.

[4] A. B. Hashemi, M. R. Meybodi, Web Usage Mining Using Distributed Learning Automata, f 12th Annual CSI Computer Conference of Iran, Tehran, Iran, 2007.

[5] J. Kleinberg, Authoritative Sources in a Hyperlinked Environment, Journal of the ACM, Vol. 46, 1999.

[6] P. Kazienko, M. Adamski, AdROSA - Adaptive Personalization of Web Advertising. Information Sciences 177(11), 2007, pp. 2269-2295.

[7] P. Kazienko, Filtering of Web Recommendation Lists Using Positive and Negative Usage Patterns , Springer-Verlag Berlin Heidelberg, 2007.

[8] F. Heylighen, J. Bollen, Hebbian Algorithms for a Digital Library Recommendation System, Proceedings of the International Conference on Parallel Processing Workshops (ICPPW 02), 2002, pp. 439-446.

[9] J. Zhu, J. Hong, J. G. Hughes, Mining Conceptual Link Hierarchies from Web Log Files for Adaptive Web Site Navigation, ACM Transactions on internet Technology, 2003.

[10] R. Forsati, M. R. Meybodi, An Efficient Algorithm based on Web Usage Data and Structure of the Site for Web Page Recommendation, in the Second Data Mining Conference, Amirkabir University of Technology, Tehran, Iran, 2008.

[11] B. Mobasher, H. Dai, T. Luo, M. Nakagawa, Effective Personalization based on Association Rule Discovery from Web Usage Data, Proceedings of the 3rd ACM Workshop on Web Information and Data Management, 2001.