

۱۲۴۹

سیستم پیشنهاد دهنده وب با استفاده از وب معنایی، اتوماتای یادگیر توزیع شده و پارتیشن بندی گراف

شهرزاد معتمدی مهر^۱، مجید تاران^۲، علی برادران هاشمی^۳، محمدرضا میبیدی^۴

چکیده

هدف سیستم های پیشنهاد دهنده وب هدایت کاربران به سمت صفحاتی است که به بهترین وجه نیازها و علایق آنها را برآورده سازد. در این مقاله یک الگوریتم جدید مبتنی بر وب معنایی، اتوماتای یادگیر توزیع شده و پارتیشن بندی گراف پیشنهاد می گردد. در الگوریتم پیشنهادی یک اتوماتای یادگیر توزیع شده بر اساس داده های استفاده کاربران از وب، گراف پیوند بین صفحات و وب معنایی موجود در URL، شباهت صفحات یک سایت با یکدیگر را مشخص می کند. سپس یک الگوریتم PageRank بر اساس این اتوماتای یادگیر توزیع شده امتیاز صفحات را محاسبه می کند. الگوریتم پیشنهادی با ایجاد یک مدل مارکوف بر اساس اطلاعات فوق صفحات جدیدی را برای ادامه حرکت هر کاربر در سایت به وی پیشنهاد می دهد. نتایج آزمایشات انجام شده نشان می دهد که الگوریتم پیشنهادی در مقایسه با روش های گزارش شده مبتنی بر قوانین انجمنی و اتوماتای یادگیر توزیع شده از دقت بیشتری برخوردار است.

کلمات کلیدی

اتوماتای یادگیر، وب معنایی، داده کاوی استفاده از وب، سیستم های پیشنهاد دهنده

کنفرانس داده کاوی ایران

^۱ دانشکده فنی و مهندسی، دانشگاه خوارزمی، تهران، ایران، motamedi@khu.ac.ir

^۲ شرکت خدمات انفورماتیک، تهران، ایران، m_taran@isc.iranet.net

^۳ دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران، a_hashemi@aut.ac.ir

^۴ دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران، mmeybodi@aut.ac.ir

Web Recommendation using Semantic Web and Distributed Learning Automata and Graph Partitioning

Shahrzad Motamedi Mehr; Majid Taran; Ali B. Hashemi; M.R. Meybodi

ABSTRACT

Recommendation systems aim at directing users toward the resources that best meet their needs and interests. One of the challenging tasks in improving web recommendation algorithms is the simultaneous use of users's activity log and hyperlink graph of the web site. In this paper, we propose a new recommendation algorithm based on semantic web and web usage data and hyperlink graph of a web site. In the proposed algorithm, a distributed learning automata learns similarity between web pages of a web site using web usage data and hyperlink graph of the web site and semantic web in URL address. Then, a usage based page rank for all pages of the web site is calculated using the probabilities of actions in the distributed learning automata. The proposed algorithm uses these information to build a Markov model which will be used to recommend new web pages for a user. Experiments show that the proposed method outperforms Association Rule Mining algorithm and the only learning automata based method reported in the literature in terms of precision and coverage.

KEYWORDS

Learning Automata, Semantic Web, Web Usage Mining, Recommendations systems



کنفرانس داده کاوی ایران

۱. مقدمه

وب، محیطی وسیع، متنوع و پویا است که کاربران متعدد اسناد خود را در آن منتشر می کنند. وب طی یک فرآیند آشفته و غیر متمرکز رشد می کند و این روند منجر به تولید حجم وسیعی از مستندات متصل به یکدیگر گشته است که از هیچ گونه سازماندهی منطقی برخوردار نیستند. با توجه به حجم وسیع اطلاعات در وب، مدیریت آن با ابزارهای سنتی تقریباً غیر ممکن است و ابزارها و روش هایی نو برای مدیریت آن مورد نیاز است.

برای حل این مشکل، شخصی کردن وب به یک پدیده محبوب به منظور سفارشی کردن محیط های وب تبدیل شده است. هدف از سیستم های شخصی ساز فراهم کردن نیازهای کاربران، بدون اینکه به طور صریح آن ها را بیان کنند یا نشان بدهند، می باشد [۴]. شخصی سازی وب مجموعه ای از عملیات است که تجربه وب را برای یک کاربر خاص یا مجموعه ای از کاربران سازمان دهی می کند [۱۲].

در [۱۲] از روش مبتنی بر قوانین انجمنی (AR) که با استفاده از کاوش آیتم های تکراری به دسته بندی صفحات می پردازد، استفاده شده است. الگوریتم ارائه شده در [۹] مبتنی بر آنالیز لینک ها می باشد که صفحات وب و کاربران سایت را به صورت گره و ابرپیوند مدل می کند و از الگوریتم HITS^۲ برای ارزیابی اهمیت آنها در گراف استفاده می کند و هدف آن اندازه گیری تخصص کاربران و اهمیت صفحات وب است. در [۶] دو متد مجزای رتبه بندی بر اساس آنالیز لینک ها ارائه داده شده است. Mobasher از درجه اتصالات بین صفحات سایت به عنوان فاکتوری تعیین کننده برای پیشنهاد بر اساس کاوش آیتم های تکرار شونده یا کشف الگوهای ترتیبی استفاده می کند [۱۲]. اتوماتای یادگیر توزیع شده قبلاً برای رتبه بندی صفحات وب [۷][۵][۳] و تعیین شباهت اسناد وب بکاربرده شده است. الگوریتم پیشنهادی نیز یک روش مبتنی بر اتوماتای یادگیر توزیعی است. نتایج بررسی ها نشان می دهد که الگوریتم پیشنهادی در مقایسه با سایر روش ها کارایی بالا تری دارد.

در [۱] برای محاسبه شباهت بین صفحات وب از اتوماتای یادگیر توزیع شده استفاده شده است که شباهت بین صفحات را با استفاده از فعل وانفعال کاربر و رابطه تراگذاری تعیین می کند. در [۲] اتوماتای یادگیر توزیع شده شباهت بین صفحات وب را با استفاده از اطلاعات پیمایش کاربران قبلی و پیوند گراف وب سایت تعیین می کند که نسبت به روش ارائه شده در [۱] از کارایی بالاتری برخوردار است در این روش با افزایش تعداد صفحات، مقدار دقت کم نمی شود. برای بالا بردن کارایی الگوریتم با تعداد صفحات زیاد از پارتیشن بندی گراف با استفاده از الگوریتم های چند سطحی (Multilevel) استفاده شده است. در این روش به منظور کاهش اثر اطلاعات ناصحیح، کاربری که از پارتیشن خود خارج شود مسیر اشتباهی طی کرده و میزان شباهت محاسبه شده برای صفحات مسیر خارج از محدوده، با توجه به رابطه ای مشخص کاهش می یابد. جریمه دیگری که در روش پیشنهادی برای کاربر در نظر گرفته شده وجود دور در مسیر پیمایشی کاربر می باشد. در این مقاله با استفاده از الگوریتم ارائه شده در [۲] و وب معنایی موجود در URL شباهت صفحات وب تعیین می گردد. در روش پیشنهادی از Wikipedia برای یافتن ارتباطات و تشکیل گراف معنایی استفاده می گردد. در این روش در صورتی که حرکت کاربر منطبق با لینک موجود در گراف حاصل باشد پاداش تعلق می گیرد.

روش پیشنهادی با ترکیب داده های استفاده کاربران و داده های ساختاری صفحات وب الگوریتمی ترکیبی مبتنی بر اتوماتای یادگیر توزیع شده، پارتیشن بندی گراف و وب معنایی موجود در URL و الگوریتم PageRank به منظور رتبه بندی و پیشنهاد صفحات ارائه شده است. در روش پیشنهادی کاربر با صرف کمترین زمان به نتایج مطلوب خود دست می یابد. ویژگی دیگر روش پیشنهادی آن است که پیمایشگر طراحی شده یک پیمایشگر یادگیر است که می تواند از دانش یاد گرفته شده در پیمایش های متفاوت استفاده کند.

در ادامه ابتدا در بخش ۲ اتوماتای یادگیر و اتوماتای یادگیر توزیع شده به اختصار معرفی می شوند. در بخش ۳ وب معنایی و در بخش ۴ الگوریتم PageRank ارائه می گردد. در بخش ۵ الگوریتم پیشنهادی و در بخش ۶ مدل استفاده شده برای شبیه سازی و نتایج شبیه سازی ارائه و بررسی می گردد. بخش ۷ نتیجه گیری می باشد.

۲. اتوماتای یادگیر

اتوماتای یادگیر یک مدل انتزاعی است که بطور تصادفی یک اقدام از مجموعه متناهی اقدام های خود را انتخاب کرده و بر محیط اعمال می کند. محیط اقدام انتخاب شده توسط اتوماتای یادگیر را ارزیابی کرده و نتیجه ارزیابی خود را توسط یک سیگنال تقویتی به اتوماتای یادگیر اطلاع می دهد. سپس اتوماتای یادگیر با اطلاع از اقدام انتخاب شده و سیگنال تقویتی، وضعیت داخلی خود را بروز کرده و اقدام بعدی خود را انتخاب می کند. شکل ۱ نحوه ارتباط بین اتوماتای یادگیر و محیط را نشان می دهد.



شکل ۱. ارتباط اتوماتای یادگیر با محیط

محیط را می‌توان توسط سه‌تایی $E = \{\alpha, \beta, c\}$ نشان داد که در آن $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه ورودیها، $\beta = \{\beta_1, \beta_2, \dots, \beta_r\}$ مجموعه خروجیها و $c = \{c_1, c_2, \dots, c_r\}$ مجموعه احتمالات جریمه می‌باشد. هرگاه β مجموعه دو عضوی باشد، محیط از نوع P می‌باشد. در چنین محیطی $\beta_1 = 1$ به عنوان جریمه و $\beta_2 = 0$ به عنوان پاداش در نظر گرفته می‌شود. در محیط از نوع Q ، مجموعه β دارای تعداد متناهی عضو می‌باشد و در محیط از نوع K ، تعداد اعضا مجموعه β نامتناهی است. c_i نشان دهنده احتمال نامطلوب بودن سیگنال تقویتی محیط در پاسخ به اقدام α_i می‌باشد. در یک محیط ایستاً^۲ مقادیر c_i ها ثابت هستند، حال آنکه در یک محیط غیر ایستاً^۱ این مقادیر در طی زمان تغییر می‌کنند. بر اساس اینکه تابع بروز رسانی وضعیت اتوماتای یادگیر (که با اطلاع از اقدام انتخاب شده و سیگنال تقویتی β ، وضعیت بعدی اتوماتای یادگیر را محاسبه می‌کند) ثابت یا متغیر باشد، اتوماتای یادگیر به دو دسته اتوماتای یادگیر با ساختار ثابت و اتوماتای یادگیر با ساختار متغیر تقسیم می‌گردند. در این مقاله از اتوماتای یادگیر با ساختار متغیر استفاده شده است که در ادامه معرفی می‌شود.

اتوماتای یادگیر با ساختار متغیر توسط چهارتایی $\{\alpha, \beta, p, T\}$ نشان داده می‌شود که در آن $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ مجموعه اقدام‌های اتوماتای یادگیر، $\beta = \{\beta_1, \beta_2, \dots, \beta_r\}$ مجموعه ورودیهای اتوماتای یادگیر، $p = \{p_1, p_2, \dots, p_r\}$ بردار احتمال انتخاب هر یک از اقدام‌ها و $T, p(n+1) = T[\alpha(n), \beta(n), p(n)]$ الگوریتم یادگیری اتوماتای یادگیر می‌باشد. الگوریتم‌های یادگیری متنوعی برای اتوماتای یادگیر ارائه شده است که در ادامه یک الگوریتم یادگیری خطی برای اتوماتای یادگیر بیان می‌گردد. فرض کنید اتوماتای یادگیر در مرحله n ام اقدام α_i خود را انتخاب نموده و محیط ارزیابی خود را توسط سیگنال تقویتی $\beta(n)$ به اتوماتای یادگیر اعلام کند.

اتوماتای یادگیری که در بالا معرفی شد، دارای تعداد اقدامهای ثابتی می‌باشد. در بعضی از کاربردها به اتوماتای یادگیر با تعداد اقدام متغیر^۵ نیاز می‌باشد. یک *اتوماتای یادگیر با تعداد/اقدام متغیر*، در لحظه n ، اقدام خود را از یک زیر مجموعه غیر تهی از اقدامها بنام مجموعه اقدامهای فعال $V(n)$ انتخاب می‌کند. انتخاب مجموعه اقدامهای فعال اتوماتای یادگیر $V(n)$ توسط یک عامل خارجی و بصورت تصادفی انجام می‌شود. نحوه فعالیت این اتوماتای یادگیر بصورت زیر است.

اتوماتای یادگیر برای انتخاب یک اقدام در زمان n ابتدا مجموع احتمال اقدامهای فعال خود $K(n)$ را محاسبه و بردار $\hat{p}(n)$ را مطابق رابطه (۱) ایجاد می‌کند. آنگاه اتوماتای یادگیر یک اقدام از مجموعه اقدامهای فعال خود را بصورت تصادفی و بر اساس بردار احتمال $\hat{p}(n)$ انتخاب کرده و بر محیط اعمال می‌کند. در یک اتوماتای یادگیر با الگوریتم یادگیری خطی، اگر اقدام انتخاب شده α_i باشد، اتوماتای یادگیر پس از دریافت پاسخ محیط، بردار احتمال $\hat{p}(n)$ اقدامهای خود در صورت دریافت پاسخ مطلوب بر اساس رابطه (۲) و در صورت دریافت پاسخ نامطلوب طبق رابطه (۳) بروز می‌کند. سپس اتوماتای یادگیر بردار احتمال اقدامهای خود $p(n)$ را با استفاده از بردار $\hat{p}(n+1)$ و طبق رابطه (۴) بروز می‌کند. [۸]

$$K(n) = \sum_{\alpha_i \in V(n)} p_i(n)$$

$$\hat{p}_i(n) = \text{prob}[\alpha(n) = \alpha_i | \alpha_i \in V(n)] = \frac{p_i(n)}{K(n)} \quad (1)$$

$V(n)$ is the set of enabled actions

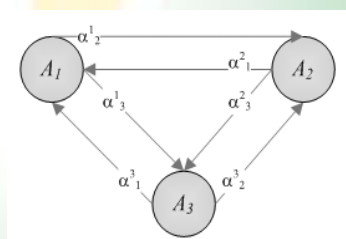
$$\begin{aligned} \hat{p}_i(n+1) &= \hat{p}_i(n) + a.(1 - \hat{p}_i(n)) \\ \hat{p}_j(n+1) &= \hat{p}_j(n) - a.\hat{p}_i(n) \quad \forall j \neq i \end{aligned} \quad (2)$$

$$\begin{aligned}\hat{p}_i(n+1) &= (1-b) \cdot \hat{p}_i(n) \\ \hat{p}_j(n+1) &= \frac{b}{\hat{r}-1} + (1-b) \hat{p}_j(n) \quad \forall j, j \neq i\end{aligned}\quad (3)$$

$$\begin{aligned}p_i(n+1) &= \hat{p}_i(n+1) \cdot K(n) && \text{for all } i, \alpha_i \in V(n) \\ p_j(n+1) &= p_j(n) && \text{for all } j, \alpha_j \notin V(n)\end{aligned}\quad (4)$$

۱.۲. اتوماتای یادگیر توزیع شده

اتوماتای یادگیر توزیع شده شبکه‌ای از چند اتوماتای یادگیر است که برای حل یک مساله مشخص با یکدیگر همکاری می‌کنند. یک اتوماتای یادگیر توزیع شده را می‌توان با یک گراف جهت‌دار مدل کرد. صورتی که مجموعه گره‌های آنرا مجموعه‌ای از اتوماتای یادگیر و یالهای خروجی هر گره مجموعه اقدامهای متناظر با اتوماتای یادگیر متناظر با آن گره است. هنگامی که اتوماتا یکی از اقدامهای خود را انتخاب می‌کند، اتوماتایی که در دیگر انتهای یال متناظر با آن اقدام قرار دارد، فعال می‌شود. بعنوان مثال در شکل ۲ هر اتوماتا ۲ اقدام دارد. اگر اتوماتای A_1 اقدام α_3 خود را انتخاب کند، آنگاه اتوماتای A_3 فعال خواهد شد. در گام بعد، اتوماتای A_3 یکی از اقدامهای خود را انتخاب می‌کند که منجر به فعال شدن یکی از اتوماتاهای یادگیر متصل به A_3 می‌شود. در هر لحظه فقط یک اتوماتای یادگیر در اتوماتای یادگیر توزیع شده فعال می‌باشد. بصورت رسمی، یک اتوماتای یادگیر توزیع شده با n اتوماتای یادگیر توسط یک گراف (A, E) تعریف می‌شود که $A = \{A_1, A_2, \dots, A_n\}$ مجموعه اتوماتا و $E \subset A \times A$ مجموعه لبه‌های گراف است بطوریکه لبه (i, j) متناظر با اقدام α_j از اتوماتای A_i است. اگر بردار احتمال اقدامهای اتوماتای یادگیر A_j با p^j نشان داده شود، آنگاه p_m^j احتمال انتخاب اقدام α_m از اتوماتای یادگیر A_j را نشان می‌دهد که احتمال انتخاب لبه خروجی (j, m) از میان لبه‌های خروجی گره j می‌باشد. [۹][۱۰]



شکل ۲. اتوماتای یادگیر توزیع شده

۳. وب معنایی

به وب متداول امروزی، وقتی اطلاعاتش معنی دار شود، تا رایانه‌ها و افراد توانایی بیشتری برای همکاری و تعامل پیدا نمایند، وب معنایی گفته می‌شود. تعریف تکنیکی تر از وب معنایی تأکید بر نیازی دارد که بتواند بین برنامه‌های نرم افزاری در سطح معنایی و نه تنها در سطح کامپایل کردن نحو، به رد و بدل کردن اطلاعات بپردازد. [۱۱]

از URI برای مشخص کردن مفاهیم در وب معنایی استفاده می‌شود. [۱۳] برای مثال URL نوعی URI است که برای مشخص کردن منابع در وب استفاده می‌شود. قسمت اصلی وب معنایی آن‌تولوژی‌ها هستند که ارتباط بین برچسب‌های اسناد وب معنایی و اشیاء واقعی که اسناد مذکور آنها را تشریح می‌کنند، برقرار می‌کند.

اسناد در وب معنایی ساختاری گرافی دارند که با ساختار اسناد وب کنونی کاملاً متفاوت است. به عبارت دیگر وب معنایی وبی از اسناد نیست بلکه وبی از منابع و روابط بین آنهاست.

۴. الگوریتم PageRank

الگوریتم PageRank معروفترین الگوریتم تحلیل پیوند صفحات وب می‌باشد که در سال ۱۹۹۸ ارائه گردید [۱۵] [۱۴]. این الگوریتم با انتساب وزن به هر صفحه نتایج جستجو را بر اساس این وزن‌ها مرتب می‌کند. از دید الگوریتم PageRank صفحه‌ای مهم و معتبر است که از صفحات مهم و معتبر دیگر مورد اشاره باشد. در این دیدگاه تنها به وجود پیوندها توجه نمی‌شود، بلکه کیفیت آنها نیز مورد استفاده قرار می‌گیرد. این الگوریتم جستجو در وب را بصورت یک حرکت تصادفی در نظر می‌گیرد که در آن گردشگر بطور تصادفی یک لینک را انتخاب نموده و لینک‌ها را پشت سر هم دنبال می‌کند تا به هدف برسد، همچنین در صورت شروع یک مسیر دیگر برای پیمایش، به صفحه جدید پرش می‌کند. به

عبارت دیگر الگوریتم PageRank به عنوان مدلی از حرکت تصادفی بر روی وب در نظر گرفته می شود [۱۴]. این مدل معادل با یک زنجیره مارکوف می باشد. حالت های این زنجیره صفحات وب هستند و گذار از یک حالت به حالت دیگر معادل انتخاب یک پیوند در صفحه جاری و رفتن به صفحه مورد اشاره است.

فرض کنید $G = (V, E)$ گراف متناظر وب باشد. $u \rightarrow v$ را به عنوان وجود پیوند از صفحه u به v و $\deg(u)$ را برابر با درجه خروجی صفحه u در نظر می گیریم. فرض کنیم p ماتریس انتقال گذار بین صفحات باشد. اگر گردش تصادفی در لحظه k در صفحه u قرار داشته باشد در لحظه $k+1$ به احتمال مساوی طبق رابطه (۵) به یکی از صفحات مجاور $\{v | u \rightarrow v\}$ خواهد رفت.

$$p_{ij} = \begin{cases} \frac{1}{\deg(u_i)} & : \text{if } (u_i \rightarrow u_j) \in E \\ 0 & : \text{otherwise} \end{cases} \quad (5)$$

برای اینکه زنجیره مارکوف دارای توزیع پایدار باشد، هیچ صفحه ای نباید درجه خروجی صفر داشته باشد. در گراف وب صفحات زیادی وجود دارند که بدون پیوند هستند. برای رفع این مشکل فرض می شود وقتی که گردش گر تصادفی به چنین صفحاتی می رسد، به طور تصادفی و با احتمال برابر از صفحه ای دیگر شروع به گردش می کند. اگر n تعداد صفحات وب و \bar{v} بردار ستونی n بعدی توزیع یکنواخت احتمال باشد و \bar{d} بردار ستونی n بعدی مشخص کننده رؤس با درجه خروجی صفر باشد ($d_u = 1$ به معنای درجه خروجی صفر برای راس u می باشد) در این صورت ماتریس p' طبق رابطه (۶) بدست می آید.

$$D = \bar{d} \bar{v}^T \quad (6)$$

$$P' = P + D$$

شرط بعدی لزوم توزیع احتمالی پایدار در زنجیره های مارکوف، کاهش ناپذیر بودن و به عبارت دیگر به شدت همبند بودن گراف وب است. برای رفع این کمبود فرض می کنیم که گردشگر با یک احتمال یکنواخت و بسیار کم از هر صفحه به هر صفحه دیگر می تواند جهش داشته باشد. با افزودن این ویژگی گراف وب به شدت همبند شده و دارای توزیع احتمالی پایدار خواهد بود. در این صورت ماتریس گذار طبق رابطه (۷) بدست می آید.

$$E = [1]_{n \times 1} \bar{v}^T \quad (7)$$

$$P'' = cP' + (1-c)E, \quad 0 \leq c \leq 1$$

که در آن c ضریب تعدیل با مقدار معمول ۰٫۸۵ می باشد. \bar{v} بردار شخصی سازی نامیده می شود که می تواند به نفع صفحات خاص با مقداری بیشتر در درایه مورد نظر آن صفحه، تنظیم شود [۱۶].

در [۱۷] Eirinaki و Vazirgiannis الگوریتم PageRank شخصی سازی مبتنی بر استفاده (UPR^{*}) را که بر اساس رفتار کاربران قبلی عمل می کند، برای رتبه بندی صفحات وب پیشنهاد کرده اند. w_i وزن صفحه i است که مدت زمان مشاهده صفحه i و w_{ij} وزن انتقال از صفحه i به j است و مدت زمان انتقال به صفحه j که بلافاصله بعد از صفحه i مشاهده می شود را نشان می دهد. در UPR احتمال انتقال از صفحه i به j متناسب با لینک به صفحه j در صفحه i است و طبق رابطه (۸) بدست می آید.

$$P_{ij}^{UPR} = \frac{w_{ij}}{\sum_{x_k \in out(k)} w_{ik}} \quad (8)$$

بطوری که $out(i)$ مجموعه صفحات لینک شده از صفحه i را مشخص می کند. بردار شخصی سازی \bar{v} بر اساس رابطه (۹) تعریف می شود.

$$v_{ij}^{UPR} = \frac{w_i}{\sum_k w_k} \quad (9)$$

۵. الگوریتم پیشنهادی

روش ارائه شده برای پیشنهاد صفحات به کاربر از دو مرحله کلی تشکیل شده است. در مرحله اول، با استفاده از Wikipedia روابط استخراج و ساختار ارتباطی حاصل از گراف وب معنایی موجود در URI ها بدست می آید، سپس با استفاده از الگوریتم ارائه شده در [۲] و

گراف حاصل از وب معنایی شباهت بین صفحات وب بدست می آید. سپس در مرحله دوم یک رتبه بندی جدید صفحه مبتنی بر استفاده از وب، با بکاربردن شباهت تعیین شده با استفاده از الگوریتم ارائه شده در [۳] محاسبه می شود. رتبه صفحه و شباهت بین صفحات محاسبه شده برای ایجاد مدل زنجیره مارکوف انتقال های کاربران در وب سایت استفاده می گردد. این مدل مارکوف برای پیش بینی احتمال ملاقات صفحات جدید برای پیشنهاد به کاربر استفاده می شود. جزئیات الگوریتم پیشنهادی در زیر آمده است.

۱.۵. تعیین شباهت بین صفحات

الگوریتم پیشنهادی، یک اتوماتای یادگیر توزیع شده را بکار می گیرد که شباهت بین صفحات را با استفاده از فعل وانفعال کاربر و گراف پیوندها [۲] و گراف معنایی تعیین می کند. در روش پیشنهادی میزان جریمه و پاداش مطابق با الگوریتم ارائه شده در [۲] محاسبه می گردد با این تفاوت که در روش پیشنهادی چنانچه در حرکت کاربر از صفحه i به j لینک ارتباطی در گراف معنایی وجود داشته باشد نیز پاداش تعلق می گیرد. نحوه پاداش در این الگوریتم با در نظر گرفتن رابطه تراگذاری می باشد. مثلاً اگر کاربر در یک مسیر به ترتیب صفحات $i1, i2, i3$ را مشاهده کرده باشد، با فرض اینکه وجود شباهتی بین محتوای این صفحات موجب این انتخاب کاربر شده است، به اعمال هر یک از اتوماتاهای متناظر با این صفحات پاداش داده می شود.

- عمل $i2$ از اتوماتای $i1$ ، عمل $i3$ از اتوماتای $i2$
- عمل $i3$ از اتوماتای $i2$

پاداشی که به هر یک از اعمال $i2, i3$ از اتوماتای یادگیر $i1$ داده می شود، طبق یک ضریب کاهشی، به ترتیب کاهش داده می شود. چرا که فاصله صفحات متناظر آنها در مسیر کاربر به ترتیب افزایش می یابد. همچنین در صورتی که صفحه $i1, i2$ در گراف وب معنایی به یکدیگر متصل و صفحه $i2$ و $i3$ غیرمتصل باشند، پاداشی که عمل $i2$ در اتوماتای $i1$ می گیرد، بیشتر از پاداشی است که عمل $i3$ در اتوماتای $i2$ می گیرد.

- پاداش دادن به اعمال انتخاب شده توسط اتوماتای یادگیر به چهار عامل بستگی دارد:

۱. مسیرهای طی شده توسط کاربران
۲. فاصله صفحات در مسیرهای طی شده
۳. پیوند بین صفحات در گراف وب
۴. پیوند بین صفحات در گراف معنایی

- جریمه دادن اعمال انتخاب شده توسط اتوماتای یادگیر [۲]، به دو عامل بستگی دارد:

۱. وجود دور در مسیر حرکت کاربر
۲. خارج شدن از زیر گراف مربوط به اتوماتا

۲.۵. رتبه بندی صفحات مبتنی بر استفاده

در [۶] فرصتی و میبیدی با استفاده از اتوماتای یادگیر توزیع شده برای محاسبه PageRank شخصی سازی مبتنی بر استفاده، ارائه کرده اند. در این روش، احتمال انتقال از صفحه i به j برابر با احتمال انتخاب اقدام j در اتوماتای i (رابطه (۱۰)) و بردار شخصی سازی \vec{v} مطابق با رابطه (۹) است.

$$p_{ij}^{DLA-UPR} = s(i, j) \quad (10)$$

در الگوریتم پیشنهادی، ماتریس انتقال p و بردار شخصی سازی \vec{v} در الگوریتم اصلی PageRank بجای استفاده از ساختار پیوندها، مبتنی بر داده کاوی استفاده از وب محاسبه می شود. به همین دلیل اتوماتای یادگیر توزیع شده (DLA) ماتریس احتمال انتقال p از طریق رفتار کاربران موجود در لاگ فایل های^۸ سایت، یاد بگیرند. تعداد دفعات مشاهده هر صفحه توسط کاربران به عنوان معیاری برای مقادردهی بردار شخصی سازی \vec{v} استفاده می گردد. این مقدار دهی بر اساس تعداد مشاهدات هر صفحه است و به درستی یک بردار احتمال است زیرا که مجموع همه عناصر آن برابر ۱ می باشد. بدیهی است که هر چه صفحه ای دفعات زیادی مشاهده شده باشد، نسبت به صفحات دیگر مهم تر

می‌باشد. با داشتن ماتریس احتمال انتقال p و بردار شخصی سازی \vec{v} که از طریق اطلاعات پیمایش های کاربران قبلی بدست می‌آیند، الگوریتم PageRank طبق رابطه (۱۱) برای محاسبه رتبه هر صفحه استفاده می‌شود.

$$DLA - UPR(p_i) = d \cdot \sum_{p_j \in in(p_i)} \frac{DLA - UPR(p_j)}{|out(p_j)|} + (1-d) \frac{1}{N} \quad (11)$$

$$DLA - UPR(p_i) = d \cdot \sum_{p_j \in in(p_i)} s(j, i) + (1-d) \frac{1}{N}$$

بطوری که $out(p_j)$ به مجموعه صفحات لینک شده از صفحه j اشاره دارد و $in(p_j)$ مجموعه صفحاتی است که به صفحه j لینک دارند. N تعداد صفحات وب، $UPR(p_j)$ رتبه صفحه p_j و d ضریب تعدیل می‌باشد که عددی بین صفر و یک است و ۰٫۸۵ مقداردهی شده است.

۳.۵. پیشنهاد صفحات

هدف از شخصی سازی بر اساس اطلاعات پیمایش کاربران محاسبه یک مجموعه پیشنهادی، rs ، برای نشست کاربر جاری می‌باشد [۱۸]. 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. فرض کنیم که کاربری که در حال گردش در سایت است و مسیر $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow \dots \rightarrow p_k$ را پیموده است. تعداد آخرین صفحاتی را که توسط کاربر مشاهده شده و برای پیشنهاد صفحات جدید مورد استفاده قرار می‌گیرد را پنجره پیشنهاد می‌نامیم و اندازه آن را با rw نشان می‌دهیم که حداکثر برابر با تمام صفحات مشاهده شده و حداقل برابر با آخرین صفحه مشاهده شده می‌باشد. برای پیشنهاد صفحه P_{k+1} به کاربر از خاصیت مارکوف گراف استفاده می‌کنیم. طبق قاعده زنجیر مارکوف احتمال انتخاب مسیر در گراف دارای ویژگی مارکوف، از رابطه (۱۲) به دست می‌آید:

$$\Pr(p_1 \rightarrow p_2 \rightarrow p_3 \dots \rightarrow p_k) = \Pr(p_1) \times \prod_{i=2}^k \Pr(p_i | p_{i-1} \dots p_{i-m}) \quad (12)$$

به عنوان مثال، احتمال مسیر $p_1 \rightarrow p_2 \rightarrow p_3$ برابر است با:

$$\Pr(p_1 \rightarrow p_2 \rightarrow p_3) = \Pr(p_1) \Pr(p_2 | p_1) \Pr(p_3 | p_2) = \Pr(p_1) \frac{\Pr(p_1 \rightarrow p_2) \Pr(p_2 \rightarrow p_3)}{\Pr(p_1) \Pr(p_2)} \quad (13)$$

که در آن $\Pr(\bullet \rightarrow \bullet)$ برابر با احتمال گذار بین دو صفحه است و $\Pr(\bullet)$ احتمال حالت پایدار صفحه متناظر می‌باشد که در دو بخش قبل به ترتیب در ماتریس p و بردار \vec{x} محاسبه شدند. برای پیشنهاد صفحه به کاربر، به ازای صفحات مختلف P_{k+1} که در مسیر $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow \dots \rightarrow p_k \rightarrow p_{k+1}$ ملاقات نشده‌اند، احتمال مسیر را محاسبه می‌نماییم. احتمال هر مسیر امتیاز صفحه P_{k+1} را برای پیشنهاد به کاربر نشان می‌دهد. با مرتب کردن صفحات بر اساس امتیاز آنها، صفحاتی با بیشترین امتیاز به کاربر پیشنهاد می‌شود. برای هر یک از تعداد صفحات پیشنهادی دقیقاً فقط d صفحه (برابر با تعداد صفحات پیشنهادی) که بیشترین امتیاز را دارند، به کاربر پیشنهاد می‌شود.

کنفرانس داده کاوی ایران

۶. ارزیابی الگوریتم پیشنهادی

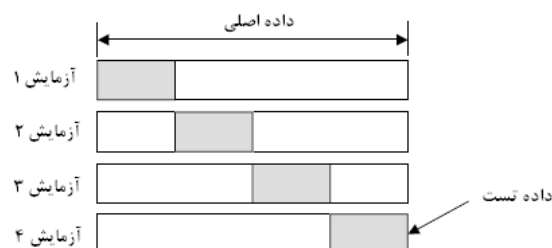
در این قسمت ابتدا مدل بکار رفته برای تولید داده استفاده از وب و معیار ارزیابی تشریح می‌گردد. سپس نتایج آزمایشات الگوریتم پیشنهادی با الگوریتم های قبلی مقایسه می‌گردد.

۱.۶. مدل شبیه سازی

دو روش عمده برای ارزیابی الگوریتم‌هایی که از اطلاعات پیمایش کاربران استفاده می‌کنند وجود دارد. روش اول، استفاده از صفحات وب واقعی و داده‌های واقعی کاربران وب موجود در فایل‌های ثبت رخداد سایت‌ها می‌باشد. روش دوم مدل ارائه شده در [۱۹] می‌باشد. در این روش Liu و همکارانش نظم موجود در رفتارهای کاربران در محیط وب را با استفاده از یک مدل مبتنی بر عامل، مشخص و اعتبار مدل خود را با استفاده از چندین سایت وب بزرگ مانند مایکروسافت، تایید کرده‌اند. در این مقاله ما از داده‌های استاندارد سایت CTI DePaul استفاده

می‌کنیم. این مجموعه داده اطلاعات نشست کاربران را به مدت ۲ هفته در سایت CTI DePaul در سال ۲۰۰۲ شامل می‌شود [۲۰]. این اطلاعات پیش‌پردازش شده و نشست‌های با اندازه ۱ و غیر استاندارد از آن حذف شده‌اند و در نهایت اطلاعات ۱۳۷۴۵ کاربر که از ۶۸۳ صفحه دیدن کرده‌اند در فایل‌های جداگانه قرار داده شده است.

برای انتخاب دو مجموعه آموزشی و تست از معیار K -Fold استفاده شده است [۲۱]. در این روش نمونه اصلی بطور تصادفی به k زیر نمونه تقسیم می‌گردد. برای هر k زیر نمونه، تنها یکی از زیر نمونه‌های بدست آمده به عنوان داده تست و $k-1$ زیر نمونه باقیمانده به عنوان داده آموزشی استفاده می‌گردد. این پروسه به تعداد k بار تکرار می‌شود و به این ترتیب دقیقاً همه k زیر نمونه موجود، به عنوان داده تست استفاده می‌گردد. در نهایت برای تولید یک تخمین واحد، بین k نتیجه میانگین گرفته می‌شود. مزیت روش K -Fold این است که با تکرار زیر نمونه‌های تصادفی تمام مجموعه هم به عنوان داده آموزشی و هم داده تست استفاده می‌شوند. این روش در شکل ۳ نشان داده شده است.



شکل ۳. روش K-Fold

در اتوماتای یادگیر توزیع شده، احتمال انتخاب عمل j در اتوماتای i ، میزان ارتباط دو صفحه i ام و j ام را نشان می‌دهد. ساختار ارتباطی پیشنهادی با یک ماتریس $n \times n$ به نام P بازنمایی می‌شود. در صورت فعال بودن عمل j در اتوماتای i ، درایه P_{ij} این ماتریس برابر با احتمال عمل j در اتوماتای i و در غیر اینصورت برابر با صفر قرار داده می‌شود. از آنجاییکه فرایند آموزش ترتیب توالی دسترسی‌ها را در نظر می‌گیرد نتایج فرایند یادگیری با یک ماتریس نامتقارن $(p_{ij} \neq p_{ji})$ بازنمایی می‌شود. ماتریس نامتقارن تولید شده مبتنی بر اتوماتای یادگیر را ماتریس انتقال صفحات می‌نامیم. از حاصلضرب ماتریس نامتقارن P با ماتریس ترانزپوز P^T ، ماتریس متقارن جدیدی به نام ماتریس شباهت S طبق رابطه (۱۴) حاصل می‌شود.

$$S = P \cdot P^T$$

$$s_{ij} = \sum_k a_{ik} a_{kj} \quad (14)$$

درایه s_{ij} در این ماتریس، درجه شباهت دو صفحه i ام و j ام را نشان می‌دهد.

۲.۶. معیار ارزیابی

برای ارزیابی، دو معیار "پوشش" و "دقت" معرفی می‌شوند. این دو معیار، بسیار شبیه به معیارهای متداول در بازیابی اطلاعات یعنی "فراخوانی" و "دقت" بازیابی اسناد هستند.

دقت پیشنهادها برابر با "نسبت پیشنهادهای درست به کل پیشنهادها" است. منظور از پیشنهاد درست، پیشنهادی است که با توجه به بخش دیده شده (پیشوند) یک جلسه کاربر تولید شده و در ادامه جلسه کاربر (پسوند) رخ دهد. اگر تعداد U جلسه کاربر را در نظر بگیریم، برای هر جلسه مثل u به ترتیب صفحات بازدید شده را، یک به یک، به مجموعه صفحات بازدید شده اضافه می‌کنیم. سپس، با دیدن هر صفحه p پیشنهادهایی تولید می‌کنیم. این مجموعه پیشنهاد را $R(p)$ (مجموعه صفحات پیشنهاد شده پس از بازدید کاربر از صفحه p) می‌نامیم. سپس $R(p)$ با قسمت باقیمانده از جلسه کاربر، که آن را $Tail(p)$ یا به اختصار $T(p)$ می‌نامیم، مقایسه می‌شود. دقت پیشنهادها برابر با درصد اشتراک $T(p)$ و $R(p)$ خواهد بود و طبق رابطه (۱۵) محاسبه می‌گردد.

$$Precision = \frac{T(p) \cap R(p)}{R(p)} \quad (15)$$

پوشش پیشنهادها، قدرت سیستم در پیش بینی تمام صفحاتی که ممکن است مورد نظر کاربران باشد را اندازه گیری می کند. این عدد برابر با "نسبت صفحات درست صفحات پیش بینی شده در ادامه جلسه یا $T(p)$ به کل صفحات باقیمانده (تعداد صفحات $T(p)$) در جلسه در هر قدم است" و طبق رابطه (۱۶) محاسبه می گردد.

$$Coverage = \frac{T(p) \cap R(p)}{T(p)} \quad (16)$$

هر چه مقدار دقت و پوشش بالاتر باشد، کارایی الگوریتم، مطلوب تر است. بررسی این وضعیت با استفاده از معیار F1 ساده تر می باشد (رابطه (۱۷)).

$$F1 = \frac{2 \times Coverage \times Precision}{Coverage + Precision} \quad (17)$$

همانطور که رابطه (۱۷) نشان می دهد، هرچه مقدار دقت و پوشش بالاتر باشد، مقدار F1 نیز افزایش می یابد.

۳.۶. تنظیمات

در آزمایشات، برای پارتیشن بندی گراف وب سایت از ابزار Metis [۲۲]، استفاده شده است که بر اساس الگوریتم های پارتیشن بندی چند سطحی [۲۰] کار می کند. تعداد پارتیشن ها $k=25$ در نظر گرفته شده است. همچنین در الگوریتم پیشنهادی، a_0 و b_0^{cycle} ، 0.02 و 0.3 و b_0 به ترتیب 0.02 و 0.3 مقداردهی می گردد. همچنین برای آماده سازی داده تست و آموزشی مقدار پارامتر k در روش $K-Fold$ ۲۰ در نظر گرفته شده است.

۴.۶. پارامترهای موثر در ارزیابی

اندازه طول پنجره پیشنهاد (rw) و تعداد صفحات پیشنهادی که دو معیار "دقت" و "پوشش" برحسب آنها اندازه گیری می شوند، پارامترهای تاثیرگذار در کارایی الگوریتم هستند. در واقع براساس طول پنجره که مسیر پیمایش کاربر می باشد مسیر بعدی کاربر را پیش بینی می گردد. ابتدا با استفاده از مجموعه یادگیری الگوریتم اجرا شده و سپس بر اساس مقدار طول پنجره، rw صفحه متوالی را انتخاب کرده و به الگوریتم داده می شود. معیار ارزیابی رابطه معرفی شده در [۲۳] است. فرض کنیم مجموعه $rp = \{x_{rw+1}, x_{rw+2}, \dots, x_{rw+|rs|}\}$ صفحات مشاهده شده توسط کاربر در ادامه نشست واقعی باشد. درجه شباهت مجموعه پیشنهادی و مجموعه صفحات واقعی از رابطه (۱۸) به دست می آید.

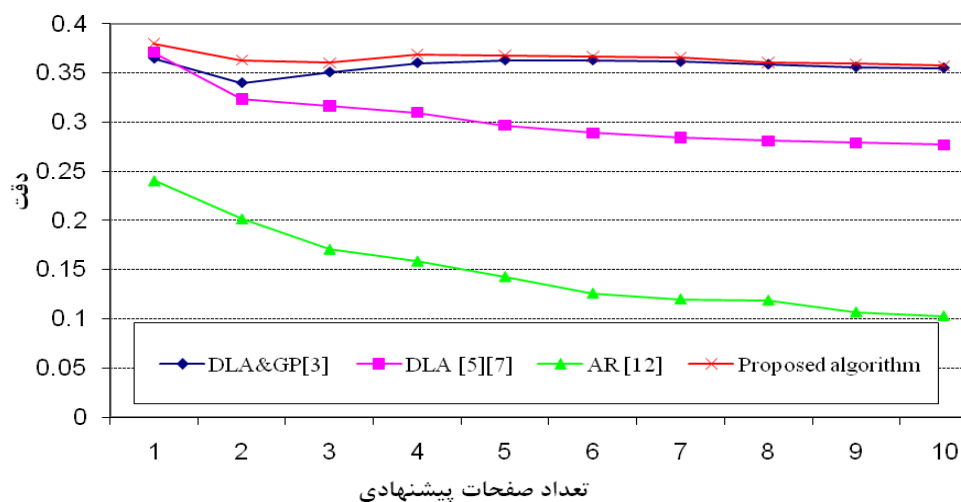
$$Sim(rs \cap rp) = \frac{rs \cap rp}{rp} \quad (18)$$

۵.۶. نتایج آزمایشات

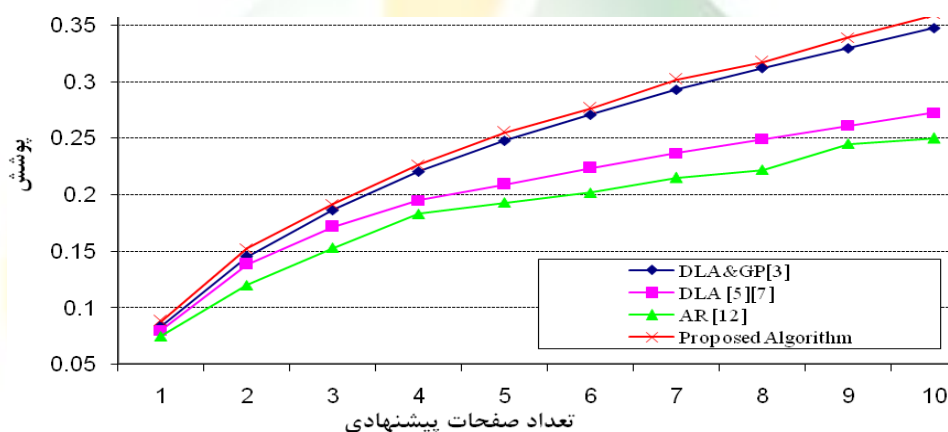
الگوریتم پیشنهادی با روش های مبتنی بر اتوماتای یادگیر توزیع شده [۷][۵][۳] که از یک روش آماری ساده نیز استفاده می کنند، مقایسه می گردد. در این روش آماری شباهت دو سند i و j ، بر اساس نسبت تعداد دفعاتی که کاربران از سند i به سند j حرکت کرده اند به تعداد دفعاتی که کاربران از سند i به هر سند دیگری مانند k حرکت نموده اند، طبق رابطه (۱۹) محاسبه می شود.

$$simpleSimilarity(i, j) = \frac{visited(i, j)}{\sum_{k=1}^n visited(i, k)} \quad (19)$$

شکل ۴ و ۵ به ترتیب دقت و پوشش الگوریتم پیشنهادی با الگوریتم های مبتنی بر اتوماتای یادگیر توزیع شده [۷][۵][۳] و قوانین انجمنی [۱۲] نسبت به تعداد صفحات پیشنهادی مختلف و با طول پنجره ۴، مقایسه شده است. همانطور که مشاهده می شود، الگوریتم پیشنهادی از دقت و پوشش بالاتری برخوردار است.

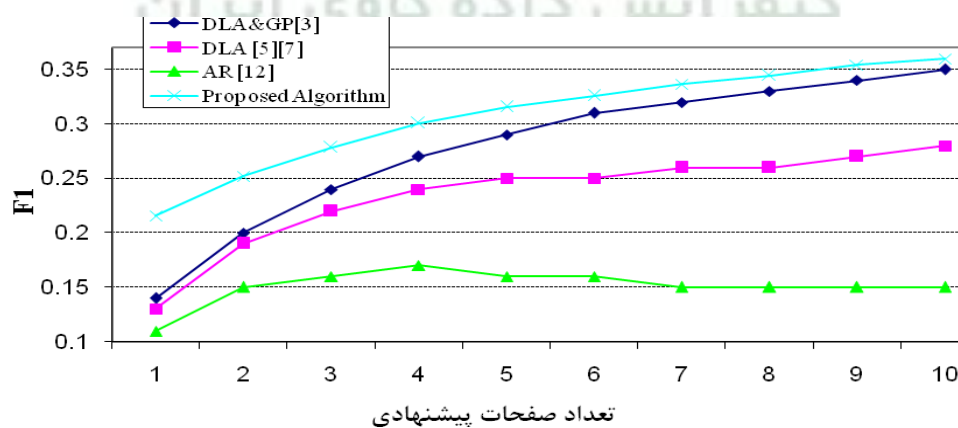


شکل ۴. مقایسه دقت الگوریتم پیشنهادی با الگوریتم های DLA [۵][۷] و AR [۱۲] و DLA&GP [۳]

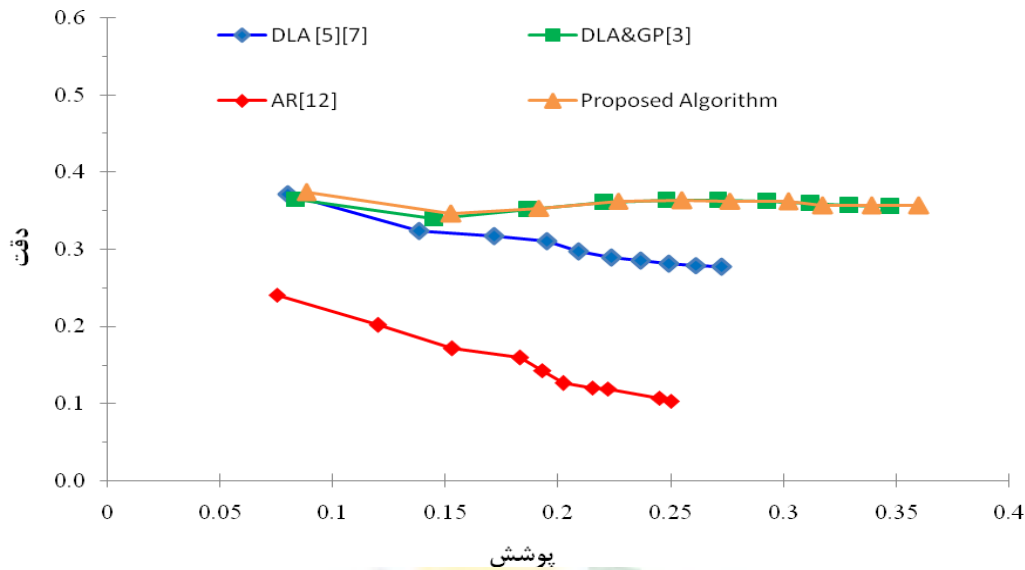


شکل ۵. مقایسه پوشش الگوریتم پیشنهادی با الگوریتم های DLA [۵][۷] و AR [۱۲] و DLA&GP [۳]

شکل ۶ مقایسه الگوریتم پیشنهادی با الگوریتم های DLA [۵][۷] و AR [۱۲] و DLA&GP [۳] براساس معیار F1 و با طول پنجره ۴ است و نشان می دهد که الگوریتم پیشنهادی نسبت به الگوریتم های قبلی نتایج از دقت و پوشش بالاتری برخوردار است.



شکل ۶. مقایسه الگوریتم پیشنهادی با الگوریتم های DLA [۵][۷] و AR [۱۲] و DLA&GP [۳] براساس معیار F1 و با طول پنجره ۴



شکل ۷. بررسی الگوریتم پیشنهادی با الگوریتم های DLA [۵][۷] و AR [۱۲] و DLA&GP [۳]

براساس نسبت دقت به پوشش با طول پنجره ۴

در شکل ۷ الگوریتم پیشنهادی با الگوریتم های DLA [۵][۷] و AR [۱۲] و DLA&GP [۳] مقایسه شده است که براساس مقدار دقت به پوشش و با طول پنجره ۴ به دست آمده است. همانطور که مشاهده می شود الگوریتم پیشنهادی نسبت به الگوریتم های قبلی نتایج بهتری را ارائه می دهد.

۷. نتیجه گیری

در این مقاله سعی کردیم تا با استفاده از ساختار گراف معنایی روش [۵][۷] [۳] که مبتنی بر اتوماتای یادگیر توزیع شده می باشد، بهبود دهیم. نتایج شبیه سازیها نشان داد که روش پیشنهادی در مقایسه با روش های گزارش شده مبتنی بر اتوماتای توزیع شده و قوانین انجمنی در تشخیص شباهت صفحات و پیشنهاد به کاربر از کارایی بالاتری برخوردار است. بصورتیکه دقت و پوشش بدست آمده با اندازه پنجره ۴، در الگوریتم پیشنهادی بیشتر از این مقدار در الگوریتم معرفی شده در [۵][۷] [۳] و [۱۲] می باشد.

- [1] SH. Motamedi Mehr, M. Taran, and M. R. Meybodi, "Web Usage Mining based on Distributed Learning Automata", Proceedings of 10th Fuzzy Conference of Iran, Shahid Beheshti University, Tehran, Iran, 2010.
- [2] SH. Motamedi Mehr, M. Taran, A. B. Hashemi, and M. R. Meybodi, "Determining Web Pages Similarity Using Distributed Learning Automata and Graph Partitioning," International Symposium on Artificial Intelligence and Signal Processing (AISP) IEEE Iran Section, Tehran, Iran , 2011.
- [3] SH. Motamedi Mehr, M. Taran, A. B. Hashemi, and M. R. Meybodi, " Web Recommendation Using Distributed Learning Automata and Graph Partitioning," Proceedings of 4th Data Mining of Iran, Sharif University, Tehran, Iran , 2010.
- [4] B. Mobasher, R. Cooley, J. Srivastava, "Automatic Personalization Based on Web Usage Mining", Communications of the ACM, Vol. 43, No. 8, 2000.
- [5] R. Forsati, M. R. Meybodi, "Web Personalization Using Distributed Learning Automata", Proceedings of 3th Iran Information and knowledge Conference, Mashhad, Iran, 2007.
- [6] J. Wang, Z. Chen, L. Tao, W. Ma, L. Wenyn, "Ranking User's Relevance to a Topic through Link Analysis on Web Logs", Proceeding of the WIDM '02, 2002.
- [7] R. Forsati, M. R. Meybodi, "Effective page recommendation algorithms based on distributed learning automata and weighted association rules". Expert Systems with Applications: An International Journal, 37, 2 (2010), 1316-1330.
- [8] K. S. Narendra and M.A.L. Thathachar, **Learning Automata: An Introduction**, Prentice Hall, 1989.
- [9] H. Beigy, and M. R. Meybodi, "Utilizing Distributed Learning Automata to Solve Stochastic Shortest Path Problem," International Journal of Uncertainty, Fuzziness and Knowledge-based Systems, vol. 14, no. 5, pp. 591-617, 2006.
- [10] M. A. L. Thathachar and R. Harita Bhaskar, "**Learning Automata with Changing Number of Actions**," IEEE Transactions on Systems Man and Cybernetics, vol. 17, no. 6, Nov. 1987, pp 1095-1100.
- [11] Z. Liu, W. K. Ng, E.-P. Lim, and F. Li. Towards Building Logical Views of Websites. Data & Knowledge Engineering, (2004)
- [12] 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.
- [13] Z. Liu, W. K. Ng, E.-P. Lim, and F. Li. Towards Building Logical Views of Websites. Data & Knowledge Engineering, (2004)
- [14] S. Brin, L. Page, "The anatomy of a large-scale hypertextual Web search engine". Computer Networks and ISDN Systems, 30, 1-7 (1998), 107-117.
- [15] L. Page, S. Brin, R. Motwani, T. Winograd, "The PageRank citation ranking: bringing order to the web". Computer Science Department, Stanford University, Palo Alto, California, USA, 1999.
- [16] S. Kamvar, T. Haveliwala, G. Golub, "Adaptive methods for the computation of PageRank". Linear Algebra and its Applications, 386(2004), 51-65.
- [17] M. Eirinaki, M. Vazirgiannis, "Web site personalization based on link analysis and navigational patterns". ACM Transactions on Internet Technology, 7, 4 (2007), 21.
- [18] B. Mobasher, H. Dai, T. Luo, M. Nakagawa, "Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization", Data Mining and Knowledge Discovery, pp. 61-82, 2002.
- [19] J. Liu, S. Zhang, J. Yang, "Characterizing Web Usage Regularities with Information Foraging Agents", IEEE Transactions on Knowledge and Data Engineering, pp. 566-584, 2004.
- [20] <http://maya.cs.depaul.edu/~classes/ect584/data/cti-data.zip>
- [21] G. J. McLachlan, K. A. Do, C. Ambrose, "Analyzing microarray gene expression data". Wiley, 2004.
- [22] G. Karypis, V. Kumar, "METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices". Minneapolis, City, 2007.
- [23] T. Haveliwala, "Topic-Sensitive PageRank", in Proceedings of the 11th International Conference on World Wide Web, New York: ACM Press, pp. 517-526, 2002.

¹ Association Rule

² Hyperlink-Induced Topic Search

³ Stationary

⁴ Non-Stationary

⁵ Learning automata with changing number of actions

⁶ Usage-based Personalization PageRank

⁷ Distributed Learning Automata

⁸ Log File

⁹ Transpose

¹⁰ Coverage

¹¹ Precision

¹² Recall