

روشی جدید مبتنی بر اتوماتاهای یادگیر توزیع شده برای کشف ساختار اسناد

وب

راهبه مجتبه‌ی صفاری^۱، محمد رضا میبدی^۲

۱ عضویات علمی دانشکده مهندسی کامپیوتر، دانشگاه آزاد اسلامی، واحد لاهیجان، ایران

۲ عضویات علمی دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران

mmeybodi@aut.ac.ir, mojtahedi@iau-lahijan.ac.ir

چکیده:

وب یک شبکه بزرگ از اسناد مرتبط به هم می‌باشد که بسرعت در حال رشد است. بیشتر تحقیقات انجام شده در زمینه وب شامل استفاده همزمان از اطلاعات ساختاری و اطلاعات نحوه پیمایش کاربران از اسناد وب می‌باشد. در این مقاله روشی جدید مبتنی بر اتوماتاهای یادگیر توزیع شده برای کشف ساختار ارتباطی بین اسناد وب ارائه شده که در تعیین میزان شباهت بین اسناد وب در امر خوش بندی، رتبه بندی، شخصی سازی وب و کشف زیر گرافهای متصل وب مورد استفاده قرار می‌گیرد. در الگوریتم پیشنهادی کشف ساختار، مانند سایر روشهای گزارش شده به هر سند وب یک اتوماتای یادگیر تخصیص داده می‌شود که بر اساس اطلاعات پیمایش کاربران ارتباطات بین اسناد را یاد می‌گیرد. تفاوت الگوریتم پیشنهادی نسبت به سایر روشهای موجود مبتنی بر اتوماتاهای یادگیر، عملکرد تصادفی و منتفاوت اتوماتاهای یادگیر در انتخاب اقدام بعدی و نحوه تخصیص پاداش به اقدامها می‌باشد. نتایج شبیه سازیها نشان می‌دهد که الگوریتم پیشنهادی به دلیل استفاده از اتوماتاهای یادگیر با تعداد اقدامهای متغیر و بروز رسانی پارامتر یادگیری در حین افزایش تعداد صفحات وب، نسبت به سایر الگوریتم‌های گزارش شده مبتنی بر اتوماتاهای یادگیر دارای کارایی بالاتر و پیچیدگی زمانی پایینتری می‌باشد.

واژه‌های کلیدی:

آنتروپویی، اتوماتای یادگیر، اتوماتای یادگیر توزیع شده، کاوش استفاده از وب

۱- مقدمه

استفاده از تکنیک‌های داده کاوی به منظور استخراج اتوماتیک اطلاعات از اسناد وب را کاوش وب می‌گویند. کاوش وب در سه سطح مطرح می‌شود. در سطح محتوا، در سطح ساختار و در سطح استفاده از وب. کاوش محتوای وب در مورد کشف و بازیابی اطلاعات مفید از محتوا و داده‌های وب است. کاوش ساختار وب بر کشف و چگونگی مدل کردن ساختار ارتباطات اسناد وب تأکید دارد. در سطح استفاده از وب، هدف کشف اطلاعاتی است که از نحوه استفاده کاربران با وب بدست می‌آید. بیشتر روشهای موجود برای کشف ساختار ارتباطی اسناد وب از اطلاعات موجود در لاغ فایلهای استفاده می‌نمایند. لاغ فایلهای، فایلهایی هستند که درخواستهای کاربران را به هنگام مراجعت به صفحات وب ثبت می‌کنند. افراد یک مدل ذهنی از ارتباطات بین مفاهیم مختلف در ذهن خود دارند و نحوه حرکت آنها در بین اسناد

را توسط یک سیگنال تقویتی به اتماتای یادگیر اطلاع می‌دهد. سپس اتماتای یادگیر با اطلاع از عمل انتخاب شده و سیگنال تقویتی، وضعیت داخلی خود را بروز کرده و عمل بعدی خود را انتخاب می‌کند. محیط را می‌توان توسط سه تایی $E = \{a, \beta, c\}$ نشان داد که در آن $a = \{a_1, a_2, \dots, a_r\}$ مجموعه ورودی ها، $\beta = \{\beta_1, \beta_2, \dots, \beta_r\}$ مجموعه خروجیها و $c = \{c_1, c_2, \dots, c_r\}$ مجموعه احتمالات می‌باشد. c_i نشان دهنده احتمال نامطلوب بودن سیگنال تقویتی محیط در پاسخ به عمل a_i می‌باشد. در یک محیط ایستا مقادیر c_i ثابت هستند، حال آنکه در یک محیط غیر ایستا این مقادیر در طی زمان تغییر می‌کنند. بر اساس اینکه تابع بروز رسانی وضعیت اتماتای یادگیر (که با اطلاع از عمل انتخاب شده و سیگنال تقویتی β ، وضعیت بعدی اتماتای یادگیر را محاسبه می‌کند) ثابت یا متغیر باشد، اتماتای یادگیر به دو دسته اتماتای یادگیر با ساختار ثابت و اتماتای یادگیر با ساختار متغیر تقسیم می‌گردد. در این مقاله از اتماتای یادگیر با ساختار متغیر استفاده شده است که در ادامه معرفی می‌شود. اتماتای یادگیر با ساختار متغیر توسط چهارتاپی $\{a, \beta, p, T\}$ نشان داده می‌شود که در آن $a = \{a_1, a_2, \dots, a_r\}$ مجموعه اعمال اتماتای یادگیر، $\beta = \{\beta_1, \beta_2, \dots, \beta_r\}$ مجموعه ورودی های اتماتای یادگیر، $p = \{p_1, p_2, \dots, p_r\}$ بردار احتمال انتخاب هر یک از عمل ها و $T = [T(a(n), \beta(n), p(n))]$ الگوریتم یادگیری اتماتای یادگیر می‌باشد. الگوریتم های یادگیری متعدد برای اتماتای یادگیر ارائه شده است که در ادامه یک الگوریتم یادگیری خطی برای اتماتای یادگیر بیان می‌گردد. فرض کنید اتماتای یادگیر در مرحله n اقدام a خود را انتخاب نموده و محیط ارزیابی خود را توسط سیگنال تقویتی $\beta(n)$ به اتماتای یادگیر اعلام کند. با استفاده از الگوریتم یادگیری خطی، اتماتای یادگیر بردار احتمال انتخاب اقداماتی خود را مطابق روابط ابروز می‌کند.

$$p_i(n+1) = p_i(n) + a(1 - \beta(n)). \\ (1 - p_i(n)) - b\beta(n)p_i(n)$$

$$p_j(n+1) = p_j(n) + a(1 - \beta(n)). \quad \text{if } j \neq i \\ p_j(n) + \frac{b\beta(n)}{r-1} - b\beta(n)p_j(n)$$

که در آن a پارامتر پاداش و b پارامتر جریمه می‌باشد.

اگر $a = b$ با هم برابر باشند، الگوریتم یادگیری، L_{RP} ، اگر b از

بزرگ از استناد، قابل استفاده نمی‌باشد. استفاده از روش‌های مارکوف هم به دلیل نیاز به محاسبه توان $\#$ ماتریس انتقال، دارای پیچیدگی محاسباتی بالایی می‌باشد.

در این مقاله یک الگوریتم جدید مبتنی بر اتماتای یادگیر توزیع شده، بمنظور کشف ساختار ارتباطی بین استناد و ب پیشنهاد می‌گردد. الگوریتم پیشنهادی، بدلیل استفاده از اتماتاهای یادگیر با تعداد اقداماتی متغیر، در مقایسه با الگوریتم ارائه شده در [۱]، با افزایش تعداد استناد و ب، دارای همگرایی بالاتر است. در روش ارائه شده در [۱]، بعلت استفاده از اتماتاهای یادگیر با تعداد اعمال برابر استناد، با افزایش تعداد استناد و ب طول بردار اعمال اتماتای یادگیر افزایش یافته که این باعث کاهش سرعت همگرایی می‌گردد. همچنین روش پیشنهادی بدلیل متغیر در نظر گرفتن پارامتر یادگیری، بر خلاف روش مبتنی بر اتماتاهای یادگیر با تعداد اقداماتی متغیر [۳] با افزایش تعداد استناد، نیازی به تنظیم مجدد پارامتر یادگیری ندارد. همچنین روش پیشنهادی برخلاف روش گزارش شده در [۲]، همزمان با حرکت کاربر از یک سند به سند دیگر، میزان ارتباط بین استناد را بصورت برخط محاسبه می‌نماید. علاوه بر این، بعلت اینکه در الگوریتم پیشنهادی فرایند محاسبه دور وجود ندارد، در مقایسه با روش ارائه شده در [۲]، دارای سیار پیشنهادی کمتری می‌باشد. بمنظور ارزیابی و مقایسه روش پیشنهادی با سایر روش‌ها از مدل ارائه شده توسط Liu و همکاران [۱۱]، استفاده شده است.

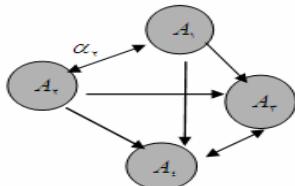
ادامه مقاله بدین صورت سازماندهی شده است. در بخش ۲ اتماتاهای یادگیر و اتماتای یادگیر توزیع شده به اختصار شرح داده می‌شوند. در بخش ۳ الگوریتم پیشنهادی برای کشف ساختار استناد و ب با استفاده از اتماتای یادگیر توزیع شده ارائه می‌گردد. در بخش ۴ نتایج شبیه سازی آمده است. بخش ۵ نتیجه گیری می‌باشد.

۲- اتماتاهای یادگیر

اتماتای یادگیر یک مدل انتزاعی است که بطور تصادفی یک عمل از مجموعه متناهی اعمال خود را انتخاب کرده و بر محیط اعمال می‌کند. محیط عمل انتخاب شده توسط اتماتای یادگیر را ارزیابی کرده و نتیجه ارزیابی خود

۱-۲- اتماتای یادگیر توزیع شده

اتماتای یادگیر توزیع شده شبکه‌ای از چند اتماتای یادگیر است که برای حل یک مساله مشخص با یکدیگر همکاری می‌کنند. یک اتماتای یادگیر توزیع شده را می‌توان بصورت یک گراف جهت‌دار نمایش داد. بصورتیکه مجموعه‌گرهای آن را مجموعه‌ای از اتماتاهای یادگیر و یالهای خروجی هر گره مجموعه اعمال متناظر با اتماتای یادگیر متناظر با آن گره است. هنگامی که اتماتا یکی از اعمال خود را انتخاب می‌کند، اتماتایی که در دیگر انتهای یال متناظر با آن عمل قرار دارد، فعل می‌شود. در هر لحظه فقط یک اتماتای یادگیر در اتماتای یادگیر توزیع شده فعل می‌باشد. بعنوان مثال در شکل ۱، یک اتماتای یادگیر توزیع شده با چهار اتماتای یادگیر را نشان می‌دهد. برای اطلاعات بیشتر درباره اتماتای یادگیر توزیع شده میتوان به مراجع [۱۰-۴] مراجعه کرد.



شکل ۱: اتماتای یادگیر توزیع شده

۳- الگوریتم پیشنهادی

در روش پیشنهادی، اسناد وب و کاربران استفاده کننده از آن نقش یک محیط تصادفی را برای اتماتای یادگیر موجود در اتماتای یادگیر توزیع شده ایفا می‌کنند. خروجی یک اتماتای یادگیر توزیع شده، یک دنباله از اسناد وب مرور شده توسط یک کاربر هستند که مسیر حرکت کاربر را به سمت یک سند وب مورد نظر نشان می‌دهد. محیط با استفاده از این دنباله، پاسخی برای اتماتای یادگیر توزیع شده تولید کرده و با استفاده از این پاسخ، ساختار داخلی اتماتای یادگیر در اتماتای یادگیر توزیع شده، طبق الگوریتم یادگیری بروز می‌شود. در این مقاله برای تعیین ساختار ارتباطی بین اسناد وب، در یک مجموعه با n سند، از یک اتماتای یادگیر توزیع شده با n اتماتای یادگیر با تعداد اقدامهای متغیر [۱۰]، که هر یک حداقل $n-1$ اقدام دارند، استفاده می‌شود. برای هر اتماتای یادگیر در هر زمان تنها

a خیلی کوچکتر باشد، الگوریتم یادگیری، L_{REP} و اگر b صفر باشد، الگوریتم یادگیری، L_{RI} نام دارد.

اتماتای یادگیری که در بالا معرفی شد، دارای تعداد اقدام های ثابتی می باشد. در بعضی از کاربردها به اتماتای یادگیر با تعداد اقدام متغیر نیاز می باشد [۱۰]. یک اتماتای 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 می کند. انتخاب مجموعه اقدامهای فعال اتماتای یادگیر $V(n)$ توسط یک عامل خارجی و بصورت تصادفی انجام می شود. نحوه فعالیت این اتماتای یادگیر بصورت زیر است. اتماتای 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 (۳) ایجاد می کند. آنگاه اتماتای یادگیر یک اقدام از مجموعه اقدامهای فعال خود را به صورت تصادفی و بر اساس بردار احتمال $\hat{P}(n)$ انتخاب کرده و بر محیط اعمال می کند. در یک اتماتای یادگیر با الگوریتم یادگیری خطی، اگر اقدام انتخاب شده α_i باشد، اتماتای یادگیر پس از دریافت پاسخ محیط، بردار احتمال $\hat{P}(n)$ اقدامهای خود را در صورت دریافت پاسخ مطلوب بر اساس رابطه (۴) و در صورت دریافت پاسخ نامطلوب طبق رابطه (۵) بروز می نماید، سپس اتماتای یادگیر بردار احتمال اقدامهای خود $P(n)$ را با استفاده از بردار $\hat{P}(n+1)$ و طبق رابطه (۶) بروز می کند.

$$K(n) = \sum_{\alpha_i \in V(n)} p_i(n) \quad (2)$$

$$\hat{P}_i(n) = prob[\alpha(n) = \alpha_i | \alpha_i \in V(n)] = \frac{p_i(n)}{K(n)} \quad (3)$$

که در فرمول بالا منظور از $V(n)$ تعداد اقدامهای فعال می باشد.

$$\hat{P}_i(n+1) = \hat{P}_i(n) + a \cdot (1 - \hat{P}_i(n)) \quad (4)$$

$$\hat{P}_j(n+1) = \hat{P}_j(n) - a \cdot \hat{P}_j(n) \quad \forall j \neq i \quad (5)$$

$$\hat{P}_i(n+1) = \hat{P}_i(n) + (1-b) \cdot (1 - \hat{P}_i(n)) \quad (5)$$

$$\hat{P}_j(n+1) = \frac{b}{r-1} + (1-b)\hat{P}_j(n) \quad \forall j \neq i \quad (6)$$

$$P_i(n+1) = \hat{P}_i(n+1) \cdot K(n) \quad (6)$$

$$P_j(n+1) = P_j(n) \quad j, \alpha_j \notin V(n)$$

کد این الگوریتم بصورت زیر می باشد. مقدار بیشتر برای $E_{\alpha}^{cur_page}$, نشان دهنده ارتباط بیشتر دو سند α و cur_page می باشد و بالعکس مقدار کمتر این معیار نشان دهنده ارتباط کمتر این دو سند است.

Proposed_DLA_Algorithm:

Variables: **route** : Array of [number of users][user's path] that is obtained from Liu model
L[number of user]: Array of [number of pages][number of pages].
DLA: distributed learning automata which contains n LA having n-1 actions.
 Begin
 For all users do
 Step_id=1;
 Cur_page=route[user_id][Step_id];
 While (Step_id<=Max_forage & S<Maxthd & S >Minthd) do
 If (Dist(Cur_page, α)<Threshold)
 If $A_{\alpha}^{cur_page} == 0$ then
 $A_{\alpha}^{cur_page} = 1;$
 End // If
 If (L[user_id](Cur_page, α)==1)
 Set $\beta = 0$
 $E_{\alpha}^{cur_page} = -(\hat{P}_{\alpha}^{cur_page} \log \hat{P}_{\alpha}^{cur_page} + (1 - \hat{P}_{\alpha}^{cur_page}) \log (1 - \hat{P}_{\alpha}^{cur_page}))$
 $a_{\alpha} = \frac{E_{\alpha}^{cur_page}}{E_{\alpha}^{cur_page} + 1}$
 $\hat{P}_{\alpha}^{cur_page}(n+1) = \hat{P}_{\alpha}^{cur_page}(n) + a_{\alpha}(1 - \hat{P}_{\alpha}^{cur_page}(n))$
 $\hat{P}_j^{cur_page}(n+1) = \hat{P}_j^{cur_page}(n) - a_{\alpha}\hat{P}_j^{cur_page}(n) \quad \forall j \neq \alpha$
 Else
 $\hat{P}_{\alpha}^{cur_page}(n+1) = \hat{P}_{\alpha}^{cur_page}(n)$
 $\hat{P}_j^{cur_page}(n+1) = \hat{P}_j^{cur_page}(n) \quad \forall j \neq \alpha$
 End // If
 Step_id=Step_id+1;
 Cur_page= α ;
 Else
 $p_{\alpha}^{cur_page}$ probability is divided between other Cur_page's Actions.
 $p_{\alpha}^{cur_page} = 0;$
 End // If
 End // While
 End // For
 End

شکل ۲: شبه کد الگوریتم پیشنهادی

۴- نتایج آزمایشها

در این بخش ابتدا مدل شبیه سازی و سپس نتایج شبیه سازی الگوریتم پیشنهادی و مقایسه آنها با الگوریتم های موجود ارائه می شوند.

زیرمجموعه ای از اقدامهایش فعال و می تواند قابل استفاده باشد [۱۰]. در روش پیشنهادی، در ابتدای کار تمامی اقدامهای اتوماتای یادگیر در اتوماتای یادگیر توزیع شده غیر فعال می باشند. با شروع حرکت کاربر از سند، اتوماتای منتظر با آن سند در اتوماتای یادگیر توزیع شده فعال می شود سپس اتوماتای فعال شده در اتوماتای یادگیر توزیع شده، یک اقدام را بطور تصادفی انتخاب نموده و در صورتیکه فاصله اقلیدسی بین سند منتظر با اتوماتای α و سند منتظر با اقدام انتخابی بیشتر از یک مقدار آستانه اتصال باشد آن اقدام حذف شده و احتمال آن بین بقیه تقسیم می شود. در صورتیکه فاصله اقلیدسی بین سند منتظر با اتوماتای α و سند منتظر با اقدام انتخابی کمتر از یک مقدار آستانه اتصال باشد آنگاه اگر ارتباط بین این دو سند در لایه فایل، به ازای کاربر مورد نظر موجود باشد، پاسخ محیط مطلوب بوده و اتوماتای α به اقدام انتخابی خود پاداش می دهد، در غیر اینصورت پاسخ محیط نامطلوب بوده و اتوماتای α باید اقدام انتخابی را به میزان پارامتر b طبق الگوریتم یادگیری جریمه نماید، بدلیل استفاده از الگوریتم یادگیری L_{RI} ، مقدار b صفر می باشد، همچنین ضریب پاداش دریافتی توسط اقدام α ، بر اساس مفهوم آنتروپی، بصورت a_{α} ، تعریف می شود. آنتروپی تعیین می کند که آیا پروفایل علاقه هر کاربر متوازن است یا نه. پروفایل علائق کاربران نشان دهنده رفتار آنها در جستجو می باشد. فرق الگوریتم پیشنهادی با الگوریتمهای پیشین مبتنی بر اتوماتای یادگیر توزیع شده، در نحوه عملکرد اتوماتاهای یادگیر در اتوماتای یادگیر توزیع شده می باشد. در تمام الگوریتمهای موجود، هر اتوماتا برای انتخاب عمل خود به لایه فایل کاربران نگاه می کند و در هر مرحله همان اقدامهایی را فعال می کند که کاربر از آنها عبور می کند. بعارت دیگر نحوه عملکرد اتوماتای تخصیص داده شده به هر سند بر اساس اطلاعات نحوه حرکت کاربر موجود در آن سند می باشد، بنابراین در تمام الگوریتمهای موجود، عملکرد انتخاب اقدامهای اتوماتا قابل پیش بینی و بر اساس مدل حرکات کاربر است، اما در الگوریتم پیشنهادی اتوماتای یادگیر در هر لحظه بصورت تصادفی اقدام خود را انتخاب نموده و عمل بروزرسانی مقدار احتمال این اقدام را بر اساس مراجعة به لایه فایل و مقایسه با آن، انجام می دهد. شبه

$$Corr(P, P') = \frac{\sum PP' - (\sum P)(\sum P')}{\sqrt{(\sum P^2 - (\sum P)^2)/N}(\sum P' - (\sum P')^2/N)} \quad (12)$$

جدول ۱: پارامترهای شبیه سازی

۰.۷	حد آستانه اتصال
۲۰	تعداد اسناد
۲۰۰۰	تعداد کاربران
۵	تعداد موضوعات
۰.۲	مقدار ثبات سند اولیه در موضوعات مختلف T_c
-	ضریب ثابت کاهش اشتیاق کاربر ΔM_t^c
-	ضریب متغیر کاهش اشتیاق کاربر ΔM_t^v
۱	پارامتر توزیع قانون-توانی توزیع احتمال α_p علاقه کاربران
۱.۲	ضریب پاداش دریافتی از مشاهده یک سند λ
۰.۵	ضریب جذب اطلاعات از یک سند توسط یک کاربر μ_m میانگین توزیع نرمال
۰.۹۷	ΔM_t^v
۰.۲۵	σ_m واریانس توزیع نرمال
-	μ_m میانگین توزیع نرمال برای مقدار افزایش یک گره برای یک موضوع خاص
۳	پارامتر توزیع قانون-توانی توزیع احتمالهای وزنهای مطلب برای هر سند σ واریانس توزیع نرمال برای مقدار افزایش یک گره برای یک موضوع خاص
۰.۲۵	ضریب کاهش علاقه کاربر θ
۰.۲	حداقل اشتیاق کاربر برای ادامه جستجو

۴- ارزیابی کارایی الگوریتمهای یادگیری

قبل از مقایسه الگوریتم پیشنهادی تعیین ساختار با سایر روش‌های موجود، به بررسی و ارزیابی کارایی الگوریتمهای یادگیری استاندارد L_{RP} و L_{RI} ، می‌پردازیم. این آزمایش بمنظور بررسی تأثیر مقادیر پارامتر یادگیری بر روی تعداد اسناد انجام شد. به ازای تعداد اسناد حداکثر مقدار کورولیشن و تعداد تکرارهای لازم برای رسیدن به این سطح از کورولیشن، برای الگوریتم یادگیری L_{RI} ، در جدول ۲ نشان داده شده است.

۱-۴- مدل شبیه سازی

برای ارزیابی الگوریتم پیشنهادی، از مدل معرفی شده در [۱۱]، برای نشان دادن ساختار وب و چگونگی استفاده کاربران از وب، استفاده می‌شود. اعتبار این مدل توسط Liu و همکاران با استفاده از اطلاعات بدست آمده از چندین وب سایت بزرگ مانند مایکروسافت تأیید شده است. بر این اساس، در این مقاله پروفایل علاقه کاربران بصورت توزیع قانون-توانی و توزیع محتوای اسناد وب، بصورت توزیع نرمال در نظر گرفته شده است. سایر پارامترهای استفاده شده در مدل [۱۱] برای شبیه سازیهای انجام شده در این مقاله در جدول ۱ نشان داده شده است. در این مدل، هر سند وب با یک بردار محتوا نمایش داده می‌شود. طول این بردار برابر با تعداد موضوعات موجود در سیستم است. هر عضو این بردار، میزان ارتباط سند متناظر با آن بردار را با یکی از اسناد نشان می‌دهد. با استفاده از بردار محتوا دو سند نوی فاصله اقلیدسی بین دو سند ($d_{i,j}$)، طبق رابطه (۱۰) محاسبه می‌شود. در این مقاله معیار شباهت دو سند در حالت ایده آل، بصورت عکس فاصله اقلیدسی این دو سند تعریف شده است. برای ارزیابی الگوریتم پیشنهادی، با سایر روش‌ها، از معیاری به نام کورولیشن استفاده می‌کنیم. کورولیشن، میزان همبستگی خطی بین داده‌ها را مشخص می‌کند. کورولیشن دو مجموعه داده P و P' طبق رابطه (۱۲) محاسبه می‌شود. در این رابطه، N تعداد داده هاست. مجموعه P ساختار ایجاد شده در حالت ایده آل توسط مدل شبیه سازی و P' ساختار ایجاد شده توسط الگوریتم پیشنهادی، است.

$$C_n = \{cw_n^1, cw_n^2, \dots, cw_n^M\} \quad (7)$$

$$p = \{p_{ij} | i, j = 1, 2, 3, \dots, n, \quad i \neq j\} \quad (8)$$

$$p_{ij} = \frac{(d_{ij})^{-1}}{\sum_{k=1}^{k=n} (d_{ik})^{-1}} \quad (9)$$

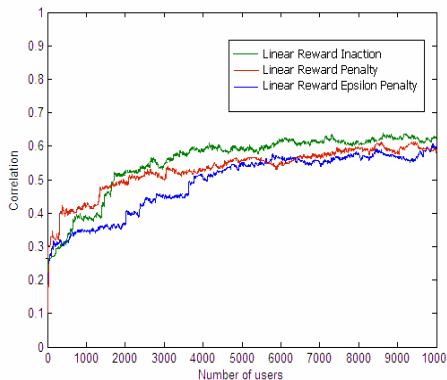
$$d_{ij} = \sqrt{\sum_{k=1}^{k=M} (cw_i^k - cw_j^k)} \quad (10)$$

$$p' = \{p'_{ij} | i, j = 1, 2, 3, \dots, n, \quad i \neq j\} \quad (11)$$

p'_{ij} = احتمال انتخاب اقدام j ام از اتمماتای i است، در صورتیکه این اقدام غیر فعال باشد مقدار این احتمال صفر می‌باشد.

۱۰۰۰	۵۰۰۰	۲۰۰۰	۱۰۰۰	التوریم پادگیری
۰,۶۱۸۲	۰,۵۵۴	۰,۵۰۳	۰,۳۷۵	L_{RI}
۶۶۱۷	۴۴۷۰	۱۹۹۹	۹۴۱	
۰,۵۸۳	۰,۵۷۲	۰,۴۲۷	۰,۳۶۵	L_{REP}
۸۳۷۱	۴۸۴۵	۱۵۶۰	۹۰۴	
۰,۵۹۴	۰,۵۴۳	۰,۴۲۵	۰,۴۰	L_{RP}
۶۲۰۴	۴۶۲۹	۱۵۷۷	۹۲۳	

همانگونه که در جدول بالا مشاهده می شود با افزایش تعداد مراحل یادگیری (تعداد کاربران)، میزان کورولیشن به ازای هر یک از الگوریتمهای یادگیری افزایش می یابد. به منظور ارزیابی کارایی یادگیری L_{RP} , L_{REP} و L_{RI} در نحوه بروز رسانی بردار احتمالات اتوماتاهای یادگیر و تعیین ساختار ارتباطی بین اسناد، شبکه شامل ۲۱ سند در نظر گرفتیم و طول بردار اعمال اتوماتاهای یادگیر را برابر با تعداد اسناد تنظیم نمودیم. کورولیشن بین ساختار ارتباطی بدست آمده از هر الگوریتم یادگیری و ساختار ارتباطی ایده آل را تا تکرار ۱۰۰۰۰ محسوبه نمودیم. در انجام محاسبات مقدار پارامتر های یادگیری a و b را بترتیب $0,0,5$ و $0,0,1$ تعریف نمودیم. نتایج آزمایش در شکل ۳ نشان داده شده است و مشخص می کند که الگوریتم یادگیری L_{RI} در مقایسه با دو الگوریتم یادگیری دیگر دارای سطح کورولیشن بالاتری است، بعارت دیگر ساختار ارتباطی بدست آمده با استفاده از این الگوریتم یادگیری، به ساختار ارتباطی ایده آل نزدیکتر می باشد.



شکل ۳: مقایسه کورولیشن الگوریتمهای یادگیری

به ازای تعداد اسناد ۲۱ و مراحل یادگیری ۱۰۰۰

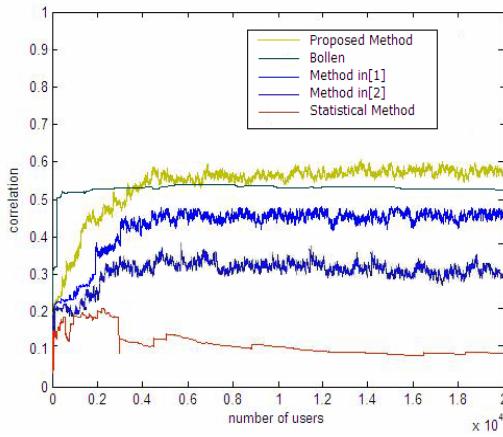
جدول ۲: حداکثر مقدار کورولیشن بدست آمده و تعداد مراحل

یادگیری الگوریتم یادگیری L_{RI} برای استناد مختلف به ازای تعداد کاربران ۵۰۰۰

تعداد استناد						
پارامتر یادگیری						
۰,۱	۰,۱	۰,۱	۰,۱	۰,۱	۰,۱	$a = 0,001$
-۰,۱۷	-۰,۱۸	-۰,۲۲	-۰,۲۶	-۰,۲۹	-۰,۳۱	
۰,۴۹	۰,۴۹۳	۰,۴۹۳	۰,۴۹۶	۰,۴۹۳	۰,۴۹۷	
۰,۷	۰,۷۲	۰,۷۴	۰,۷۹	۰,۷۹	۰,۷۹	$a = 0,01$
۰,۹۰	۰,۹۱	۰,۹۱	۰,۹۱	۰,۹۱	۰,۹۱	
-۰,۱۹	-۰,۱۹	-۰,۲۶	-۰,۲۶	-۰,۲۶	-۰,۲۶	$a = 0,05$
۰,۴۹۷	۰,۴۹۹	۰,۴۹۹	۰,۴۹۱	۰,۴۷۷	۰,۴۷۶	
-۰,۱۸	-۰,۲۲	-۰,۲۷	-۰,۳۲	-۰,۳۳	-۰,۳۴	$a = 0,07$
۰,۹۱	۰,۹۱	۰,۹۱	۰,۹۱	۰,۹۱	۰,۹۱	
-۰,۱۷	-۰,۱۹	-۰,۲۷	-۰,۳۰	-۰,۳۰	-۰,۳۰	$a = 0,1$
۰,۴۹۰	۰,۴۹۰	۰,۴۹۰	۰,۴۹۰	۰,۴۹۰	۰,۴۹۰	

همانگونه که در جدول ۲ نشان داده شده است با افزایش تعداد اسناد وب، و با ثابت نگه داشتن حداکثر تعداد مراحل یادگیری، میزان رسیدن به حداکثر کورولیشن کاهش می یابد. آن به این دلیل است که با افزایش تعداد اسناد، طول بردار اعمال اتوماتا افزایش یافته و میزان همگرایی کاهش می یابد. به همین دلیل لازم است که پارامتر یادگیری مناسب را به ازای افزایش تعداد اسناد وب تنظیم نمود، منظور از پارامتر یادگیری مناسب، پارامتر یادگیری می باشد که در ازای آن میزان کورولیشن ساختار ارتباطی و ساختار ایده آل در کمترین تکرار، بیشترین مقدار دارد. نتایج آزمایش نشان می دهد که مقدار پارامتر یادگیری رابطه عکس با تعداد اسناد (طول بردار اعمال اتوماتا) دارد. عبارت دیگر همانطور که در جدول ۲ مشاهده می شود با افزایش تعداد اسناد و با تنظیم مقدار کمتر برای پارامتر یادگیری به حداکثر سطح کورولیشن می رسیم. بنابراین بمنظور رسیدن به مقدار حداکثر کورولیشن، لازم است که دو عامل پارامتر یادگیری و تعداد مراحل یادگیری را بطور مناسب انتخاب نماییم. برای بررسی تأثیر تعداد مراحل یادگیری در میزان کورولیشن آزمایش دیگری انجام گرفت، نتایج این آزمایش در جدول ۳ نشان داده شده است. در این آزمایش تعداد اسناد را ۲۱ پارامتر یادگیری a و b را بترتیب $0,0,5$ و $0,0,1$ در نظر گرفتیم.

جدول ۳: تأثیر تعداد مراحل یادگیری در حداکثر مقدار کورولیشن بدست آمده به ازای هر یک از الگوریتمهای یادگیری



شکل ۴: مقایسه الگوریتم پیشنهادی با سایر روش‌های گزارش شده

به منظور بررسی تأثیر تعداد اسناد وب و تعداد کاربران بر روی کارایی الگوریتم پیشنهادی و الگوریتم ارائه شده در [۱] (از لحاظ کارایی بهتر از سایر روش‌های موجود مبتنی بر اتوماتیک یادگیر عمل می‌کند)، آزمایش دیگری انجام دادیم. در این آزمایش، حداقل مقدار کورولیشن بدست آمده از دو روش ارائه شده در [۱] و روش پیشنهادی، معیار ارزیابی ما می‌باشد. همانطور که در جداول ۴ و ۵ دیده می‌شود، الگوریتم پیشنهادی، همواره دارای کورولیشن بالاتری نسبت به الگوریتم ارائه شده در [۱] است. با افزایش تعداد اسناد، کورولیشن در هر دو الگوریتم کاهش می‌یابد، دلیل آن این است که با افزایش تعداد اسناد، طول بردار اعمال اتوماتا افزایش می‌یابد (جدول ۴). هرچه تعداد کاربران بیشتر می‌شود، انتظار می‌رود که به سطح کورولیشن بهتری دست یابیم، این مسأله در جدول ۵ مشاهده می‌شود.

جدول ۴: حداقل کورولیشن بدست آمده برای روش‌های مختلف به ازای تعداد کاربران ۲۰۰۰۰ و تعداد اسناد ۳۰، ۲۰ و ۱۰

تعداد اسناد					روش
۴۰	۳۰	۲۰	۱۰		روش پیشنهادی
۰.۵۳۷۸	۰.۵۶۳۱	۰.۶۱۲۱	۰.۷۹۵۳		
۰.۴۳۸۲	۰.۴۸۰۲	۰.۵۳۸۷	۰.۶۸۸۷	[۱]	روش ارائه شده در [۱]

جدول ۵: حداقل کورولیشن بدست آمده برای روش‌های مختلف به ازای تعداد اسناد ۲۰ و تعداد کاربران ۲۰۰۰۰، ۳۰۰۰۰ و ۴۰۰۰۰

۴-۳-۴- ارزیابی کارایی الگوریتم پیشنهادی تعیین ساختار

الگوریتم یادگیری استفاده شده در روش‌های تعیین ساختار برای بروزرسانی بردار احتمالات اوتماتاهای یادگیر، L_{RI} می‌باشد. شکل ۴، مقایسه الگوریتم پیشنهادی را با الگوریتم ارائه شده در [۱]، [۲]، روش بولن و روش آماری مطرح شده در [۴]، نشان می‌دهد. در این آزمایش تعداد کاربران ۲۰۰۰۰، تعداد موضوعات مرتبط با هر سند ۵ و تعداد اسناد وب ۲۰ در نظر گرفته شده است. الگوریتم پیشنهادی برای تعیین ساختار اسناد وب، بهتر از سایر الگوریتم‌های ارائه شده مبتنی بر اتوماتیک یادگیر، عمل می‌کند، بعلت اینکه دو عامل مهم را در مورد آن لحاظ کرده ایم. اول اینکه تعداد اقداماتی هر اتوماتیک یادگیر مطابق با [۱۰]، متغیر در نظر گرفته شده و این باعث می‌شود که در هر لحظه فقط زیر مجموعه‌ای از اقداماتی فعال یک اتوماتا، بروزرسانی شوند. دوم اینکه، نحوه بروزرسانی الگوریتم یادگیری بر اساس مفهوم آنتروپی [۱]، می‌باشد. ترکیب این دو عامل باعث می‌شود، نسبت به الگوریتمی که طول بردار اعمال آن برابر با تعداد اسناد وب است و در هر لحظه کل بردار اعمال بروز می‌گردد [۱]، سریعتر همگرا شود. همچنین، روش ارائه شده به دلیل اینکه از مفهوم آنتروپی استفاده می‌کند نسبت به روش مبتنی بر اتوماتیک یادگیر با اقداماتی متغیر، با افزایش تعداد اسناد، نیاز به تنظیم پارامتر یادگیری ندارد. به منظور نمایش یک کران پایین برای کارایی این دو الگوریتم، از یک روش آماری ساده، استفاده شده است. در این روش آماری، شباهت دو سند را به صورت نسبت تعداد دفعاتی که کاربران از سند i به سند j رجوع کرده اند ($N(i,j)$ ، به تعداد دفعاتی که کاربران از سند i به هر سند دیگری مراجعه کرده اند، محاسبه می‌شود).

$$\text{Similarity}(i, j) = \frac{N(i, j)}{\sum_{k=1}^n N(i, k)}$$

- [4] Meybodi, M.R. and Beigy, H., "Solving Stochastic Path Problem Using Distributed Learning Automata," Proceedings of The Sixth Annual International CSI Computer Conference, CSIC2001, Isfahan, Iran, pp. 86-70, Feb. 20-22, 2001.
- [5] Beigy, H. and Meybodi, M.R., "A New Distributed Learning Automata Based Algorithm For Solving Stochastic Shortest Path Problem", Proceedings of the Sixth International Joint Conference on Information Science, Durham, USA, pp. 339- 343, 2002.
- [6] Alipour, M. and Meybodi, M.R, "Solving Dynamic Traveling Salesman Problem Using Responsive Distributed Learning Automata", Proceedings of the Second International Conference on Information and Knowledge Technology (IKT2005), Tehran, Iran, May 24-26, 2005.
- [7] Alipour, M. and Meybodi, M.R., "Solving Probabilistic Traveling Sales Man Problem Using Distributed Learning Automata", Proceedings of 11th Annual CSI Computer Conference of Iran, Fundamental Science Research Center (IPM), Computer Science Research Lab., Tehran, Iran, pp. 673- 678, Jan. 24-26, 2006.
- [8] Alipour, M. and Meybodi, M.R., "Solving Maximal independent Set Problem Using Distributed Learning Automata", 14th Iranian Electrical Engineering Proceedings of (ICEE2006), Amirkabir University, Tehran, Iran, May 16-18, 2006.
- [9] Narendra, K.S. and Thathachar, M.A.L., "Learning Automata: An Introduction", Prentice Hall, 1989.
- [10] Thathachar, M.A.L. and Harita Bhaskar, R., "Learning Automata with Changing Number of Actions", IEEE Transactions on Systems Man and Cybernetics, vol. 17, no.6, pp. 1095-1100, Nov. 1987.
- [11] Liu, J., Zhang, S. and Yang, J., "Characterizing Web Usage Regularities with Information Foraging Agents", IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 4, pp. 566-584, April 2004.
- [12] Heylighen, F. and Bollen, J., "Hebbian Algorithms for a Digital Library Recommendation System", Proceedings of the International Conference on Parallel Processing Workshops (ICPPW'02), 2002, pp. 439-446.
- [13] Teles, W., Weigang, L., and Ralha, C., "AntWeb-The Adaptive Web Server Based on the Ants' Behavior", Proceeding of IEEE/WIC international Conference on Web Intelligence (WI'03), PP. 558-564, 2003.
- [14] Jianhan, Zhu., "Mining Web Site Link Structures for Adaptive Web Site Navigation and Search", Ph.D Thesis, University of Ulster at Jordanstown, October 2003.

تعداد کاربران	روش	روش پیشنهادی	روش ارائه شده در [1]	روش پیشنهادی	تعداد کاربران
۴۵۰۰۰	۳۵۰۰۰	۲۰۰۰۰	۱۰۰۰۰	۰۷۱۲۴	۰۶۸۴۳
۰۶۱۲۱	۰۶۰۲۳	۰۵۶۲۱	۰۵۰۲۳	۰۵۰۴۸	۰۵۴۹۰

نتیجه‌گیری

در این مقاله یک روش جدید مبتنی بر اتماتای یادگیر توزیع شده برای کشف ساختار ارتباطی بین اسناد وب پیشنهاد گردید الگوریتم پیشنهادی، بدلیل استفاده از اتماتاهای یادگیر با تعداد اقدامهای متغیر و بروز رسانی پارامتر یادگیری با افزایش تعداد اسناد وب، دارای سه مزیت می باشد که عبارتند از: ۱- به دلیل استفاده از اتماتاهای یادگیر با تعداد اقدامهای متغیر، الگوریتم پیشنهادی، نسبت به روشی که تعداد اقدامهای آن برابر با تعداد اسناد می باشد، با افزایش تعداد اسناد وب، دارای کوچولیش بالاتر است ۲- عدم نیاز به تنظیم مجدد پارامترهای یادگیری اتماتاهای یادگیر، با تغییر تعداد اسناد وب، به دلیل استفاده از مفهوم آنتروپی ۳- قابلیت استفاده بر خط. نتایج شبیه سازیها نشان می دهند روش پیشنهادی به دلیل عملکرد کاملاً تصادفی اتماتاهای درانتخاب اقدام بعدی و دادن پاداش از طریق مقایسه سند منتظر این اقدام با سند انتخابی کاربر در لای فایل، متفاوت از سایر روشهای موجود مبتنی بر اتماتای یادگیر می باشد و نسبت به آنها دارای کارایی بالاتر است. از ساختار ارتباطی بدست آمده می توان بمنظور تعیین شباهت بین اسناد وب، رتبه بندی، خوشه بندی اسناد و کشف زیر گرافهای متصل وب استفاده نمود.

مراجع

- [1] بابک اناری و محمد رضا مبدی، "یک روش مبتنی بر اتماتای یادگیر توزیع شده برای تعیین ساختار اسناد وب"، مجموعه مقالات دوازدهمین کنفرانس بین المللی انجمن کامپیوتر ایران، تهران، ایران، ۱۳۸۵.
- [2] Saati, s. and Meybodi, M.R., "A Self Organizing Model for Document Structure Using Distributed Learning Automata", Proceedings of the Second International Conference on Information and Knowledge Technology (IKT 2005), Tehran, Iran, May 24-26, 2005.
- [3] Hashemi, A.B. and Meybodi, M.R., "Web Usage Mining Using Distributed Learning Automata", Computer Engineering Department, Technical Report, 2005.