کاوش استفاده از وب با استفاده از کلونی مورچه‌ها

علی برادران هاشمی محمدرضا میبدی سعید شیری قیداری

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

دانشگاه صنعتی امیرکبیر، تهران، ایران

{ a_hashemi, mmeybodi, shiry}@aut.ac.ir

چکیده

یکی از مسایل مطرح در داده‌کاوی وب، تعیین میزان شباهت اسناد با یکدیگر از طریق اطلاعات درباره چگونگی استفاده کاربران از وب می‌باشد. در این مقاله روشی مبتنی بر کلونی مورچه‌ها که از اطلاعات چگونگی استفاده کاربران از وب استفاده می‌کند به منظور تشخیص شباهت صفحات وب پیشنهاد می‌گردد. این روش بر این ایده استوار است که اگر تعدادی از کاربران تعدادی از صفحات وب را پی در پی درخواست کنند، احتمالا این صفحات به نیازهای اطلاعاتی یکسانی پاسخ داده‌اند و در این صورت با همدیگر شباهت دارند. در روش پیشنهادی که بر اساس رفتار کلونی مورچه‌ها عمل می کند به هر کاربر یک مورچه تخصیص داده می شود. این مورچه با دنبال کردن مسیر کاربر سعی در یادگیری صفحات مشابه دارد. از نتایج حاصل از این روش می‌توان برای ارائه صفحات پیشنهادی مشابه با یک صفحه بر اساس علایق یک یا چند کاربر و یا خوشه‌بندی صفحات مشابه استفاده نمود. نتایج شبیه‌سازیها نشان داده است که روش پیشنهادی در مقایسه با روش هب و دو روش گزارش شده مبتنی بر اتوماتای توزیع شده، در تشخیص شباهت صفحات از کارایی بالاتری برخوردار است.

کلمات کلیدی: کاوش استفاده از وب، کلونی مورچه‌ها.

۱ مقدمه

تشخیص شباهت بین اسناد یک مجموعه یکی از اهداف روشهای بازیابی اطلاعات می‌باشد. از اطلاعات در باره شباهت بین اسناد (صفحات وب) می‌توان برای ارائه اسناد مشابه به کاربران به منظور یافتن اطلاعات مورد نظر خود استفاده کرد. روش‌های متعددی برای تشخیص شباهت بین اسناد وجود دارد. قدیمی‌ترین روش، استفاده از نظر یک فرد خبره می‌باشد. این روش معمولا با دسته‌بندی اسناد بر اساس طبقه‌بندی موضوعی انجام می‌شود. استفاده از کلمات کلیدی در مقالات علمی یا صفحات وب برای یافتن شباهت بین اسناد نیز میتواند استفاده شود. استفاده از کلمات کلیدی دارای مشکلاتی مانند وجود کلمات مترادف[۱] (کلماتی با ظاهر متفاوت ولی معنای یکسان)، کلمات متشابه[۲] (کلماتی با ظاهر یکسان ولی در معنی متفاوت) میباشد. علاوه بر این زمانیکه که موضوعی برای کاربر با موضوع جدیدی روبرو میشود، پیدا کردن کلمات کلیدی مناسب کاری مشکل میباشد و تنها توسط افرادی که به زمینه سند آشنایی دارند می‌تواند استخراج شود. این مشکل در مورد اسناد الکترونیکی مانند صفحات وب، با استفاده از روشهای بازیابی اطلاعات تا اندازه زیادی کاهش یافته است. بعنوان مثال با استخراج کلمات اصلی یک متن، کلمات کلیدی آن را مشخص می‌کنند [15]. البته این روشها برای اسناد غیر متنی مانند تصاویر، فیلمها و اسناد صوتی کمتر استفاده شده است. از دیگر روشهای سنتی برای تعیین ارتباط اسناد مانند مقالات علمی با یکدیگر استفاده از اطلاعات در باره مراجع هر مقاله و بررسی ارتباط آنها از لحاظ کتابشناسی می‌باشد [11].

اکثر تحقیقات انجام شده در زمینه داده‌کاوی بر اساس تحلیل محتوای اسناد (داده‌کاوی محتوا[۳]) و یا ساختار گراف ارتباط اسناد (داده‌کاوی ساختار[۴]) بوده است. علاوه بر اطلاعات بدست آمده از این دو روش، می‌توان از اطلاعات در باره رفتار کاربران (با استفاده از فایلهای ثبت وقایع[۵] در سرویس‌دهنده‌های وب یا برنامه‌های در سمت کاربر) برای تعیین ارتباط بین اسناد [8]، پیشنهاد صفحات[3][16][4]، تغییر ساختار سایت وب

[17]، شخصی کردن سرویس‌هایی مانند وب [13][14][7]، بهینه‌سازی موتورهای جستجو [9] استفاده کرد. در [18] کاربردهای اطلاعات استفاده از سیستم بطور مفصل ارائه شده است.

در تعدادی از روشهای گزارش شده مانند روش گزارش شده در [16] با توجه به بازخوردهای ارائه شده توسط کاربران، سیستم به صفحات وب امتیازاتی می‌دهد که این امتیازات برای پیشنهاد صفحات به کاربر مورد استفاده قرار می‌گیرد. استفاده از بازخورد کاربران موجب ایجاد وظیفه‌ای ناخواسته برای کاربران شده و باعث نارضایتی آنها می‌شود. در بعضی از روشها از اطلاعات ثبت شده مشخص برای هر کاربر استفاده می‌شود. بعنوان مثال amazon.com بعنوان یک سایت فروش الکترونیکی بر روی وب، با استفاده از اطلاعات در باره خرید کاربران، ممکن است به کاربری که می‌خواهد جنس a را بخرد، پیشنهاد خرید جنس b را نیز بدهد چرا که کاربرانی که جنس a را خریده‌اند، معمولا جنس b را نیز خریده‌اند. هرچند، استفاده از اینگونه روشها به دلایلی مانند مسائل مرتبط با حریم شخصی کاربران یا محدودیتهای سرویس‌دهنده‌های وب امکان‌پذیر نمی‌باشد. بهمین دلیل معمولا در روشهای داده‌کاوی اطلاعات استفاده از وب از فایلهای ثبت وقایع در سرویس‌دهنده‌های وب (که تنها اطلاعات درخواستهای کاربران را در بر دارند) استفاده می‌شود [12][17]. از آنجاییکه استفاده از اطلاعات وب بصورت ناشناس صورت می‌گیرد، استفاده از چنین روشی تنها برای سایتهای محدودی امکان‌پذیر است. بهمین علت معمولا در چنین سیستم‌هایی از فایلهای ثبت وقایع در سرویس‌دهنده‌های وب و بدون دسترسی به اطلاعات شخصی هر کاربر استفاده می‌شود[12][17].

در [8] رویکرد جدیدی برای مساله *داده‌کاوی اطلاعات استفاده از وب*[۶] ارائه شده است. ایده این روش بر این اساس است که اگر دو سند به یک نیاز اطلاعاتی پاسخ دهند، آنگاه آن دو سند مشابه می‌باشند. در این روش فرض بر این است که کاربران از محتویات سندی که می‌خواهند آنرا در گام بعدی خود انتخاب کنند آگاهی نسبی دارند و بر اساس نیاز اطلاعاتی خود سند بعدی را انتخاب می‌کنند و حرکت کاربران در بین اسناد اتفاقی نیست. در واقع کاربر با استفاده از اطلاعات خود ارتباطی مجازی بین اسناد ایجاد کرده و آنها را مشاهده می‌کند. این ارتباط لزوما منطبق بر ارتباطات قابل مشاهده اسناد (مانند ارتباط اسناد بر اساس کلمات کلیدی تعریف شده یا ارتباطات کتابشناسی) نمی‌باشد بلکه می‌تواند برگرفته از مدل ذهنی کاربر باشد. از آنجاییکه فرض شده است که کاربر اطلاعات کافی در مورد اسناد مشاهده شده دارند، بنابراین انتظار می‌رود که اسناد مشابه در یک موضوع با یکدیگر مورد استفاده قرار گیرند. در این روش، با تحلیل داده‌های استفاده، بدون تلاش مضاعف کاربر یا افراد خبره (مانند کتابداران)، اطلاعات با ارزشی بدست می‌آید. در روش فوق ارتباطات بین اسناد با استفاده از روشی مانند قانون هب [19] اصلاح می‌گردد. به این صورت که با حرکت کاربر از سند i به سند j، تنها اتصال بین این دو سند ($a(i, j)$) تقویت می‌شود. که تقویت اتصال دو سند i و j متناظر با افزایش میزان شباهت این دو سند در نظر گرفته شده است. در نسخه توسعه یافته این الگوریتم، با حرکت کاربر از سند i به سند j، نه تنها اتصال بین این دو سند تقویت می‌شود، بلکه با در نظر گرفتن رابطه تراگذری، اتصال سند i به سندهای دیگری که کاربر بعد از مشاهده سند j در ادامه مسیر خود مشاهده می‌کند، با در نظر گرفتن یک ضریب کاهش(b)، تقویت می‌گردد.

با استفاده از ایده مطرح شده در بالا، در [۱] یک روش خودسازمانده مبتنی بر اتوماتای یادگیر توزیع‌شده برای تعیین شباهت اسناد در یک کتابخانه دیجیتال ارائه شده است. در این روش یک اتوماتای یادگیر توزیع‌شده متناظر با گراف ارتباطات اسناد کتابخانه دیجیتال در نظر گرفته می‌شود. بصورتی که هر اتوماتای یادگیر در اتوماتای یادگیر توزیع شده دارای تعدادی محدودی اقدام می‌باشد و هر اقدام متناظر با یک سند در مجموعه اسناد است. در این روش تنها اسنادی که در مسیر مستقیم حرکت کاربر از سند آغازین تا آخرین سند مشاهده شده قرار دارند، مشابه در نظر گرفته می‌شوند. بر این اساس پس از خروج هر کاربر از سیستم، با بررسی مسیر حرکت او، به اقدامهای اتوماتای یادگیر متناظر با اسنادی که در مسیر حرکت کاربر از نخستین صفحه تا آخرین صفحه قرار داشته‌اند پاداش و اقدامهای متناظر با اسنادی که قسمتی از یک دور هستند، جریمه می‌شوند.

در [۲] نیز با استفاده از اتوماتای یادگیر توزیع‌شده روشی برای تعیین شباهت اسناد وب ارائه شده است. در این روش مانند[۱] گراف اتوماتای یادگیر توزیع‌شده متناظر با گراف ارتباطات اسناد می‌باشد. با این تفاوت که در این روس متناظر با هر سند یک اتوماتای یادگیر با تعداد اقدامهای متغیر درنظر گرفته می‌شود. در ابتدای فعالیت این اتوماتا، همه اقدامهای آن

غیر فعال هستند. با مشاهده پیوسته دو سند i و j اقدام jام از اتوماتای I فعال شده و پاداش می‌گیرد. در این روش شباهت دو سند I و j برابر با احتمال انتخاب اقدام jام از اتوماتای I در نظر گرفته شده است. نتایج شبیه‌سازی‌ها نشان داده است که دقت این روش در تشخیص شباهت اسناد بهتر از روش خودسازمانده [۱] و روش هب می‌باشد.

در این مقاله، الگوریتمی با استفاده از ایده مطرح شده در [8] و [20] ارائه شده است که با استفاده از کلونی مورچه‌ها بدنبال تعیین شباهت اسناد می‌باشد. در الگوریتم پیشنهادی به هر کاربر یک مورچه تخصیص داده می‌شود. با حرکت کاربر بر روی اسناد، مورچه متناظر با وی نیز بر روی گراف اسناد حرکت داده می‌شود. الگوریتم پیشنهادی بر اساس فرومون باقیمانده از حرکت کاربران بر روی گراف ارتباطات اسناد، میزان شباهت هر سند را با اسناد دیگر تخمین می‌زند. نتایج شبیه‌سازی انجام شده نشان می‌دهد که الگوریتم پیشنهادی در تشخیص شباهت اسناد و صفحات بهتر از الگوریتم هب و الگوریتم ارائه شده در [۱] عمل می‌کند. همچنین الگوریتم پیشنهادی با حذف جستجو برای یافتن دور در مسیر حرکت کاربر، علاوه بر کاهش بار محاسباتی این الگوریتم نسبت به الگوریتم ارائه شده در [۱]، امکان اجرای برخط آنرا نیز میسر می‌سازد.

ساختار ادامه این مقاله بصورت زیر است. در ادامه در بخش ۲ کلونی مورچه‌ها معرفی می‌شوند. در بخش ۳ الگوریتم پیشنهادی بیان می‌شود. در بخش ۴ پس از معرفی مدل استفاده شده برای شبیه‌سازی، نتایج شبیه‌سازی ارائه و بحث می‌شود. در نهایت، در بخش ۵ نتیجه‌گیری مقاله بیان می‌گردد.

۲ کلونی مورچه‌ها

دونوبورگ و همکاران با بررسی رفتار کلونی مورچه‌ها برای جستجوی غذا دریافتند که مورچه‌ها با بر جای گذاشتن یک ماده شیمیایی (اصطلاحا فرومون) مسیری بین محل غذا و لانه ایجاد می‌کنند [5]. هر مورچه در مسیر حرکت خود از لانه به محل غذا مقداری فرومون بر روی زمین می‌ریزد که موجب جذب سایر مورچه‌ها می‌شود. به این ترتیب که هر چه میزان فرومون یک مسیر بیشتر باشد، مورچه‌های بیشتری به آن جذب می‌شوند. بنابراین در بین دو مسیر کوتاه‌تر و بلندتر به سمت غذا، اگر تعداد مورچه‌هایی که از هر دو مسیر عبور می‌کنند برابر باشد، پس از مدتی مقدار فرومون ریخته شد در مسیر کوتاهتر بیشتر خواهد بود و مورچه‌ها بیشتر به سمت مسیر کوتاهتر جذب می‌شوند. علت اینست که در مدت زمان یکسان، تعداد دفعاتی که یک مورچه مسیر کوتاهتر را طی می‌کند بیشتر است و در نتیجه فرومون بیشتری نیز در این مسیر از خود باقی می‌گذارد. بهمین دلیل پس از مدتی، مورچه‌هایی که در مسیر طولانی‌تر در حال رفت و آمد هستند، به سمت مسیر کوتاهتر جذب می‌شوند. در این حالت به مرور زمان از مقدار فرومون ریخته شده در مسیر طولانی‌تر کاسته و به مقدار فرومون ریخته شده در مسیر کوتاهتر نیز افزوده می‌شود. در نهایت پس از مدتی مورچه‌ها فقط در مسیر کوتاهتر حرکت خواهند کرد.

یک مشکل روش کلونی مورچه‌ها در پیدا کردن کوتاهترین مسیر، اضافه شدن یک مسیر کوتاهتر به سیستم، پس از همگرا شدن کلونی به کوتاهترین مسیر قبلی است. از آنجاییکه در مسیر انتخابی فعلی مقدار فرومون ریخته شده بسیار بیشتر از سایر مسیرها، از جمله مسیر کوتاهتر اضافه شده، می‌باشد، مورچه‌ها مسیر کوتاهتر جدید را انتخاب نمی‌کنند. این مساله در محیط‌های پویا (مانند وب) دارای اهمیت بیشتری می‌باشد. یک راه‌حل، کاهش اثر فرومون ریخته شده با گذشت زمان است (فرایند تبخیر). بدین صورت که اگر فرومون ریخته شده در مسیر تبخیر شود، بهترین مسیر انتخاب شده قدیمی پس از مدت زمان محدودی جای خود را به مسیر کوتاه‌تر جدید می‌دهد. علت اینست که بعضی از مورچه‌ها بطور تصادفی مسیر کوتاهتر جدید را انتخاب می‌کنند. و بتدریج مقدار فرومون باقیمانده در مسیر جدید بیشتر از مسیر انتخاب شده قبلی می‌شود.

چگونگی انتخاب کوتاهترین مسیر توسط یک کلونی مورچه بعنوان ایده الگوریتم‌های کلونی مورچه‌ها برای استفاده در مسائلی که فضای حالت بزرگی دارند استفاده می‌شوند [6]. برای توصیف بهتر این الگوریتم، کاربرد این الگوریتم در مساله شناخته شده فروشنده دوره‌گرد ارائه می‌شود.

مجموعه‌ای از n شهر را در نظر بگیرید که مساله فروشنده دوره‌گرد، پیداکردن کوتاهترین مسیر برای تنها یک بار ملاقات همه n شهر است. مساله بصورت یک گراف (N,E) مدل می‌شود که در آن N مجموعه شهرها (گره‌های گراف) و E مجموعه مسیرهای بین شهرها (یالهای گراف) است. فاصله بین دو شهر i و j (هزینه یا وزن یال بین دو گره i و j) بصورت d_{ij} نمایش داده می‌شود. فرض کنید $b_i(t)$ ، $i=1...n$

تعداد مورچه‌ها در شهر i در زمان t و $m = \sum_{i=1}^{n} b_i(t)$ مجموع تعداد مورچه‌ها باشد.

هر مورچه، شهر بعدی برای ملاقات را با احتمالی که تابعی از فاصله آن شهر و مقدار فرومون موجود بر یال مورد نظر است، انتخاب می‌کند. برای جلوگیری از ایجاد دور در مسیر انتخاب شده توسط یک مورچه، انتخاب یالهای منتهی به یک شهر ملاقات شده، ممنوع است. هنگامیکه یک مورچه همه شهرها را ملاقات می‌کند، فرومون خود را بر تمام یالهایی که از آنها عبور کرده است، قرار می‌دهد.

مقدار فرومون موجود بر روی یال (i,j) در زمان t با $\tau_{ij}(t)$ نشان داده می‌شود. هر مورچه در زمان t شهر بعدی را برای ملاقات انتخاب می‌کند و آنرا در زمان t+1 ملاقات می‌کند. بنابراین اگر در هر گام (در بازه زمانی t تا t+1) m حرکت توسط m مورچه موجود انجام شود، در هر n گام الگوریتم (که یک دور گفته می‌شود) هر مورچه یک تور را کامل کرده است. در این زمان مقدار فرومون ریخته شده بر روی یالها بر اساس رابطه بروز می‌شود.

$$\tau_{ij}(t+n) = \rho.\tau_{ij}(t) + \Delta\tau_{ij} \qquad (۱)$$

که $1-\rho$ نسبت تبخیر فرومون در فاصله t تا t+n را مشخص می‌کند. برای جلوگیری از تجمع نامحدود فرومون بر روی یک یال، برای ضریب ρ محدودیت $0 < \rho < 1$ در نظر گرفته می‌شود.

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \qquad (۲)$$

مقدار فرومونی است که مورچه k در واحد طول بر روی مسیر (i,j) و در فاصله زمانی t تا t+n از خود باقی می‌گذارد($\Delta\tau_{ij}^{k}$) بصورت رابطه زیر محاسبه می‌شود.

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{L_k} & \text{if } k^{th} \text{ ant uses edge (i, j) at time (t, t + n)} \\ 0 & \text{otherwise} \end{cases} \qquad (۳)$$

که Q یک مقدار ثابت و L_k طول مسیر طی شده توسط مورچه k می‌باشد.

احتمال حرکت از شهر i به شهر j برای مورچه k در زمان t ($p_{ij}^{k}(t)$) بر اساس رابطه (۴) بیان می‌شود . در این رابطه η_{ij} میدان دید و برابر با $\frac{1}{d_{ij}}$ می‌باشد (شهرهای نزدیک‌تر احتمال انتخاب بیشتری دارند). permitted_k مجموعه شهرهایی است که مورچه k تا کنون آنها را ملاقات نکرده است و می‌تواند در صورت امکان در گام بعدی آنها را ملاقات کند. α و β نیز بترتیب پارامترهای تاثیر مقدار فرومون ریخته شده بر روی یک یال و میدان دید آن یال (عکس اندازه یال) است. با قرار دادن مقدار $\alpha = 0$ تاثیر فرومون‌ها از بین رفته و تنها انتخاب شهر بعدی بر اساس نزدیکی آن انجام می‌شود که در این حالت الگوریتم به یک الگوریتم تصادفی حریصانه با چندین نقطه شروع تبدیل می‌شود.

$$p_{ij}^{k}(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}}{\sum_{j \in permitted_k} [\tau_{ij}(t)]^{\alpha}.[\eta_{ij}]^{\beta}} & \text{if } j \in \text{Allowed}_k \\ 0 & \text{otherwise} \end{cases} \qquad (۴)$$

۳ الگوریتم پیشنهادی

در این بخش دو الگوریتم پیشنهادی کاوش اطلاعات استفاده مبتنی بر کلونی مورچه‌ها را معرفی می‌کنیم. این الگوریتم‌ها برای تشخیص شباهت صفحات با توجه به رفتار کاربران استفاده می‌شود و می‌توان از آنها بعنوان گام اولیه فرایند دسته‌بندی اسناد (قبل از اعمال نظر یک فرد خبره) یا در یک سیستم پیشنهاد دهنده استفاده نمود. در حالتی که کلمات کلیدی اسناد مشخص نبوده یا محتوای آنها قابل استخراج نباشند (مانند اسناد چندرسانه‌ای) استفاده از این الگوریتم و سایر روشهایی که تنها بر اساس اطلاعات استفاده از اسناد عمل می‌کنند، بیشتر مورد توجه می‌باشد.

۳,۱ الگوریتم کلونی‌مورچه-۱

در این الگوریتم گراف صفحات وب و ارتباطات آنها بعنوان مسیرهای قابل عبور مورچه‌ها در نظر گرفته می‌شود. ایده این الگوریتم به این صورت است که متناظر با هر کاربر یک مورچه در نظر گرفته می‌شود که همراه با حرکت کاربر در فضای اسناد، بر روی گراف اسناد حرکت می‌کند. در این نسخه از الگوریتم پیشنهادی، با حرکت کاربر از سند i به سند j، مورچه متناظر با آن کاربر نیز از یال (i,j) در گراف اسناد عبور می‌کند و با عبور خود از این مسیر از خود بر روی آن فرومون باقی می‌گذارد. نحوه فعالیت این الگوریتم بصورت زیر می‌باشد.

در آغاز فعالیت الگوریتم، مقدار فرومون موجود در هر یک از مسیرهای گراف صفحات یکسان و برابر با صفر می‌باشد. با ورود یک کاربر به سایت، یک مورچه به این کاربر تخصیص داده می‌شود. بصورتی که مسیر حرکت این مورچه متناظر با

مسیر حرکت کاربر در بین اسناد می‌باشد. با حرکت کاربر در بین اسناد، مورچه متناظر با آن کاربر نیز بر روی گراف اسناد حرکت کرده و بر روی هر یال فرومونی (به میزان Q) باقی می‌گذارد. همچنین با عبور هر مورچه از روی هر یال، میزان تبخیر فرومون آن یال نیز محاسبه می‌شود (بر اساس ضریب تبخیر $\rho-1$). شبه‌کد الگوریتم پیشنهادی در شکل ۱. شبه‌کد الگوریتم پیشنهادی بر اساس کلونی مورچه‌ها نشان داده شده است.

```
Procedure Ant1_usage_minig
variables:
  τ_{cur_doc,next_doc} = 0 ; /* for all documents
  ρ ; // 1- ρ is the evaporation coefficient
  user_log: Array of [Number of Users][Users Path]
    /* user log, documents viewed by each user.
     each records contains trace of a user. */
begin
  for all users do
    doc_id = 1;
    while user is browsing the site
      cur_doc = user_log[user_id][ doc_id ];
      doc_id = doc_id + 1;
    /* find next document( α )visited by current user */
      α = user_log[user_id][ doc_id ];
      τ_{cur_doc,next_doc} = ρ.τ_{cur_doc,next_doc} + Q ;
    end
  end
end
```

شکل ۱. شبه‌کد الگوریتم پیشنهادی بر اساس کلونی مورچه‌ها

در این الگوریتم میزان شباهت دو سند متناظر با میزان فرومون ریخته شده در یال متصل کننده آن دو سند به یکدیگر می‌باشد.

$$d'_{ij} = \tau(i,j) \qquad (۵)$$

در شروع الگوریتم مقدار اولیه فرومون هر یال برابر با یک مقدار مثبت کوچک یا برابر با صفر در نظر گرفته می‌شود.

۳,۲ الگوریتم کلونی‌مورچه-۲

در این الگوریتم با توجه به اینکه شباهت دو سند با یکدیگر، ارتباطی دو طرفه است، در الگوریتم کلونی مورچه-۲ مقدار فرومون ریخته شده در مسیری از گره i به گره j و مسیر عکس آن یعنی از گره j به گره i با هم ذخیره می‌شود. بعبارت دیگر $\tau(i,j)$ با $\tau(j,i)$ یکی در نظر گرفته می‌شود. برای این کار از ماتریس τ بصورت یک ماتریس بالا مثلثی برای نگهداری فرومون مسیرها استفاده می‌شود بطوریکه مقدار فرومون موجود بر روی مسیری از گره i به گره j ($\tau(i,j)$) بر اساس رابطه (۶) بدست می‌آید:

$$\tau(i,j) = \begin{cases} \tau'(i,j) & if \quad i \le j \\ \tau'(j,i) & if \quad i > j \end{cases} \qquad (۶)$$

پس از اجرای این الگوریتم شباهت دو سند i و j بصورت رابطه (۷) محاسبه می‌شود.

$$ant\_relation(i,j) = \tau(i,j) \quad for \quad i=1,2,\cdots,n \quad j=1,2,\cdots,n \qquad (۷)$$

۴ شبیه‌سازی

در این بخش نتایج بدست آمده از شبیه‌سازی الگوریتم پیشنهادی را با یک روش آماری ساده، روش هب[8] و دو روش مبتنی بر اتوماتای یادگیر [۱] و [۲] مقایسه می‌کنیم.

۴,۱ مدل شبیه‌سازی

در این مقاله از مدل معرفی شده در [10] برای مدل کردن رفتار کاربران در محیط وب استفاده شده است. در [10] نظم موجود در رفتارهای کاربران در محیط وب با استفاده از یک مدل مبتنی بر عامل مشخص و اعتبار مدل خود را با استفاده از اطلاعات استفاده از وب چندین سایت وب بزرگ مانند مایکروسافت، تایید شده‌است. بر این اساس، در این مقاله مطابق با مدل رفتار کاربران، پروفایل علاقه کاربران بصورت توزیع قانون-توانی[۷] و توزیع محتوای اسناد بصورت توزیع نرمال در نظر گرفته شده است. سایر پارامترهای استفاده شده در مدل [10] برای شبیه‌سازیهای انجام شده در این قسمت در جدول ۱ نشان داده شده است.

حد آستانه ایجاد اتصال	۰/۷
تعداد کاربران	۱۰۰۰۰
تعداد اسناد	۲۶
تعداد موضوع‌ها	۴
T_c مقدار ثابت سند اولیه (صفحه اولیه سایت) در موضوعات مختلف	۰/۲
ΔM_t^c ضریب ثابت کاهش اشتیاق کاربر	-
ΔM_t^v ضریب متغیر کاهش اشتیاق کاربر	-
α_u پارامتر توزیع قانون-توانی توزیع احتمال علایق کاربران	۱
ϕ ضریب پاداش دریافتی از مشاهده یک سند	۱/۲
λ ضریب جذب اطلاعات از یک سند توسط یک کاربر	۰/۵
μ_m میانگین توزیع نرمال ΔM_t^v	۵/۹۷
σ_m واریانس توزیع نرمال ΔM_t^v	۰/۲۵

μ_t میانگین توزیع نرمال برای مقدار افزایش یک گره برای یک موضوع خاص	-
α_p پارامتر توزیع قانون-توانی توزیع احتمال وزنهای مطالب برای هر سند	۳
σ_t واریانس توزیع نرمال برای مقدار افزایش یک گره برای یک موضوع خاص	۰/۲۵
θ ضریب کاهش علاقه کاربر	۱
حداقل اشتیاق کاربر برای ادامه جستجو	۰/۲

جدول ۱: پارامترهای استفاده شده در مدل شبیه‌سازی

۴,۲ شاخص ارزیابی

میزان شباهت بدست آمده برای اسناد با استفاده از الگوریتم پیشنهادی، با مقدار شباهت قرارداده شده در مدل شبیه‌سازی بصورت زیر مقایسه می‌شود.

در مدل استفاده شده M موضوع برای اسناد تولید شده تعریف شده است. میزان ارتباط (شباهت) سند i با هر یک از این موضوعات مانند موضوع j با cw_i^j مشخص می‌شود و یک درایه بردار محتوای سند i را می‌سازد(رابطه (۸)). بر این اساس شباهت اسناد در یک مجموعه را توسط یک ماتریس بنام ماتریس شباهت(D) نشان می‌دهیم. بصورتی که هر درایه d_{ij} این ماتریس فاصله اقلیدسی بردارهای محتوای دو سند i و j است (رابطه (۸)) و بصورت رابطه (۹) محاسبه می‌شود.

در الگوریتم پیشنهادی شباهت دو سند برابر با میزان فرومون ریخته شده در یال متصل کننده آن دو سند به یکدیگر می‌باشد. به این ترتیب ماتریس شباهت حاصل از الگوریتم پیشنهادی (D') بر اساس رابطه (۱۰) محاسبه می‌گردد. هر چه دو ماتریس D و D' به یکدیگر شبیه‌تر باشند، الگوریتم بهتر عمل کرده است.

$$C_i = \begin{bmatrix} cw_i^1 & cw_i^2 & \cdots & cw_i^M \end{bmatrix} \quad (۸)$$

$$d_{ij} = \sqrt{\sum_{k=1}^{M} \left(cw_i^k - cw_j^k\right)^2} \quad (۹)$$

$$d'_{ij} = \tau(i,j) \quad (۱۰)$$

برای نشان دادن کارایی الگوریتم پیشنهادی از کوریلیشن دو ماتریس D و D' استفاده می‌کنیم(رابطه (۱۱)). از آنجاییکه میزان شباهت دو سند با فاصله بردار محتوای آنها (d_{ij}) ارتباط عکس دارد، در صورتی که الگوریتم خوب عمل کند مقدار کوریلیشن منفی خواهد شد. هر چه این مقدار کوچکتر باشد، الگوریتم بهتر عمل کرده است. در ادامه مقاله دقت روشهای مورد بررسی معادل با کوریلیشن دو ماتریس D (ماتریس شباهت واقعی) و D' (ماتریس شباهت حاصل از الگوریتم) در نظر گرفته می‌شود.

$$Correlation(D, D')$$
$$D = \{d_{ij} \mid i, j = 1,2,\cdots,n, \quad i \neq j\} \quad (۱۱)$$
$$D' = \{d'_{ij} \mid i, j = 1,2,\cdots,n, \quad i \neq j\}$$

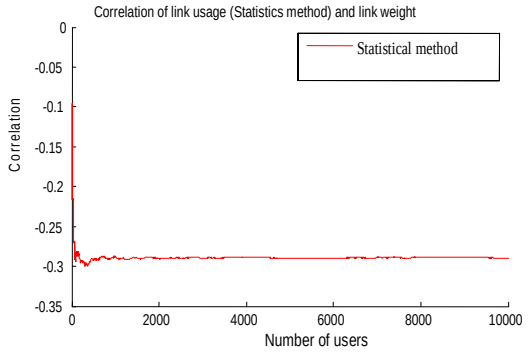
۴,۳ نتایج شبیه‌سازی

در این مقاله نتایج الگوریتم پیشنهادی را با روش هب ساده و تعمیم‌یافته[8] مقایسه می‌کنیم. همچنین با هدف نشان دادن نتیجه ساده‌ترین روش، نتایج یک روش آماری ساده نیز در ادامه نشان داده شده است. این روش آماری ساده بصورت زیر کار می‌کند.

در این روش شباهت دو سند i و j، نسبت تعداد دفعاتی که کاربران از سند i به سند j حرکت کرده‌اند ((s(i,j)) به تعداد دفعاتی که کاربران از سند i به هر سند دیگری مانند k حرکت نموده‌اند (رابطه (۱۲)) می‌باشد.

$$similarity(i, j) = \frac{s(i, j)}{\sum_{k=1}^{n} s(i,k)} \quad (۱۲)$$

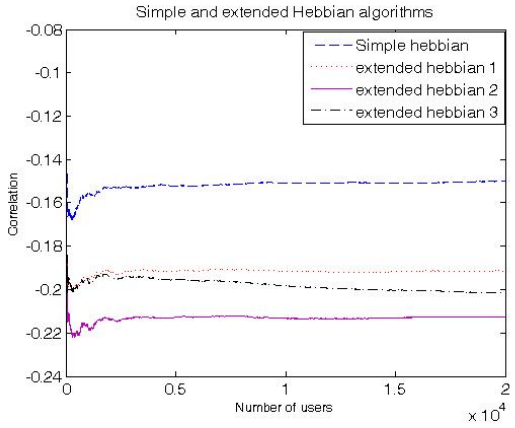
دقت روش آماری ساده در شکل ۲ مشخص شده است. همانطور که در این شکل مشاهده می‌شود، دقت پایین این الگوریتم (نزدیک به ۰/۳) بیانگر اینست که نمی‌توان از این الگوریتم به تنهایی برای تشخیص شباهت بین اسناد استفاده کرد. اما از این مقدار می‌توان برای نشان دادن حداقل انتظار از سایر الگوریتمها استفاده نمود.



شکل ۲. کوریلیشن روش آماری ساده

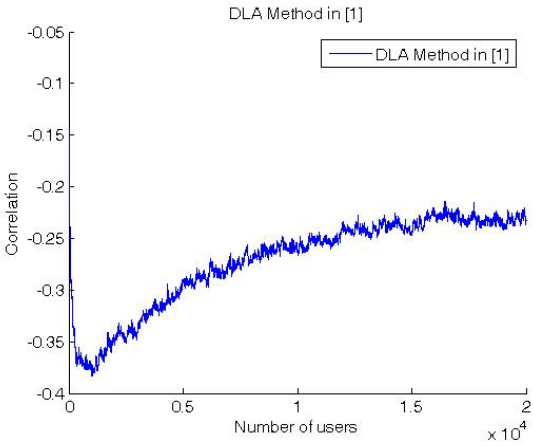
دقت الگوریتم هب ساده و تعمیم‌یافته برای تعیین شباهت اسناد در شکل ۴ نشان داده شده است. در روش هب ساده و

توسعه یافته مقدار a_{ij} با حرکت یک کاربر بر روی لینک (i,j) افزایش می‌یابد. بنابراین انتظار می‌رود زمانیکه فاصله اقلیدسی بین دو گره i و j کم می‌باشد (کوچک بودن مقدار d_{ij}) مقدار a_{ij} بزرگ باشد. این فرض با در نظر گرفتن اینکه عاملها بصورت عاقلانه اتصالات با وزن (فاصله) کمتر را برای حرکت بعدی خود انتخاب می‌کنند، متناسب است. بنابراین انتظار می‌رود که کوریلیشن مقادیر a_{ij} با مقادیر واقعی وزنهای (فواصل) گره‌ها d_{ij} منفی باشد. همانطور که در شکل ۴ نیز مشاهده می‌شود، مقادیر کوریلیشن ماتریس A و D مقداری منفی می‌باشد.



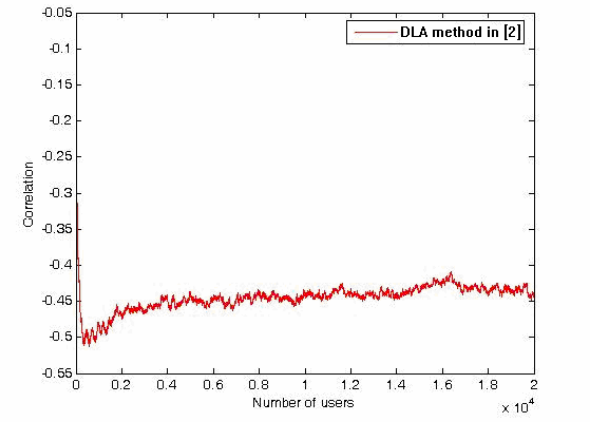
شکل ۳. کوریلیشن ماتریس شباهت، الگوریتم هب ساده و تعمیم‌یافته

دقت الگوریتم معرفی شده در [۱] در شکل ۴ نشان داده شده است. همانطور که مشاهده می‌شود، دقت این الگوریتم بیشتر از بهترین نسخه آزمایش شده الگوریتم هب می‌باشد. اما بر خلاف الگوریتم‌های هب، برای رسیدن به یک وضعیت ثابت، نیاز به مشاهده رفتار تعداد زیادی از کاربران دارد.



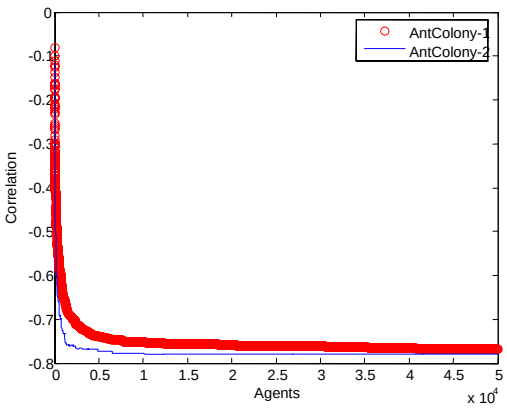
شکل ۴. کوریلیشن ماتریس شباهت، الگوریتم ارائه شده در [۱]

دقت روش مبتنی بر اتوماتای یادگیر توزیع‌شده با تعداد اقدامهای متغیر[۲] برای تشخیص شباهت اسناد بهتر از سایر روشها مطرح شده می‌باشد. همانطور که در شکل ۵ مشاهده می‌شود، مقدار کوریلیشن ماتریس شباهت حاصل از این روش با ماتریس شباهت اسناد نزدیک به ۰,۴۵ می‌باشد که در مقایسه با سایر روشهای فوق، بیش از دو برابر افزایش دقت نشان داده است.



شکل ۵. کوریلیشن ماتریس شباهت، الگوریتم ارائه شده در [۲]

در شکل ۶ دقت الگوریتم پیشنهادی با استفاده از کلونی مورچه‌ها نشان داده شده است. در هر دو الگوریتم کلونی مورچه۱ و ۲ پس از بررسی رفتار ۲۰۰۰ کاربر، کوریلیشن ماتریس شباهت بدست آمده و ماتریس شباهت اسناد، به ۷۰٪- نزدیک می‌شود. همانطور که انتظار می‌رود، الگوریتم کلونی مورچه-۲ از الگوریتم کلونی مورچه-۱ کمی بهتر عمل می‌کند. می‌توان علت این کارایی بهتر را مشاهده نمونه‌های بیشتر در مدت زمان یکسان دانست. چرا که در الگوریتم کلونی مورچه-۲ بازای عبور یک مورچه از مسیر (i,j) و مسیر (j,i) یک مقدار بر اساس رابطه (۴) بروز می‌شود در حالی که در الگوریتم کلونی مورچه-۱ دو مقدار $\tau(i,j)$ و $\tau(j,i)$ بروز می‌شوند که موجب سریعتر شدن همگرایی الگوریتم کلونی مورچه-۲ می‌شود.



شکل ۶.کوریلیشن ماتریس شباهت، الگوریتم‌های پیشنهادی کلونی‌مورچه

۵ نتیجه‌گیری

در این مقاله الگوریتم جدیدی با استفاده از کلونی مورچه‌ها برای کاوش داده‌های استفاده از وب ارائه شده است. این الگوریتم تنها با استفاده از داده‌های استفاده از یک سایت وب می‌تواند شباهت بین صفحات را بر اساس نحوه استفاده کاربران از سایت تشخیص دهد. نتایج بدست آمده نشان می‌دهد که این الگوریتم نسبت به روش هب، روش خودسازمانده [۱] و روش مبتنی بر اتوماتای یادگیر توزیع‌شده[۲] از دقت بیشتری برای تشخیص شباهت اسناد برخوردار است.

مراجع

[۱] سعید ساعتی و محمدرضا میبدی، "یک مدل خودسازمانده برای ساختار اطلاعاتی اسناد با استفاده از اتوماتای یادگیر توزیع‌شده،" مجموعه مقالات دومین کنفرانس بین‌المللی فناوری اطلاعات و دانش، تهران، ایران، ۱۳۸۴.

[۲] علی برادران هاشمی و محمدرضا میبدی، داده‌کاوی استفاده از وب با استفاده از اتوماتای یادگیر توزیع‌شده، دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، تهران، ایران، ۱۳۸۵.

[3] R. Armstrong, D. Freitag, T. Joachims, and T. Mitchell, "WebWatcher: A learning apprentice for the world wide web," Proceedings of AAAI Spring Symposium on Information Gathering, AAAI Press, 1995, pp 6–12.

[4] M. Balabanovic and Y. Shoham, "Learning information retrieval agents: Experiments with automated web browsing," Proceedings of AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, AAAI Press, 1995, pp. 13-18.

[5] R. Beckers, S. Goss, Jean-Louis Deneubourg, and J. M. Pasteels, "Colony size, communication and ant foraging strategy," PSYCHE (CAMBRIDGE), vol. 96, no. 3, 1989, pp. 239-256.

[6] M. Dorigo, V. Maniezzo & A. Colorni , "The Ant System: Optimization by a colony of cooperating agents," IEEE Transactions on Systems, Man, and Cybernetics-Part B, vol. 26, no. 1, 1996, pp. 29-41.

[7] Pierrakos, G. Paliouras, C. Papatheodorou, and C.D. Spyropoulos, "Web Usage Mining as a Tool for Personalization: A Survey," User Modeling and User-Adapted Interaction, vol. 13, no. 4, 2003, pp. 311-372.

[8] F. Heylighen and 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] T. Joachims, "Optimizing search engines using click through data," In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-02), 2002, pp. 133–142.

[10] J. Liu, S. Zhang, and J. Yang, "Characterizing Web Usage Regularities with Information Foraging Agents," IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 4, April 2004, pp. 566-584.

[11] José Manuel Barrueco Cruz, Thomas Krichel, "Automated Extraction of Citation Data in a Distributed Digital Library," Proceedings of the 2nd International Workshop on New Developments in Digital Libraries, 2002, pp 51-62.

[12] D. Mladenis, Personal WebWatcher: Implementation and design. Technical Report IJS-DP-7472, Department of Intelligent Systems, Joz, es Stefan Institute, 1996.

[13] B. Mobasher, R. Cooley, and J. Srivastava, "Automatic personalization based on web usage mining," Communications of the ACM, vol. 43, no. 8, 2000, pp. 142–151.

[14] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa, "Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization," Data Mining and Knowledge Discovery, vol. 6, no. 1, 2002, pp. 61–82.

[15] Junichiro Mori, Yutaka Matsuo, Mitsuru Ishizuka, and Boi Faltings, "Keyword Extraction from the Web for FOAF Metadata," Proceeding of 1st Int'l Workshop on Friend of a Friend, Social Networking and the Semantic Web, Galway, Ireland, 2004.

[16] M. Pazzani, J. Muramatsu, and D. Billsus, "Syskill & Webert: Identifying interesting web sites," In Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-96), AAAI Press, 1996, pp. 54–61.

[17] Mike Perkowitz and Oren Etzioni, "Adaptive Web Sites," Communications of ACM, vol. 43, no. 8, 2000, pp. 152-158.

[18] J. Srivastava, R. Cooley, M. Deshpande, and P.N. Tan, "Web usage mining: Discovery and applications of usage patterns from web data," SIGKDD explorations, vol. 1, no. 2, 2000, pp. 12–23.

[19] D. O. Hebb, The organization of behavior: A neuropsychological theory, Wiley-Interscience, New York, 1949.

[20] W. M. Teles, L. Weigang, and C. G. Ralha, "AntWeb: The Adaptive Web Server Based on the Ants' Behavior," Proceedings of IEEE/WIC International Conference on Web Intelligence, 2003, pp. 558-564.

---

۱ Synonym

۲ Homonym

۳ Content mining

۴ Structure mining

۵ Log files

۶ Web usage mining

۷ Power-law