پیمایش موضوعی وب با استفاده از اتوماتای یادگیر توزیع‌شده و پارتیشن‌بندی گراف

مجید تاران[1]، شهرزاد معتمدی مهر[2]، علی برادران هاشمی[3] و محمدرضا میبدی[4]
[1]شرکت خدمات انفورماتیک، تهران، ایران، m_taran@isc.iranet.net
[2]دانشکده فنی و مهندسی، دانشگاه خوارزمی، تهران، ایران، motamedi@tmu.ac.ir
[3]دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران، a_hashemi@aut.ac.ir
[4]دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی امیرکبیر، تهران، ایران، mmeybodi@aut.ac.ir

چکیده - پیمایش وب جهت یافتن صفحاتی که توسط موتور جستجو شاخص گذاری شوند، از اهمیت بالایی برخوردار است. از آن جا که حجم صفحات وب بسیار بالا و همواره رو به افزایش است، موتورهای جستجو قادر به شاخص گذاری صفحات محدودی هستند. پیمایشگرهای موضوعی یا پیمایشگرهای متمرکز، در روند پیمایش خود به صورت انتخابگر عمل می‌کنند و صفحاتی را برای پیمایش انتخاب می‌کنند که تا حد ممکن در ارتباط با موضوعی خاص باشند. در این مقاله روشی ترکیبی مبتنی براتوماتای یادگیر توزیع شده و پارتیشن‌بندی گراف برای پیمایش موضوعی صفحات وب پیشنهاد می‌گردد. روش پیشنهادی با استفاده از الگوریتم *HITS* و ساختار پیوند بین صفحات که از طریق رفتار کاربر در مشاهده این صفحات بدست آمده است، صفحاتی را برای پیمایش انتخاب می‌کند. در این مقاله از پارتیشن‌بندی گراف وب برای بهبود کارایی استفاده شده است. به منظور ارزیابی، نتایج روش پیشنهادی با نتایج الگوریتم عرض اول، *Best First* و الگوریتمی دیگری مبتنی بر اتوماتای یادگیر توزیع شده مقایسه شده است. نتایج آزمایشها حاکی از کارایی روش پیشنهادی دارد.

کلید واژه- پیمایش موضوعی وب، اتوماتای یادگیر توزیع شده، پارتیشن بندی گراف، داده‌های استفاده از وب، الگوریتم HITS

مقدمه

با گسترش روز افزون شبکه جهانی اینترنت، بازیابی اسناد مرتبط با نیاز اطلاعاتی کاربران که در قالب یک پرس و جو مطرح می‌شود، روز به روز دشوارتر می‌گردد. موتورهای جستجو برای بازیابی اسناد مرتبط با یک پرس و جو از یک الگوریتم رتبه بندی استفاده می کنند که میزان ارتباط اسناد شاخص‌گذاری شده توسط موتور جستجو را با پرس‌وجوی کاربر محاسبه می-کند. اما الگوریتم رتبه‌بندی هر اندازه که دقیق باشد، اگر صفحه-ای توسط موتور جستجو شاخص‌گذاری نشده باشد، بازیابی نخواهد شد. از آن جا که حجم صفحات وب بسیار بالا و همواره رو به افزایش است، موتورهای جستجو قادر به شاخص‌گذاری صفحات محدودی هستند. به همین جهت از الگوریتم‌های پیمایش موضوعی جهت پیمایش صفحات مرتبط استفاده می-شود. پیمایشگرهای موضوعی یا پیمایشگرهای متمرکز، در روند پیمایش خود به صورت انتخابگر عمل می‌کنند و صفحاتی را برای پیمایش انتخاب می‌کنند که تا حد ممکن در ارتباط با موضوعی خاص باشند.

پیمایشگرهای موضوعی، باید بتوانند مستقل از مجموعه اولیه صفحات (که کار خود را با آن آغاز می‌کنند) به مجموعه‌ای از صفحات مرتبط با موضوعی خاص دست یابند. پیمایشگرها با استفاده از پیوندهای موجود در صفحاتی که قبلا بازیابی کرده‌اند، صفحات جدید را بازیابی می‌کنند. وقتی که صفحه‌ای بازیابی می شود، پیوندهای موجود در آن، به لیست صفحات بازیابی نشده اضافه می‌شود. این لیست، صف پیمایش(Crawl Frontier) نامیده می شود. چالش اصلی در پیمایش موضوعی آن است که مناسب ترین پیوند برای پیمایش از صف فوق الذکر انتخاب شود. یک پیمایشگر موضوعی باید از اطلاعات به دست آمده از پیمایش های قبلی برای تخمین میزان ارتباط یک پیوند جدید استفاده کند.

الگوریتمی که در انتخاب پیوند بعدی برای پیمایش به کار می‌رود، به هدف پیمایشگر بستگی دارد.

الگوریتم‌های پیمایش موضوعی را می‌توان به دو دسته کلی پیمایش بدون استفاده از دانش پیش‌زمینه (Background Knowledge) و پیمایش به کمک دانش پیش‌زمینه تقسیم کرد. در گروه اول، انتخاب پیوند برای پیمایش تنها بر اساس اطلاعات

موجود در صفحه وب انجام می‌شود. اما در گروه دوم برای تصمیم‌گیری از یک منبع خارجی مانند یک طبقه‌بندی از اسناد، هستان شناسی (Ontology) ، ... استفاده می‌گردد. البته الگوریتم‌های دیگری نیز پیشنهاد شده‌اند که در هیچ یک از این دو گروه قرار نمی گیرند.

از جمله روشهایی که در دسته اول قرار دارند می توان به الگوریتم عرض اول[3]، Best First [4]، Shark Search [5] و PageRank [6] اشاره کرد.

پیمایش عرض اول (Breath First) ، ساده ترین روش برای پیمایش می باشد. این الگوریتم در سال 1994 توسط Pinkerton ارائه شد. در این نوع پیمایش، صف پیمایش، یک صف FIFO است که با استفاده از آن پیوندها به همان ترتیبی که در صف قرار گرفته اند، پیمایش می شوند. زمانی که صف پر می شود، پیمایشگر می تواند تنها یکی از پیوندهای صفحه بازیابی شده را به صف اضافه کند. پیمایشگر عرض اول، به عنوان پیمایشگر پایه به کار می رود و از هیچ دانشی راجع به موضوع پیمایش استفاده نمی کند. الگوریتم BestFirst توسط cho و همکارانش در سال 1998 ارائه شد. ایده اصلی این الگوریتم آن است که از صف پیمایش، پیوندی که بهتر از سایرین معیاری را برآورده می کند، برای پیمایش انتخاب شود. در یک حالت ساده، معیار انتخاب، میزان شباهت صفحه حاوی پیوند (صفحه مبدا) با کلمات کلیدی موضوع مورد نظر می باشد. بنابراین از میزان شباهت صفحه مبدا با کلمات کلیدی موضوع برای تخمین مرتبط بودن صفحاتی که این پیوندها به آنها اشاره می کنند، استفاده می شود. الگوریتم SharkSearch در سال 1998 توسط Hersovici ارائه شد که نوعی از الگوریتم FishSearch [7] محسوب می شود. در FishSearch، پیمایشگر، در نقاطی از وب جستجوی خود را ادامه می دهد که صفحات مرتبط مشاهده شده اند و جستجو را در نقاطی که صفحات مرتبط یافته نشده اند، متوقف می کند. الگوریتم SharkSearch دو مزیت اصلی بر الگوریتم FishSearch دارد. تابعی که برای اندازه گیری میزان ارتباط صفحات به کار می رود، در الگوریتم FishSearch دودویی و در الگوریتم SharkSearch پیوسته و حقیقی است. همچنین امتیازی که به پیوندهای موجود در صف پیمایش اختصاص داده می شود، در الگوریتم SharkSearch پیچیده‌تر است. در الگوریتم FishSearch میزان ارتباط صفحه وب با موضوع مورد نظر بر اساس تطابق کلمات یا عبارات تعیین می شود. این امتیاز به متن پیوند، متن اطراف پیوند، امتیاز صفحات اشاره کننده به صفحه حاوی پیوند بستگی دارد. این الگوریتم را می‌توان گونه پیچیده‌تری از الگوریتم BestFirst دانست.

الگوریتم های ارایه شده [12][11][10][9][8] روش هایی هستند که از یک دانش پیش زمینه برای پیمایش موضوعی استفاده می کنند.

روش های پیمایش موضوعی وب که تاکنون گزارش شده است، از پیوندهای بین صفحات وب، متن صفحات وب و Tokenهای موجود در پیوندها استفاده می‌کنند. استفاده از این اطلاعات به تنهایی نمی‌تواند به خوبی میزان ارتباط یک صفحه با موضوعی خاص را نشان دهد. به علاوه برخی پدیده‌ها در وب مانند سایت های بی‌انتها (Bottomless Sites) ، تله‌های Spiderها و پیوندهای تبلیغاتی و مصنوعی، پیمایشگرها را با مشکل مواجه می کند.

در[1] و [2] روش های جدیدی مبتنی بر اتوماتای یادگیر توزیع شده جهت تعیین شباهت بین صفحات ارائه شده است.

در [1] 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 تراگذری تعیین می کند. در [2] 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 [1] 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 رابطه‌ای مشخص کاهش می یابد. جریمه دیگری که در روش پیشنهادی برای کاربر در نظرگرفته شده وجود دور در مسیر پیمایشی کاربر می باشد.

در این مقاله با استفاده ازالگوریتم ارائه شده در[2] شباهت صفحات وب تعیین می گردد.

روش پیشنهادی در گروه روش های پیمایش موضوعی وب قرار می گیرد که از یک دانش پیش زمینه (طبقه بندی از اسناد، هستان شناسی، ...) برای پیمایش استفاده می کنند. دانش پیش زمینه در این جا همان ساختار ارتباطی است که با استفاده از اتوماتای یادگیر توزیع شده و پارتیشن بندی گراف بدست آمده است. به علاوه از آن جا که در روش پیشنهادی از متن صفحات

وب استفاده نمی‌شود، این روش از نظر هزینه محاسباتی و زمان پردازش از روش‌های پیمایش مبتنی بر محتوای صفحات به صرفه‌تر می‌باشد.

ویژگی دیگر روش پیشنهادی آن است که پیمایشگر طراحی شده یک پیمایشگر یادگیر است که می تواند از دانش یاد گرفته شده در پیمایش‌های متفاوت استفاده کند.

همچنین یکی دیگر از مشکلات پیمایشگرهای موضوعی راکد شدن (Stagnation) می باشد به آن معنی که پیمایشگر دیگر قادر به جمع آوری صفحات مرتبط نمی باشد. این مشکل هیچ گاه برای پیمایشگرهای معمولی رخ نمی دهد. اما در پیمایشگرهای موضوعی در صورتی که موضوع مورد نظر خاص و محدود باشد، صفحات وب مرتبط به اندازه کافی با یکدیگر پیوند نداشته باشند و یا پیمایشگر در انتخاب پیوندها اشتباه کرده باشد، این مشکل ممکن است رخ دهد. در پیمایشگر طراحی شده در این بخش از آن جا که پیمایش تنها بر اساس محتوا و پیوندهای بین صفحات انجام نمی شود، این مشکل کاهش می یابد.

ادامه مقاله بدین صورت سازماندهی شده است. در بخش 2 اتوماتای یادگیر و اتوماتای یادگیر توزیع شده به اختصار معرفی می‌شوند. در بخش 3 الگوریتم پیشنهادی و در بخش 4 پس از معرفی مدل استفاده شده برای شبیه سازی، نتایج شبیه سازی ارائه می‌شود. بخش 5 نتیجه گیری می‌باشد.

1- اتوماتاهای یادگیر

اتوماتای یادگیر یک مدل انتزاعی است که بطور تصادفی یک اقدام از مجموعه متناهی اقدام‌های خود را انتخاب کرده و بر محیط اعمال می‌کند. محیط اقدام انتخاب شده توسط اتوماتای یادگیر را ارزیابی کرده و نتیجه ارزیابی خود را توسط یک سیگنال تقویتی به اتوماتای یادگیر اطلاع می‌دهد. سپس اتوماتای یادگیر با اطلاع از اقدام انتخاب شده و سیگنال تقویتی، وضعیت داخلی خود را بروز کرده و اقدام بعدی خود را انتخاب می‌کند.

محیط را می‌توان توسط سه‌تایی $E = \{a, b, c\}$ نشان داد که در آن $a = \{a_1, a_2, ..., a_r\}$ مجموعه ورودیها، $b = \{b_1, b_2, ..., b_r\}$ مجموعه خروجیها و $c = \{c_1, c_2, ..., c_r\}$ مجموعه احتمالات جریمه می‌باشد. 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 $\{a, b, p, T\}$ نشان داده می‌شود که در آن $a = \{a_1, a_2, \mathbf{L}, a_r\}$ مجموعه اقدام‌های اتوماتای یادگیر، $b = \{b_1, b_2, \mathbf{L}, b_r\}$ مجموعه ورودیهای اتوماتای یادگیر، $p = \{p_1, p_2, \mathbf{L}, p_r\}$ بردار احتمال انتخاب هر یک از اقدام‌ها و T، $p(n+1) = T[a(n), b(n), p(n)]$ الگوریتم یادگیری اتوماتای یادگیر می‌باشد. الگوریتم‌های یادگیری متنوعی برای اتوماتای یادگیر ارائه شده است با استفاده از الگوریتم یادگیری خطی، اتوماتای یادگیر بردار احتمال انتخاب اقدام‌های خود را مطابق رابطه (1) تنظیم می‌کند.

$$\begin{aligned} p_i(n+1) &= p_i(n) + a.(1-b(n)).(1-p_i(n)) - b.b(n).p_i(n) \\ p_j(n+1) &= p_j(n) + a(1-b(n)).p_j(n) + \frac{b.b(n)}{r-1} - b.b(n).p_j(n) \qquad if\ j \neq i \end{aligned} \tag{1}$$

که a پارامتر پاداش و b پارامتر جریمه می‌باشد. اگر a و b با هم برابر باشند، الگوریتم L_{R-P}، اگر b از a خیلی کوچکتر باشد، الگوریتم L_{ReP} و اگر b صفر باشد، الگوریتم L_{R-I} نام دارد[14].

1-1- اتوماتای یادگیر توزیع شده

اتوماتای یادگیر توزیع‌شده شبکه‌ای از چند اتوماتای یادگیر است که برای حل یک مساله مشخص با یکدیگر همکاری می‌کنند. یک اتوماتای یادگیر توزیع‌شده را می‌توان با یک گراف جهت‌دار مدل کرد. بصورتی 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. در هر لحظه فقط یک اتوماتای یادگیر در اتوماتای یادگیر توزیع‌شده فعال می‌باشد[13][15].

2- روش پیشنهادی

روش پیشنهادی برای پیمایش موضوعی از دو مرحله کلی تشکیل شده است. در مرحله اول الگوریتم ارائه شده، صفحات وب جدید را بر اساس پیمایش کاربر، استفاده کاربران قبلی و اطلاعات پیوندی گراف سایت به کاربر جاری پیشنهاد می دهد. بدین منظور اتوماتای یادگیر توزیع شده شباهت بین صفحات وب را با استفاده از اطلاعات پیمایش کاربران قبلی و پیوند گراف وب سایت تعیین می کند[2]. در مرحله دوم، با استفاده از ساختار ارتباطی به دست آمده از مرحله قبل، پیمایش موضوعی وب انجام می شود و صفحات مرتبط با یک موضوع خاص جمع آوری

می گردد. پس از آن که مرحله اول برای یک بار انجام شد، می توان مرحله دوم را برای دفعات متعدد انجام داد.

پس از اصلاح احتمال انتخاب اعمال اتوماتای یادگیر توزیع شده، امتیاز Hub و Authority هر یک از صفحات مجموعه پایه طبق فرمول (2) و (3) بروز می شود.

$$Authority\ (i) = \sum_{\forall j \to i} r(j,i) \times Hub\ (j) \qquad (2)$$

$$Hub\ (i) = \sum_{\forall i \to j} r(i,j) \times Authority\ (j) \qquad (3)$$

نحوه عملکرد بخش دوم به شرح زیر است:

ابتدا موضوع مورد نظر برای پیمایش به عنوان ورودی به الگوریتم داده می شود. سپس مجموعه ای به نام مجموعه آغازین که حاوی تعدادی صفحه وب می باشد، ایجاد می گردد و پیمایشگر کار خود را با این مجموعه آغاز می کند. این مجموعه را می توان به روش های مختلف ایجاد کرد. در پیمایشگر طراحی شده در این قسمت، از آنجا که در مدل شبیه سازی استفاده شده هر صفحه دارای برداری است که میزان ارتباط این صفحه با موضوعات موجود در مدل شبیه سازی را نشان می دهد، از این بردار برای ایجاد مجموعه آغازین استفاده می کنیم. برای این منظور صفحات را به پنج گروه تقسیم می کنیم به طوری که میزان ارتباط گروه اول با موضوع انتخاب شده بین 0,8 و 1، گروه دوم بین 0,8 و 0,6، ... و گروه پنجم بین 0,2 و 0 باشد. سپس از هر گروه 20% صفحات مجموعه آغازین را به صورت تصادفی انتخاب می کنیم.

سپس یک صف اولویت که صف پیمایش نامیده می شود، ایجاد شده و هر یک از صفحات مجموعه آغازین در این صف قرار می گیرد. این صف شامل صفحاتی است که تا مرحله جاری الگوریتم توسط پیمایشگر پردازش نشده اند. هر صفحه در این صف امتیازی دارد که اولویت آن برای پیمایش را تعیین می کند. صفحات بر اساس این امتیاز مرتب شده و در صف پیمایش قرار می گیرند. در ابتدا امتیاز کلیه صفحات مجموعه آغازین با یکدیگر برابر می باشد. همچنین مجموعه‌ای به نام مجموعه پیمایش ایجاد می گردد که شامل صفحاتی خواهد بود که پیمایش می شوند و در این مرحله تهی می باشد. سپس تا زمانی که تعداد صفحات پیمایش شده به حد آستانه ای برسد، عملیات زیر تکرار می شود:

صفحه ای که در ابتدای صف پیمایش قرار دارد، انتخاب و از صف حذف می گردد و به مجموعه صفحات پیمایش شده اضافه می شود. سپس صفحاتی که این صفحه در ساختار ارتباطی به آن ها اشاره می کند، استخراج می شوند و برای هر یک از آنها امتیازی که اولویتشان را برای پیمایش تعیین می کند، محاسبه می شود. این امتیاز به دو عامل بستگی دارد و برای صفحه ای مانند $Page_j$ طبق رابطه (4) محاسبه می شود.

$$crawl_score(Page_j) = r(i,j) \times hub(i) \qquad (4)$$

که در آن $r(i,j)$ میزان ارتباط صفحه i (صفحه ای که به صفحه j اشاره می کند)و صفحه j می باشد و از ساختار ارتباطی محاسبه شده به دست می آید. hub(i) امتیاز Hub صفحه i است که با استفاده از الگوریتم HITS محاسبه می شود. در این جا مجموعه پایه همان صفحاتی است که تا کنون پیمایش شده اند. الگوریتم HITS بر روی این صفحات اعمال می شود و امتیاز Hub صفحه i محاسبه می گردد.

پس از آن که امتیاز هر یک از صفحات استخراج شده محاسبه شد، این صفحات در صف پیمایش قرار می گیرد و صف مرتب می گردد. سپس صفحه ای که در ابتدای صف قرار دارد از صف انتخاب می شود و عملیات فوق الذکر برای آن تکرار می شود.

نکته ای که در طراحی این پیمایشگر در نظر گرفته شده آن است که برای پیمایش موضوعی، تنها به کارگیری میزان ارتباط صفحات کافی نیست و باید صفحاتی برای پیمایش انتخاب شوند که علاوه بر مرتبط بودن دارای پیوند به صفحات دیگر نیز باشند. برای رفع این مشکل علاوه بر در نظر گرفتن میزان ارتباط یک صفحه با موضوع مورد نظر (که بر اساس ساختار ارتباطی تعیین می شود)، معیاری دیگری نیز برای انتخاب یک صفحه برای پیمایش در نظر گرفته می شود. این معیار همان امتیاز Hub می باشد که با استفاده از الگوریتم HITS محاسبه می شود. با این تفاوت که در الگوریتم HITS امتیاز Hub برای مجموعه پایه محاسبه می شود، اما در این جا امتیاز Hub را با استفاده از مجموعه صفحاتی که تا کنون پیمایش شده اند، محاسبه می کنیم. صفحه ای که امتیاز Hub بالایی دارد، به صفحات مرتبط با یک موضوع اشاره می کند و به این ترتیب مشکل فوق الذکر را برطرف می نماید.

از آن جا که روش پیشنهادی از جمله روش هایی است که از یک دانش پیش زمینه استفاده می کند، مانند روش های این گروه وابسته به کیفیت دانش پیش زمینه می باشد. همان گونه که در پیمایش موضوعی با استفاده از طبقه بندی کننده صفحات و پیمایش موضوعی مبتنی بر هستان شناسی، کیفیت پیمایش به طبقه بندی کننده و هستان شناسی استفاده شده بستگی دارد، کارایی روش پیشنهادی نیز به کیفیت ساختار ارتباطی به دست آمده وابسته است.

شبه کد الگوریتم پیشنهادی در شکل 1 آمده است.

```
// DLA based Focused Crawling Algorithm
Procedure DLA_Crawling_algorithm
    create Initial Crawl Set based on Crawl Topic
    fill the Crawl Queue with pages of Initial Set
    set the priority of all pages in Crawl Queue equally
    Crawl Set = Null
    while (not termination) do // a definite number of pages have been crawled
        page_i= pick the page at the head of Crawl Queue
        add page_i to Crawl Set
        PointedPageSet= extract the pages pointed by page_i
        for each page_j in PointedPageSet do
            hub(i)= compute hub score of page_i by applying HITS algorithm on Crawl
            Set r(i,j)= retrieve the relevance score of page_i and page_j from Relationship
                Structure
            Crawl_Score(page_j)= r(i,j) * hub (i)
            add page_j to Crawl Queue
        end for
        sort Crawl Queue
    end while
    output Crawl Set
```

شکل 1. شبه کد الگوریتم پیمایش موضوعی مبتنی بر DLA

3- نتایج شبیه سازی

در این بخش نتایج بدست آمده از شبیه‌سازی الگوریتم پیشنهادی ارائه می‌شود.

3-1- معیار ارزیابی پیمایشگر

دو شاخص اصلی که برای ارزیابی عملکرد پیمایشگر به کار می روند، عبارتند از: نرخ حاصل و فراخوان هدف.

- نرخ حاصل: این شاخص میانگین میزان ارتباط صفحات بازیابی شده را اندازه گیری می کند. نرخ حاصل با H(t) نشان داده می شود و پس از پیمایش t صفحه از رابطه (5) محاسبه می شود.

$$H(t) = \frac{1}{t}\sum_{i=1}^{t} r_i \qquad (5)$$

که r_i میزان ارتباط صفحه i با موضوع پیمایش می باشد. نرخ حاصل، مشابه با معیار ارزیابی دقت است که در کاربردهای بازیابی اطلاعات استفاده می شود.

- فراخوان هدف: فراخوان هدف، میزان صفحات مرتبط بازیابی شده به کل صفحات مرتبط موجود در وب را اندازه گیری می کند. از آنجا که مجموعه کل صفحات مرتبط در وب، مشخص نیست، مجموعه ای به عنوان مجموعه هدف تعریف می شود که با استفاده از آن معیار فراخوان هدف به عنوان تخمینی از فراخوان واقعی محاسبه می شود. برای تعریف مجموعه هدف، می توان از یک نمونه تصادفی از صفحات موجود در وب استفاده کرد. هرچند، نمی توان انتظار داشت که این مجموعه یک نمونه تصادفی واقعی از صفحات مرتبط وب باشد، اما تخمین مناسبی برای محاسبه میزان فراخوان فراهم می کند. این معیار نیز مشابه معیار فراخوان در کاربردهای بازیابی اطلاعات می باشد.

3-2- مدل شبیه سازی

برای شبیه‌سازی الگوریتم پیشنهادی و مقایسه آن با سایر روشها از مدل معرفی شده در [17] برای نشان دادن ساختار صفحات وب و چگونگی استفاده کاربران، استفاده شده است. اعتبار این مدل توسط Lui و همکاران [17] با استفاده از اطلاعات استفاده از وب چندین سایت وب بزرگ مانند مایکروسافت، تایید شده است.

جدول 1: پارامترهای شبیه‌سازی ها

حد آستانه ایجاد اتصال	0/7
تعداد کاربران	50000
تعداد اسناد	500
تعداد موضوع‌ها	5
T_c مقدار ثابت سند اولیه (صفحه اولیه سایت) در موضوعات مختلف	0/2
ΔM_t^c ضریب ثابت کاهش اشتیاق کاربر	-
ΔM_t^v ضریب متغیر کاهش اشتیاق کاربر	-
a_u پارامتر توزیع قانون-توانی توزیع احتمال علایق کاربران	1
f ضریب پاداش دریافتی از مشاهده یک سند	1/2
l ضریب جذب اطلاعات از یک سند توسط یک کاربر	0/5
m_m میانگین توزیع نرمال ΔM_t^v	5/97
s_m واریانس توزیع نرمال ΔM_t^v	0/25
m_t میانگین توزیع نرمال برای مقدار افزایش یک گره برای یک موضوع خاص	-
a_p پارامتر توزیع قانون-توانی توزیع احتمال وزنهای مطالب برای هر سند	3
s_t واریانس توزیع نرمال برای مقدار افزایش یک گره برای یک موضوع خاص	0/25
q ضریب کاهش علاقه کاربر	1
حداقل اشتیاق کاربر برای ادامه جستجو	0/2
تعداد صفحات در هرپارتیشن	25

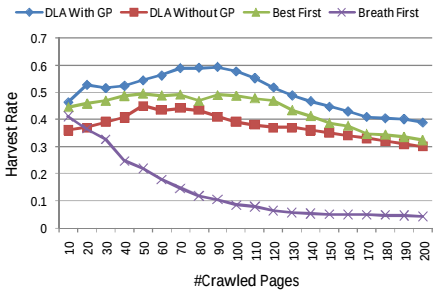
بر این اساس، در این مقاله مطابق با مدل رفتار کاربران، پروفایل علاقه کاربران بصورت توزیع قانون-توانی و توزیع محتوای صفحات وب بصورت توزیع نرمال در نظر گرفته شده است. سایر پارامترهای استفاده شده برای شبیه سازیهای انجام شده در این مقاله در جدول 1 نشان داده شده است.تعداد

صفحات پیمایش شده در هر یک از این آزمایشات برابر 200 در نظر گرفته شده است.

همچنین پارتیشن بندی گراف توسط نرم افزار Metis انجام شده است [16]. در این نرم افزار پارتیشن بندی با استفاده از روشهای چند سطحی انجام می‌شود. این نرم افزار از الگوریتمهایی برای پارتیشن بندی گراف استفاده می‌کند که در سریعترین زمان، زیر گراف‌هایی با کیفیت بالا ایجاد کند.

3-3- مقایسه نتایج

مهمترین معیار ارزیابی پیمایشگر موضوعی نرخ حاصل است که میانگین میزان ارتباط صفحات بازیابی شده را نشان می دهد. برای ارزیابی این معیار سه موضوع از میان موضوعات موجود در مدل شبیه‌سازی انتخاب‌شده و پیمایش برای هر یک از آنها به چهار روش انجام شده است.



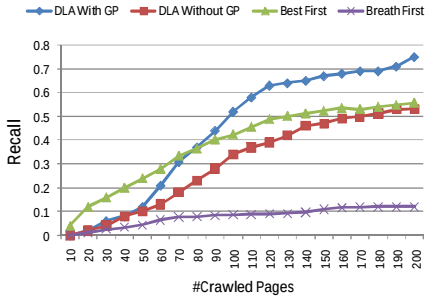
شکل 2. مقایسه نرخ حاصل برای چهار پیمایشگر

پیمایشگر اول، پیمایشگر عرض اول است و ضعیف ترین کارایی را در میان سایر پیمایشگرها دارد و به عنوان حد پایین برای ارزیابی عملکرد پیمایشگرها به کار می رود. پیمایشگر دوم، پیمایشگر BFS (Best First Search) است که نسبت به سایر پیمایشگرهای موضوعی بهتر عمل می کند. در این پیمایشگر اولویت پیوندها بر اساس میزان شباهت صفحه حاوی پیوند و مجموعه ای از صفحات مرتبط تعیین می شود. پیمایشگر سوم، پیمایشگر مبتنی بر DLA می باشد و پیمایشگر چهارم ترکیبی از DLA و پارتیشن بندی گراف می باشد.

همان طور که در شکل 2 دیده می شود، کارایی پیمایشگر عرض اول که یک پیمایشگر غیرموضوعی است، به سرعت کاهش می یابد و پس از پیمایش تعداد صفحات محدودی (200 صفحه)، میزان صفحات مرتبط بازیابی شده به صفر کاهش می یابد. در حالی که سه پیمایشگر موضوعی از کارایی بسیار بهتری برخوردارند و عملکرد پیمایشگرهای پیشنهادی بهتر از پیمایشگر BFS یا برابر با آن هستند.

شکل 3 فراخوان هدف نسبت تعداد صفحات مرتبط بازیابی شده به کل صفحات مرتبط را نشان می دهد.

در این آزمایش همچون آزمایش قبل پیمایش با استفاده از پیمایشگر عرض اول، پیمایشگر BFS، پیمایشگر مبتنی بر DLA و پیمایشگر ترکیبی مبتنی بر DLA و پارتیشن بندی گراف انجام شده و فراخوان هدف در مراحل مختلف پیمایش اندازه گیری شده است. همان طور که در شکل 3 مشاهده می شود، فراخوان هدف در پیمایشگر عرض اول ضعیف ترین و پیمایشگر مبتنی بر DLA و پارتیشن بندی گراف بهترین می باشد.

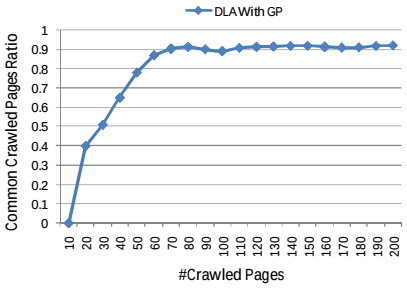


شکل 3. مقایسه فراخوان هدف برای چهار پیمایشگر

همانطور که در شکل 3 مشاهده می شود در هر چهار پیمایشگر پس از پیمایش تعداد مشخصی از صفحات فراخوان هدف به مقدار ثابتی میل می کند، چرا که تعداد صفحات مرتبط جدیدی که بازیابی می شود کاهش می یابد.

یکی از معیارهای دیگری که برای ارزیابی کارایی پیمایشگرهای موضوعی به کار می رود، میزان وابستگی آن به مجموعه آغازین می باشد.

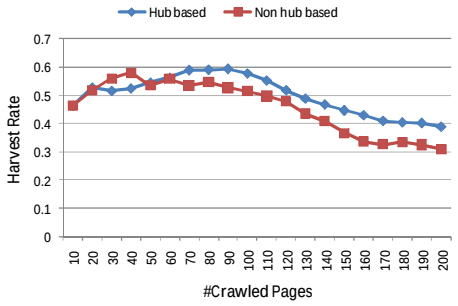
برای ارزیابی به صورت تصادفی دو مجموعه از صفحات آغازین انتخاب می کنیم.



شکل 4. وابستگی به مجموعه آغازین

همان طور که در شکل 4 دیده می‌شود، تعداد صفحات مشترک پیمایش شده به سرعت به 90 درصد میل می‌کند و این نتیجه نشان می‌دهد که پیمایشگر مبتنی بر DLA وابسته به چگونگی انتخاب مجموعه آغازین نمی‌باشند.

یکی از ویژگی‌های پیمایشگرهای پیشنهاد شده آن است که در تخصیص امتیاز پیمایش به پیوندها، علاوه بر نظر گرفتن میزان ارتباط صفحه حاوی پیوند با موضوع پیمایش، امتیاز Hub صفحه حاوی پیوند نیز در نظر گرفته می‌شود. نتایج این آزمایش برای پیمایشگر مبتنی بر DLA درشکل 5 نشان داده شده است. برای ارزیابی این ویژگی، پیماشگرهای موضوعی طراحی شده را در دو حالت مختلف با در نظر گرفتن امتیاز Hub و بدون آن اجرا می‌نماییم.



شکل 5. تاثیر به کارگیری امتیازHub در پیمایشگر مبتنی برDLA

همان طور که در شکل دیده می‌شود، در ابتدای پیمایش نتایج بهتری در حالت بدون در نظر گرفتن امتیاز Hub حاصل می شود. در حالی که با احتساب امتیاز Hub در مراحل میانی نتایج از حالت اول بهتر می شود.

4- نتیجه گیری

روشهای پیشنهادی از جمله روش هایی است که از یک دانش پیش زمینه (ساختار ارتباطی) استفاده می کند. این پیمایشگرها، یک پیمایشگر یادگیر هستند که می توانند از دانش یاد گرفته شده (ساختار ارتباطی) در پیمایش های متفاوت استفاده کنند. بنابراین نسبت به سایر پیمایشگرهایی که محاسبات و پردازش های خود را در هر بار پیمایش تکرار می کنند، به صرفه تر می باشند. هر چند کارایی روش‌های پیشنهادی به کیفیت ساختار ارتباطی به دست آمده وابسته می-باشند. همچنین این پیمایشگرها به دلیل به کارگیری امتیاز Hub در انتخاب صفحات، صفحاتی را انتخاب می کنند که حاوی پیوندهای بیشتری هستند و به این ترتیب دارای گزینه های بیشتری برای انتخاب می باشد.

نتایج ارزیابی پیمایشگرها پیشنهادی حاکی از بهبود عملکرد آن (نرخ حاصل و فراخوان هدف) نسبت به پیمایشگر Breath First یا Best First و عدم وابستگی آن به انتخاب مجموعه آغازین دارند. به این ترتیب استفاده از رفتار کاربران و پیوندهای بین صفحات در پیمایش موضوعی صفحات می تواند تاثیر بسزایی در بهبود نتایج داشته باشد.

مراجع

[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] Pinkerton, B., **"Finding What People Want: Experiences with the WebCrawler,"**, Proceedings of the 2nd International World Wide Web Conference (Chicago), 1994.

[4] Cho, J., Garacia-Molina, H., Page, L., ***"Efficient Crawling Through URL Ordering ,"***. *Comput Netw. 30*, pp. 161-172, 1998.

[5] Hersovici, M., Javoci, M., Maarek, Y.S., Pelleg, D., Shtalhaim, M., **"The Shark-Search Algorithm An Application: Tailored Web Site Mapping,"**, Proceedings of the 7th International World-Wide Web Conference, 1998.

[6] Page, L., Brin, S., Motwani, R., Winograd, T., **"The PageRank Citation Ranking: Bringing Order to the Web,"**, Stanford Publications, 1998.

[7] Debra, P., Post, R., **"Information Retrieval in the World Wide Web: Making Client-based Searching Feasible,"**, Proceedings of the 1st International World Wide Web Conference (Geneva), 1994.

[8] Chakrabarti, S., Van den Berg, M., Dom, B., **"Focused Crawling: A New Approach to Topic-Specific Web Resource Discovery,"**, Proceedings of the 8th International WWW Conference, Toronto, Canada, May 1999.

[9] Diligenti, M., Coetzee, F., Lawrence, S., Giles, C., Gori., M., **"Focused Crawling Using Context Graphs,"**, Proceedings of the 26th International Conference on Very Large Databases (VLDB 2000), Cairo, Egypt, September 2000.

[10] Su, C., et al, **"An Efficient Adaptive Focused Crawler Based on Ontology Learning ,"**, Proceedings of the Fifth International Conference on Hybrid Intelligent Systems (HIS'05), 2005.

[11] Aggarwal, C., Al-Garawi, F., Yu, P.,**"Intelligent Crawling on the World Wide Web with Arbitrary Predicates,"**, Proceedings of the 10th International World Wide Web Conference, Hong Kong, May 2001.

[12] Pant, G., Srinvasan, P., **"Learning to Crawl: Comparing Classification Schemes,"**, ACM Transactions on Information Systems, Vol. 23, No. 4, pp. 430–462, October 2005.

[13] 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.

[14] K. S. Narendra and M.A.L. Thathachar***, Learning Automata: An Introduction***, Prentice Hall, 1989.

[15] M. A. L. Thathachar and R. Harita Bhaskar, ***"Learning Automata with Changing Number of Action***s," IEEE Transactions on Systems Man and Cybernetics, vol. 17, no. 6, Nov. 1987, pp 1095-1100.

[16] G. Karypis and V. Kumar, "METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices,", Version 5.0pre2. 2007: Minneapolis.

[17] 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.