چهاردهمین کنفرانس ملی سالانه انجمن کامپیوتر ایران
دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) ایران، تهران – ۲۰ و ۲۱ اسفندماه ۱۳۸۷

یک روش جدید برای پیمایش سریع وب با استفاده از تکنیک یادگیری تقویتی

محمد جواد فتاحی حسن آباد
دانشگاه آزاد اسلامی واحد قزوین
mjfattahi@qazviniau.ac.ir

علیرضا رضوانیان
دانشگاه آزاد اسلامی واحد قزوین
rezvan@qazviniau.ac.ir

محمدرضا میبدی
دانشگاه صنعتی امیرکبیر تهران
meybodi@aut.ac.ir

چکیده: شبکه ارتباطات جهانی، در سال‌های اخیر روند سریع و روبه رشدی را طی می‌کند و تعداد صفحات وب نیز بطور روز افزون در حال افزایش است. در طی این سال‌ها همواره طراحی یک سیستم جستجوی مناسب با توانایی پاسخگویی مطلوب به نیازهای کاربران، یکی از چالش-های موجود در تکنولوژی‌های بازیابی اطلاعاتی می‌باشد. در این مقاله مسئله پیمایش وب برای پیدا کردن صفحات خاص به صورت جستجوی متمرکز بررسی شده است، و به هدف تسریع در پیمایش وب، روش جدیدی با استفاده از یادگیری تقویتی پیشنهاد شده است. نتایج آزمایشات با توجه محیط شبیه‌سازی برتری نسبی روش پیشنهادی را نشان می‌دهد.

واژه‌های کلیدی: بازیابی اطلاعات وب، موتور جستجو، پیمایش وب، پیمایشگر وب، یادگیری تقویتی

1- مقدمه

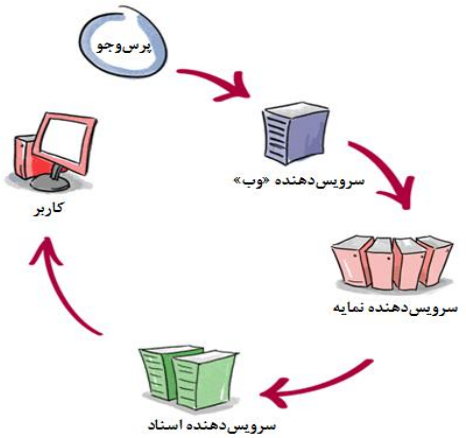
از ظهور شبکه ارتباطات جهانی، عمدتاً یافتن اطلاعات بصورت پویش فهرست‌ها و پیوندهای جمع‌آوری‌شده و یا مرتب‌شده با معیارهای سلیقه‌ای توسط افراد صورت می‌گرفته است، و در ابتدا موتورهای جستجوی خودکار مطرح و مورد نیاز نبودند، و با توجه به آنکه 40% از کاربران ورودی سایت‌های وب با توجه به اولین پیوندهای موجود در فهرست، نتایج موتورجستجو را دنبال می‌نمودند[7]، مدیران سایت‌های وب به ارسال سایت‌هایشان به این فهرست‌ها تشویق می‌شدند. اما امروزه انبوهی از آدرس‌های مختلف صفحات جدید با طیف وسیعی از منابع اطلاعاتی متنوع وجود دارد[12]. که بنابه گزارشات محققین حتی معتبرترین موتورهای جستجوی امروزی هم‌چون گوگل نیز تنها حجم محدودی از وب را پوشش داده و حتی حجم زیادی از اطلاعاتشان در ماه‌هایی از سال به روز نیست[16]. مشکل عمده موتورهای جستجو، نیاز به رسیدگی و اندازه تغییرات وب می‌باشد[4]. علاوه بر آن قسمتی از وب نیز به صورت پنهان پوشش نیافته است[14]. بنابراین به خاطر حجم بزرگ و ماهیت پویای وب روش‌ها و سیستم‌های سنتی پاسخگو نبوده و نیاز به یک سیستم نوین بازیابی اطلاعات توأم با روش‌های هوشمند و مکاشفه‌ای برای تسریع جستجو می‌باشد.

در این مقاله برای طراحی یک سیستم جستجوی مناسب و سریع، از یکی از روش‌های یادگیری ماشینی، به نام یادگیری تقویتی استفاده شده است. در روش یادگیری تقویتی با عاملی روبرو هستیم که از طریق سعی و خطا با محیط خود تعامل کرده و یاد می‌گیرد که چگونه یک عمل بهینه را انجام دهد[15]. در واقع در این مقاله روش یادگیری بر روی پیمایشگر وب اعمال شده است.

ادامه مقاله بدین صورت سازمان‌دهی شده است که ابتدا موتورهای جستجو و چرخه جستجو توضیح داده خواهد شد. سپس مروری بر ساختار پیمایشگرها و تکنیک یادگیری تقویتی خواهیم داشت، در ادامه چگونگی استفاده از یادگیری تقویتی در ساختار پیمایشگرها و الگوریتم پیشنهادی و در نهایت محیط شبیه‌سازی و نتایج ارزیابی ذکر شده است.

2- جستجو و پیمایش وب

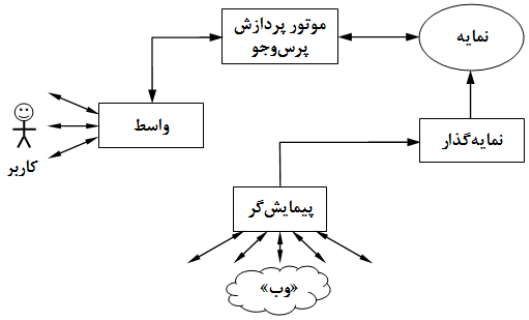
موتورهای جستجو به پایگاه‌های وبی گفته می‌شود که کاربران می‌توانند از آنها برای جستجوی مطالب موجود بر روی وب استفاده کنند. این موتورها، بعد از آنکه کاربر موضوع مورد جستجوی خود را در فرم مخصوصی که برای این کار تعبیه شده وارد می‌کند، لیستی از پایگاه‌های وبی که دربردارنده موضوع موردنظر کاربر هستند برای او جمع‌آوری و نمایش می‌دهند. چرخه ارائه پرسش تا دریافت مستندات (پایگاه‌های وب) در شکل 1 بصورت خیلی ساده نمایش داده شده است.



شکل 1: چرخه دریافت پرسش و ارائه نتایج به کاربر[19]

ابتدا کاربر نیاز اطلاعاتی خود را به صورت پرسشی مطرح می کند. ساختار این پرسش ساختار استاندارد و تعریف‌شده‌ای برای همه موتورهای جستجو نیست و با توجه به پیاده‌سازی‌های مختلف ممکن است متفاوت باشد. اما استفاده از کلمات کلیدی، متداول‌ترین نوع پذیرش پرسش است. سرویس‌دهنده وب موتور جستجو، پرسش را دریافت نموده و بعد از پردازش‌هایی مانند بررسی عملگرهای موجود در پرسش، استخراج کلمات کلیدی و ریشه‌یابی کلمات، پرسش را به سرویس‌دهنده نمایه به منظور مقایسه با مجموعه کلمات موجود در این سرویس‌دهنده ارسال می‌کند. بعد از انجام عمل مقایسه در این سرویس دهنده، مستندات مرتبط با پرسش کاربر مشخص می‌شوند. مشخصه این اسناد که شامل چکیده‌ای از متن، عنوان سند، آدرس سند و سایر ویژگی هاست، به سرویس‌دهنده اسناد ارسال می‌شود تا به کاربر نمایش داده شود.

بسیاری از موتورهای جستجو از یک معماری پیمایشگر-نمایه‌گذار متمرکز استفاده می‌کنند. پیمایش‌گرها برنامه‌هایی هستند که وب را پیمایش کرده و صفحات جدید یا به روز شده را به کارگزار می فرستند تا در آنجا نمایه‌گذاری شوند. از نمایه‌گذاری بدست آمده برای پاسخ‌دهی به پرس‌وجوی کاربران استفاده می‌شود.



شکل 2 : معماری نمادین موتور جستجوی پیمایشگر-نمایه‌گذار متمرکز [17]

در شکل فوق یک نمونه از این معماری مشاهده می‌شود. این معماری دو قسمت دارد. قسمت اول با کاربران سروکار دارد و شامل واسط کاربری و موتور پردازش جستجو است و قسمت دوم شامل واحدهای پیمایشگر و نمایه‌گذار است. برای مطالعه بیشتر در این زمینه می‌توان به [7] و [11] مراجعه نمود.

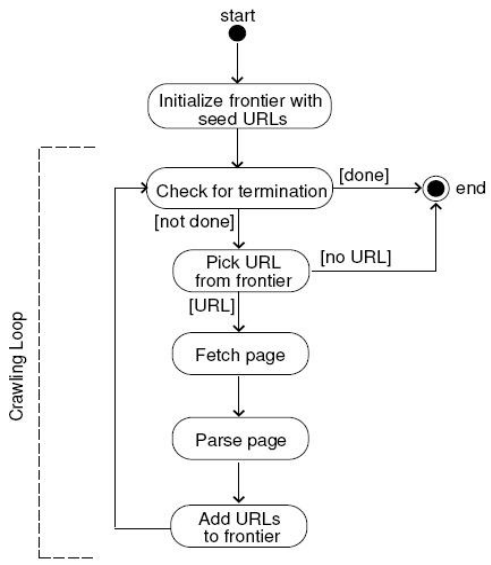
3- پیمایشگرهای وب

چنانچه ذکر شد به علت اندازه بزرگ و ماهیت پویای وب به جهت بهبود در کیفیت اطلاعات جمع‌آوری شده و نمایه‌گذاری شده می‌توان از بازیابی گزینشی اطلاعات استفاده نمود[7]، در این حالت معیارهای ارزیابی می‌تواند میزان نرخ به‌هنگام سازی صفحات، میزان علاقمندی مناطق به موضوعات، طبقه‌بندی بندی موضوعی، زمانی و یا کمی بر پایه پیمایش متمرکز باشد[6].

پیمایشگرهای وب برنامه‌هایی هستند که با پیمایش صفحه به صفحه ساختاری به شکل گراف از وب ایجاد می‌کنند. ایده پیمایشگر متمرکز اولین بار در [18] مطرح گردید. این‌گونه پیمایشگرها پیوندهای موجود در صفحات بازیابی شده را براساس موضوع و معیارهای ارزش‌یابی تعریف شده انتخاب می‌نمایند و براساس آن اولویت‌بندی (وزن‌دهی) کرده و پیوندهای با اولویت بالاتر را زودتر پیمایش می‌کنند[7]. اهمیت تسریع پیمایش وب باعث به وجود آمدن پیمایشگرهای توزیع شده است[10]، اما دقت در تنظیم پارامترهای موجود در این پیمایشگرها و ارزیابی نتایج به عنوان مشکل محسوب می‌شود [8]. با توزیعات مختلف صفحات وب نیاز به فرآیندهایی برای بررسی ابرداده‌ها بر روی صفحات وب می‌باشد که چگونگی عملکرد پیمایشگرهای مبتنی بر ابرداده‌ای در [1] به تفصیل شرح داده شده است.

هدف نهایی از طراحی پیمایشگرها بازیابی صفحات وب و ذخیره محتوای آنها می‌باشد. ساده‌ترین شکل یک پیمایشگر این است که از یک صفحه پیش‌فرض اولیه که به آن هسته نیز می‌گویند شروع می‌کند و سپس اطلاعات موجود در آن صفحه را جهت نمایه‌گذاری به موتور جستجو می‌فرستد و در نهایت از ابرپیوندهای موجود در آن صفحه استفاده می‌کند تا به صفحات دیگر مراجعه کند. و این کار ادامه می‌یابد تا به یک شرط خاتمه از پیش تعیین شده برسد. در حقیقت در اینجا ما از روش پیمایش متمرکز وب استفاده خواهیم نمود. در این روش که بسیاری از موتورهای جستجوی صاحب‌نام نیز از آن استفاده می کنند، پیمایشگر بر روی یک نوع خاص از صفحات تمرکز می کند. به عنوان مثال صفحاتی که بر روی یک موضوع خاص در مورد یک زبان مشخص، تصاویر، فایل های صوتی و ... اختصاص دارند. هدف از تمرکز، یافتن تعداد بسیاری از صفحات مورد نظر بدون استفاده از پهنای باند اضافی است.

روند کار یک پیمایشگر در شکل 3 نمایش داده شده است[3]



شکل 3 : روند کار یک پیمایشگر[3]

همان طور که ذکر شد یک پیمایشگر بعد از بازیابی یک صفحه ابرپیوندهای موجود در آن را یافته و این پیوندها را برای مراجعات بعدی در لیست مرتبی با عنوان سرلیست ذخیره می‌کند.

هر چرخه از پیمایش شامل انتخاب یک آدرس از سرلیست و سپس رفتن به صفحه مورد نظر برای جستجوی اطلاعات و نمایه‌سازی آنها و در نهایت افزودن آدرس‌های موجود در آن به سرلیست می‌باشد. قبل از اضافه کردن آدرس‌ها به سرلیست ممکن است یک ارزیابی برای مفید بودن آنها نیز صورت گیرد. عمل پیمایش ممکن است هنگامی که تعداد معینی صفحه پیمایش شد خاتمه یابد. همچنین ممکن است پیمایشگر با روبرو شدن با یک لیست خالی متوقف شود.

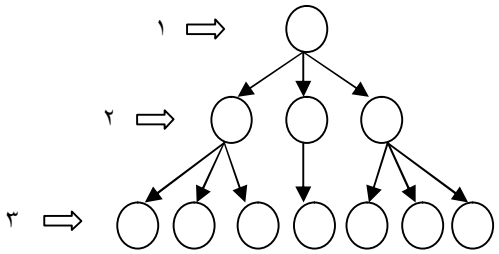
پیمایشگرها ممکن است بعنوان یک جستجو کننده گراف در نظر گرفته شوند، زیرا وب یک گراف بزرگ می‌باشد که صفحات آن گره‌های گراف هستند و پیوندهای موجود در صفحات می‌توانند بعنوان لبه‌های گراف در نظر گرفته شوند. پیمایشگر از یک گره شروع می‌کند و لبه‌ها را طی می‌کند تا به گره‌های دیگر برسد[2]

پیمایشگرها می توانند به صورت‌های متفاوتی در نظر گرفته شوند. پیمایشگری در اینجا در نظر گرفته شده است، پیمایشگر موضوعی نام دارد که سعی در یافتن صفحاتی دارد که با یک موضوع خاصی مرتبط هستند. به این صورت که ابتدا چند پیوند در مورد یک موضوع خاصی به پیمایشگر داده می‌شود، پیمایشگر بعد از مراجعه به صفحات مورد نظر و یافتن پیوند های جدید، سعی می کند فقط پیوندهایی را ذخیره و دنبال کند که با آن موضوع خاص مرتبط باشند. بعد از اینکه درباره آن موضوع به اندازه کافی پیمایش صورت گرفت ممکن یک موضوع دیگری به پیمایشگر داده شود تا آن را دنبال کند. جزئیات بیشتر در این زمینه را می‌توان در [3]، [6]، [9] و [14] یافت.

با توجه به موارد ذکر شده برای پیمایش گراف وب می‌توان از روش‌های مختلفی استفاده نمود که در ادامه دو نوع استراتژی برای پیمایش به صورت پیمایش اول سطح و پیمایش اول عمق ذکر شده است.

3-1- پیمایش اول سطح [i]

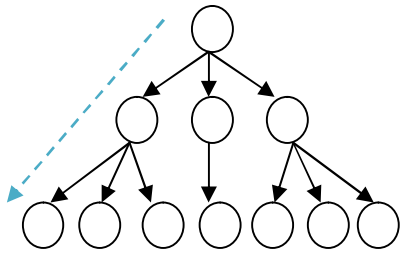
در این روش ابتدا کلیه پیوندهای موجود در اولین صفحه مورد پیمایش، ذخیره می‌شود و بعد به ترتیب به هر یک از این پیوندها مراجعه شده و صفحه متناظرشان واکشی می‌شود و مجدداً کلیه پیوندهای موجود در این صفحات ذخیره می‌شود و در نوبت‌های بعدی به آنها مراجعه می‌شود و این روند تکرار می‌شود. روند پیمایش اول سطح در شکل 4 ارائه شده است[16].



شکل ۴: پیمایش اول سطح

3-2- پیمایش اول عمق [ii]

در این روش از یک وب‌سایت اولیه شروع شده و تا عمق مورد نظر پیوندهای مجاور قبلی پیمایش می‌شود. در این روش بعد از پیمایش هر گره، نوبت به پیمایش گره همسایه در عمق بعد می‌رسد. اما در روش قبلی (پیمایش اول سطح) بعد از پیمایش هر گره، گره‌های همسایه با آن گره مورد پیمایش قرار می گرفت[16] روند پیمایش اول عمق در شکل 5 ارائه شده است.



شکل ۵: پیمایش اول عمق

4- مروری بر یادگیری تقویتی

یادگیری تقویتی یک روش یادگیری بدون ناظر می باشد. در این روش عامل با محیط اطراف خود تعامل کرده و از طریق سعی و خطا یاد می‌گیرد که چگونه اعمالی بهینه را برای رسیدن به هدف انجام دهد. فرض کنید عامل ما بتواند A عمل متفاوت را روی محیط انجام دهد و محیط نیز دارای R پاداش و S حالت ممکن باشد. در هر لحظه t عامل می‌تواند یکی از A عمل ممکن خود را انتخاب نماید و آن را روی محیط اعمال نماید . محیط در مقابل عمل انجام شده توسط عامل یک پاداشی را برای عامل در نظر می‌گیرد. اگر این پاداش مثبت باشد عامل در می‌یابد که عمل درستی را انجام داده است(در آن حالت) وگرنه اگر پاداش منفی دریافت کند، سعی می‌کند از انجام عملی که منجر به دریافت پاداش منفی شده است اجتناب نماید. همچنین عامل بعد از انجام یک عمل و دریافت پاداش آن وارد حالت بعدی در محیط می‌شود و باز یک عمل دیگر را انتخاب کرده و بر روی محیط اعمال می‌نماید.

شکل 6: عامل و محیط

بدین ترتیب عامل در محیط حرکت کرده و حالت‌ها و پاداش‌های مربوطه را به خاطر می‌سپارد و سعی می‌کند عملی انجام دهد که در نهایت تابع پاداش را ماکزیمم نماید.

اگر R_t را به‌عنوان مجموع پاداش‌هایی در نظر بگیریم که عامل با گذشت زمان t از محیط جمع‌آوری کرده است، R_t برابر خواهد بود با:

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \ldots = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \qquad 0 \le \gamma \le 1 \qquad (1)$$

که γ یک ضریب ثابت می‌باشد و بین صفر و یک قرار دارد و به حالت‌های نزدیک به حالت آغازین وزن بیشتری را نسبت می‌دهد.

اعمال عامل از قانونی مثل p تبعیت می‌کند که آنرا خط مشی می نامند. از آنجاییکه R_t یک متغیر تصادفی است لذا امید ریاضی آن تحت یک خط مشی خاص و برای یک حالت معین برابر خواهد بود با:

$$V^{\pi}(S_t) = E\{R_t \mid S_t, \pi\} = E\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid S_t, \pi\right\} \qquad (2)$$

هدف یادگیری تقویتی این است که یک خط مشی بهینه ای مثل π^* پیدا نماید به طوریکه مقدار امید ریاضی فوق را برای تمامی حالات بیشینه نماید[15].

5- استفاده از یادگیری تقویتی در پیمایش وب

همانطور که گفته شد پیمایشگرها، عامل‌هایی هستند که ابرپیوندهای موجود در صفحات وب را کشف نموده و به منظور یافتن اسناد با موتورهای جستجو مجتمع شده‌اند و همچنین به عنوان عاملی از تکنیک‌های جستجوی اطلاعات، پیمایشگرها عناصری حیاتی در ایجاد خودکار دانش‌های پایه هستند. در اینجا مسئله جستجوی وب برای پیدا کردن صفحاتی با عنوان یا موضوع خاص با استفاده از پیمایشگرها (پیمایش متمرکز) ذکر شده است.

در این روش پیمایشگر باید با استفاده از یادگیری تقویتی فقط پیوندهای مرتبط با یک موضوع را بیابد و اطلاعات آن صفحات را بازیابی کند. به این ترتیب پیمایشگر در هنگام یافتن پیوندها بایستی پیوندهای مفید را انتخاب و نگهداری کند. یعنی اگر پیوندی را انتخاب کند که منجر به موضوع بی ربطی گردد، پیمایشگر پاداش منفی دریافت می‌کند. در مقابل اگر پیوند خوبی را انتخاب کرده باشد، پاداش مثبت می‌گیرد. علاوه بر این ممکن است پیمایشگر یک پیوندی را انتخاب کند که به صفحه بی‌ربطی منجر شود، ولی بعد از چند مرحله در این راستا پیوندی یافت شود که به موضوع ما ارتباط داشته باشد در این حالت نیز پیمایشگر پاداش مثبت می‌گیرد یا به اصطلاح بطور تاخیری پاداش خواهد گرفت. پس پاداش‌ها دو نوع هستند یکی پاداش های آنی و دیگری پاداش های تاخیری[2] [13].

بنابراین می‌توان مولفه‌های یادگیری تقویتی را برای آن بصورت زیر در نظر گرفت:

1) پاداش‌ها (R): پاداش‌های آنی و پاداش‌های آینده

2) "عمل"(A): تعقیب (پیمایش) یک ابرپیوند خاص

3) "حالت"(S): مجموعه اسناد هدفی است که باید کاوش شوند. یا مجموعه پیوندهایی که یافته شده‌اند.

6- روش پیشنهادی به منظور پیمایش سریع وب

با توجه به حجم زیاد مصرفی توسط روش اول سطح و پرهزینه‌تر بودن آن [9]، در این روش از استراتژی پیمایش اول عمق استفاده شده است. در این حالت ساختار وب بصورت گرافی متصور شده است که هر پیوند معرف یک گره در گراف خواهد بود. در ابتدا برای همه گره‌ها یک مقدار اولیه‌ای در نظر گرفته می‌شود، که میزان مفید بودن آن گره‌ها را نمایش می‌دهد و در ابتدا این مقدار اولیه برای همه گره‌ها یکسان است. در هر مرحله با حرکت پیمایشگر در بین گره‌ها مقادیر آن‌ها با توجه به میزان مفید بودن یا غیر مفید بودن، بروز رسانی خواهد شد. هدف پیمایشگر این است که صرفا گره هایی را انتخاب نماید تا مجموع پاداش‌هایش را بیشنه کند.

ایده اصلی که در از آن در اینجا استفاده شده است، استفاده از خاصیت ارث بری پدر-فرزندی (عمقی در طول یک شاخه) و ارث بری برادری (گره‌هایی از یک پدر) بطور همزمان در گراف می‌باشد.

6-1- ارثبری پدر-فرزندی

بدین صورت است که بعد از اینکه پیمایشگر یک پیوند (گره) را برای پیمایش انتخاب نمود، چنانچه آن پیوند، پیوند مفیدی بود و به صفحات مرتبط با موضوع منجر شد، میزان مفید بودن کلیه فرزندان آن گره جاری نیز با نسبت خاصی افزایش یابد، چه فرزندانی که از آن گره تولید شده‌اند و چه فرزندانی که قرار است بعداً از آن تولید شوند. و چنانچه غیر مفید بود، از میزان مفید بودن فرزندان آن هم کاسته می‌شود. زیرا اگر یک صفحه وب ارتباط چندانی با موضوع مورد جستجوی نداشته باشد، به احتمال زیاد صفحاتی که از طریق این صفحه قابل

دسترسی هستند نیز ارتباط چندانی با موضوع نخواهند داشت.

2-6- ارث بری برادری

در این روش بعد از اینکه پیمایشگر یک پیوند (گره) را برای پیمایش انتخاب نمود، چنانچه آن پیوند، پیوند مفیدی بود و به صفحات مرتبط با موضوع منجر شد، میزان مفید بودن کلیه برادران آن گره جاری نیز با نسبت خاصی افزایش داده شده و چنانچه غیر مفید بود، از میزان مفید بودن برادران آن هم کاسته می‌شود. همانطور که گفته شد پیوندهایی برادر هستند که حاصل از پیمایش یک گره واحدی باشند. به عبارتی پیوندهایی هستند که در داخل یک صفحه وب قرار گرفته‌اند. در اینجا فرض بر این است که یک صفحه معمولاً به صفحاتی اشاره می‌کند که به میزان زیادی با خودش مشابهت موضوعی دارند. به غیر بعضی سایت‌ها مانند سایت‌های تبلیغاتی و..... که با اعمال کردن پاداش تاخیری نیز این مشکل حل می‌شود.

در نتیجه با استفاده از دو استراتژی فوق به طور همزمان، میزان مفید بودن صفحات نامرتبط به سرعت کاهش یافته و دیگر پیمایشگر آنها را انتخاب نخواهد نمود. و از طرفی با سرعت بیشتری پیمایشگر صفحات مرتبط را انتخاب خواهد نمود. روند انجام این الگوریتم پیشنهادی را می‌توان برای پیمایشگر بصورت زیر بیان کرد:

1) یک مقدار اولیه فرضی بعنوان میزان مفید بودن برای گره‌ها (پیوند-ها) انتخاب کنید. همچنین چند پیوند اولیه بعنوان هسته در لیست سرلیست پیمایشگر قرار دهید.

2) یک پیوند را از سرلیست انتخاب نموده و سپس با استفاده از پیمایشگر، صفحه مرتبط به آن را پیمایش کرده و کلیه پیوندهایی که در صفحه مورد نظر قرار دارند را به سرلیست اضافه نمایید. همچنین بطور همزمان با بررسی مطالب موجود در آن، در صورت نیاز اطلاعات موجود در آن صفحه را جهت نمایه‌گذاری به موتور جستجو بفرستید.

3) بسته به میزان مفید بودن گره پیمایش شده، میزان مفید بودن برادران و فرزندان آن را بروزرسانی نمایید.

4) تا زمانی که سرلیست خالی نشده است از مرحله 2 تکرار نمایید.

7- شبیه‌سازی و ارزیابی نتایج

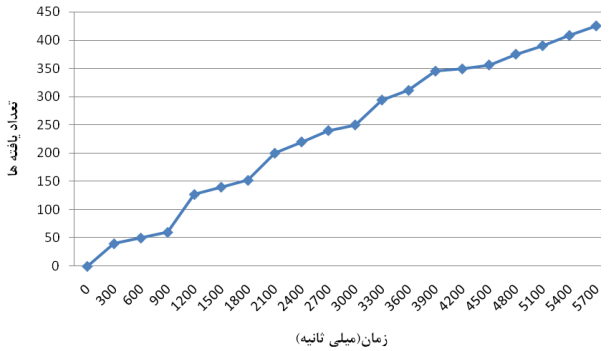
برای ایجاد یک محیط شبیه‌سازی و محاسبه دقیق‌تر زمان در حین آزمایشات، از وب‌سایت لینکستان[5] بعنوان هسته اولیه استفاده شده است و کلیه صفحات قابل دسترسی از آن تا عمق 5 و به تعداد 4866 صفحه بر روی کامپیوتر ذخیره شده است، تا انجام و تکرار آزمایشات به صورت برون‌خطی صورت پذیرد. مطالب قابل دسترسی از این سایت بصورت زیر می باشند:

فهرست مطالب		
آموزش دانشگاهها ، مدارس ...	اخبار و رسانه روزنامه ، رادیو و تلوزیون ...	اینترنت ایمیل ، جستجو ، چت ...
بهداشت و سلامتی پزشکی ، تغذیه ...	تجارت و بازار بازاریابی ، کاریابی ...	تفریح و سرگرمی آلبوم عکس ، جوک ...
جامعه و فرهنگ اشخاص ، تاریخ ...	علمی طبیعت ، مراجع ، حیوانات ...	کامپیوتر برنامه نویسی ، نرم افزار ...
وبلاگ آموزشی ، اجتماعی ...	ورزش باشگاهها ، فدراسیونها ...	هنر سینما ، موسیقی ...

شکل 7 : لیست مطالب موجود در وب‌سایت لینکستان

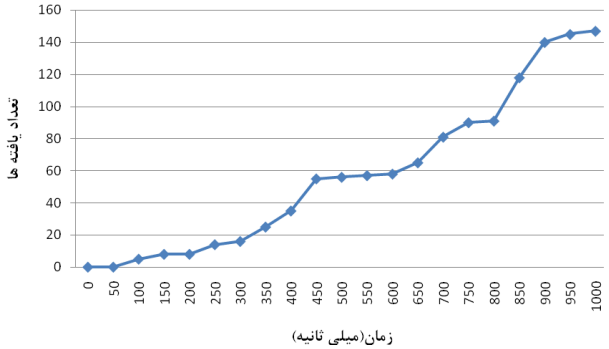
جهت انجام آزمایشات و بررسی نتایج از سه نوع پیمایشگر اول سطح، اول عمق با یادگیری تقویتی و پیمایشگر پیشنهادی استفاده و مورد مقایسه قرار گرفته است.

در نمودارهای زیر منحنی تعداد یافته‌ها را بر حسب زمان در هنگام پیمایش صفحات برای لغت فوتبال به ترتیب برای پیمایشگر اول سطح در شکل 8، اول عمق با یادگیری تقویتی در شکل 9 و پیمایشگر پیشنهادی در شکل 10 ارائه شده است.



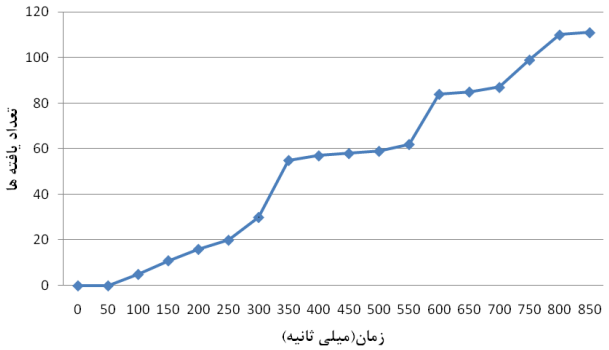
شکل 8: استفاده از روش اول سطح و بدون استفاده از یادگیری تقویتی

کل زمان لازم برای پیمایش صفحات وب در روش اول سطح 5689 میلی‌ثانیه و تعداد کل یافته‌ها 421 می‌باشد.



شکل 9: روش اول عمق و استفاده از یادگیری تقویتی

کل زمان لازم برای پیمایش صفحات وب در روش اول عمق و استفاده از یادگیری تقویتی 1038 میلی‌ثانیه و تعداد کل یافته‌ها 144 می‌باشد.

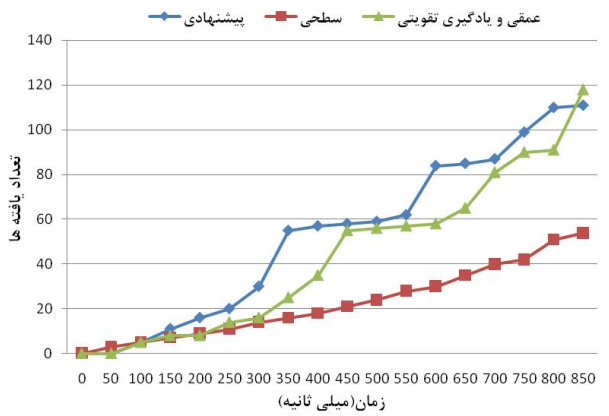


شکل 10: روش پیشنهادی

کل زمان لازم برای پیمایش صفحات وب در روش پیشنهادی 845 میلی‌ثانیه و تعداد کل یافته‌ها 116 می‌باشد.

همانطور که مشاهده می شود، روش پیشنهادی و اول عمق تعداد یافته های کمتری نسبت به روش اول سطح دارند، و این امر به این دلیل است که، این روش ها صرفا مرتبط ترین صفحات را بازیابی می کنند و زمان پیمایش آنها به مراتب از روش اول سطح کمتر می باشد. در حالی که در روش اول سطح، اگر در هر کجای صفحه، عبارتی مشابه با عبارت جستجو شده را پیدا کند، آن صفحه را بعنوان یکی از یافته هایش برمی گزیند، در حالی که ممکن است آن صفحه هیچ ربطی به موضوع مورد جستجوی ما نداشته باشد.

در نهایت نتایج مقایسات را برای هر سه روش می‌توان در شکل 10 مشاهده نمود که به خوبی کارایی روش پیشنهادی را نسبت به دو روش دیگر نشان می‌دهد.



شکل 10: مقایسه کارایی تعداد یافته‌ها با توجه به زمان برای سه روش مختلف

از طرفی همانطور که در نمودارها مشاهده می‌شود در روش اول سطح چون همه صفحات پیمایش می‌شوند تعداد یافته‌ها بر حسب زمان بصورت خطی در حال افزایش است و حافظه فراوانی مورد نیاز است. ولی در روش‌هایی که از یادگیری تقویتی استفاده می‌شود از صفحات نامرتبط صرفنظر خواهد شد، به همین دلیل در نمودار شکل‌های 8 و بخصوص شکل 9 ممکن است در یک بازه‌ای از زمان تعداد یافته‌های ما تغییر چندانی نکند ولی در عوض در بازه دیگر تعداد یافته‌ها به سرعت افزایش یابد.

همچنین برای مقایسه بهتر کارایی از هر گروه از مطالب موجود در جدول شماره 1 یک لغت برای جستجو انتخاب شده است. نتایج دقیق آزمایشات در جدول شماره 2 ذکر شده است. برای مقایسه این روش‌ها از معیارهایی نظیر درصد یافته‌ها به کل صفحات پیمایش شده و همچنین کل زمان پیمایش استفاده شده است. همانطور که در جداول مربوطه نشان داده شده است، هر چند که روش پیشنهادی در مقایسه با دو روش دیگر تعداد یافته‌های کمتری دارد، ولی این یافته‌ها بیشتر به موضوع مرتبط می‌باشند. همچنین این روش علاوه بر اینکه قادر است در زمان خیلی کمتری از سایر روش‌ها اکثر صفحات مرتبط را بازیابی نماید، برای معیار درصد یافته‌ها به کل صفحات جستجو شده نیز از روش‌های دیگر بیشتر می باشد.

جدول (1) نتایج میانگین کارایی برای پیمایشگرهای مختلف

معیار کارایی	روش سطحی- کل صفحات	روش عمقی و یادگیری تقویتی	روش پیشنهادی
تعداد صفحات جستجو شده	4866	1227	572
میانگین تعداد یافته‌ها	337	181	131
میانگین زمان جستجو	6110.45	1807.81	1378
درصد یافته به کل جستجو	% 6.93	% 11.71	% 16.87

8- نتیجه‌گیری

در سیستم‌های بازیابی اطلاعاتی وب امروزی با توجه به حجم عظیم اطلاعات وب و ماهیت پویای آن استفاده از روش‌های سنتی کارایی نداشته و منسوخ شده‌اند، در عوض استفاده از روش‌های هوشمند مورد توجه است. در این مقاله استفاده از یادگیری تقویتی در ساختار پیمایشگر پیشنهاد شد که ایده اصلی که آن استفاده از خاصیت ارث بری پدر-فرزندی (عمقی در طول یک شاخه) و ارث بری برادری (گره هایی از یک پدر) بطور همزمان در گراف می‌باشد. که ارزیابی نتایج شبیه‌سازی حاکی از بهبود کارایی سیستم جستجو و برتری روش سنتی بوده و از سرعت بالایی برخوردار شده است.

مراجع

[1] Hai Dong, F.K. Hussain, E. Chang, "State of the art in metadata abstraction crawlers", IEEE International Conference on Industrial Technology, (ICIT 2008), Pp. 1-6, 21-24 April 2008.

Engineering, Faculty of Engineering Kasetsart University, Bangkok 10900 Thailand, 2002.

[12] History and Growth of the Internet, Available: http:/www.internetworldstats.com/stats.htm ,2006

[13] Gautam pant and Filippo menczer, "MySpiders: Evolve Your Own Intelligent Web Crawlers" , Department of Management Sciences, The University of Iowa, Iowa City, IA 52242, Autonomous Agents and Multi-Agent Systems, 5, 221–229, 2002.

[14] A. Ibrahim, S.A. Fahmi, S.I. Hashmi, Ho-Jin Choi, "Addressing Effective Hidden Web Search Using Iterative Deepening Search and Graph Theory", IEEE 8th International Conference on Computer and Information Technology Workshops, (CIT Workshops 2008), Pp. 145 - 149, Sydney, QLD, 8-11 July 2008.

[15] R. S. Sutton, A. G. Barto, "Reinforcement Learning: An Introduction", MIT Press, Cambridge, March, 1998.

[16] Vladislav Shkapenyuk and Torsten Suel, "Design and implementation of a high-performance distributed web crawler", In Proceedings of the 18th International Conference on Data Engineering (ICDE), pages 357 – 368, San Jose, California, February 2002.

[17] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, "Modern Information Retrieval", ACM Press/ Addison-Wesley, 1999.

[18] Soumen Chakrabarti, Martin van den Berg, and Byron Dom, "Focused crawling: a new approach to topic-specific web resource discovery", Computer Networks, 31(11–16):1623–1640, 1999.

[19] Google Corporate Information: Technology Overview, http://www.google.com/corporate/tech.html , 2008.

[2] J. Rennie and A. McCallum ,"Using reinforcement learning to spider the web efficiently", In Proceedings International Conference on Machine Learning (ICML), 1999.

[3] A. A. Barfourosh, H. R. Motahary Nezhad, M. Onderson and D. Perlis, "Information Retrieval in WWW and Active Logic: Survey and problem definition", Technical Report in Department of Computer Science of University of Maryland and Institute of Advance Computer Science in University of Maryland, USA, CS-4291, 2002.

[4] Web Statistics (size and growth), Available: http://wcp.oclc.org/stats.html , 2002.

[5] Linkestan, Available : www.linkestan.com , 2008.

[6] Milad Shokouhi, Pirooz Chubak, Farhad Oroumchian, Hassan Bashiri, "Designing and implementation of "regional crawler" as a new strategy for crawling the web", IADIS International Conference e-Society 2004.

[7] Alireza Rezvanian, Hassan Bashiri, "Persian Web Characterization Based on SABA Web Crawler", Proceedings of 12th Annual CSI Computer Conference of Iran(CSICC'2007),Pages 852-858, Shahid Beheshti University, Tehran, Iran, February 2007.

[8] M. Nasri, S. Shariati, M. Sharifi, "Availability and Accuracy of Distributed Web Crawlers: A Model-Based Evaluation", Second UKSIM European Symposium on Computer Modeling and Simulation, (EMS 2008),Pp. 453 - 458, 8-10 Sept. 2008.

[9] Jafar Habibi, Nafise Fekr azad, "Evaluation of web crawler performance", Proceedings of 10th Annual CSI Computer Conference of Iran, 2004

[10] Yuan Wan, Hengqing Tong, "URL Assignment Algorithm of Crawler in Distributed System Based on Hash", IEEE International Conference on Networking, Sensing and Control, (ICNSC 2008), Pp. 1632 - 1635, 6-8 April 2008.

[11] Niran Angkawattanawit and Arnon Rungsawang ," Learnable Crawling: An Efficient Approach to Topic-specific Web Resource Discovery", Massive Information & Knowledge Engineering Department of Computer

جدول (2) نتایج کارایی برای پیمایشگرهای مختلف

واژه کلیدی	روش جستجو	کل صفحات جستجو شده	تعداد یافته ها	زمان جستجو (میلی ثانیه)	درصد یافته ها به کل صفحات جستجو شده
دانشگاه	سطحی - تمامی صفحات	4866	965	6375	19.83
	عمقی و یادگیری تقویتی	3047	752	4468	24.68
	پیشنهادی	2120	602	3703	28.40
روزنامه	سطحی - تمامی صفحات	4866	384	6078	7.89
	عمقی و یادگیری تقویتی	2283	212	2937	9.29
	پیشنهادی	716	74	1515	10.34
ایمیل	سطحی - تمامی صفحات	4866	449	6375	9.23
	عمقی و یادگیری تقویتی	993	159	1687	16.01
	پیشنهادی	426	128	1281	30.05
پزشک	سطحی - تمامی صفحات	4866	350	6015	7.19
	عمقی و یادگیری تقویتی	422	90	890	21.33
	پیشنهادی	272	73	953	26.84
خرید و فروش	سطحی - تمامی صفحات	4866	12	5750	0.25
	عمقی و یادگیری تقویتی	159	1	453	0.63
	پیشنهادی	71	1	640	1.41
سرگرمی	سطحی - تمامی صفحات	4866	134	6140	2.75
	عمقی و یادگیری تقویتی	806	37	1234	4.59
	پیشنهادی	251	32	890	12.75
دانشجو	سطحی - تمامی صفحات	4866	680	6296	13.97
	عمقی و یادگیری تقویتی	1328	328	1953	24.70
	پیشنهادی	805	249	1703	30.93
طبیعت	سطحی - تمامی صفحات	4866	95	6109	1.95
	عمقی و یادگیری تقویتی	912	24	1218	2.63
	پیشنهادی	233	11	937	4.72
برنامه نویسی	سطحی - تمامی صفحات	4866	92	5953	1.89
	عمقی و یادگیری تقویتی	252	9	578	3.57
	پیشنهادی	70	1	640	1.43
فوتبال	سطحی - تمامی صفحات	4866	397	6218	8.16
	عمقی و یادگیری تقویتی	2130	289	2875	13.57
	پیشنهادی	856	212	1750	24.77
سینما	سطحی - تمامی صفحات	4866	154	5906	3.16
	عمقی و یادگیری تقویتی	1166	92	1593	7.89
	پیشنهادی	472	66	1156	13.98

[i] Breath First Crawling
[ii] Depth First Crawling