استفاده از خوشه‌بندی و مدل مارکوف جهت پیش‌بینی درخواست آتی کاربر در وب

سیامک عبداله‌زاده[۱]، دانشجوی کارشناسی ارشد، محمدعلی بالافر[۲]، استادیار ، لیلی محمدخانلی[۳]، دانشیار

۱، ۲ و ۳- دانشکده مهندسی برق و کامپیوتر - دانشگاه تبریز - تبریز - ایران
[1]s_abdollahzadeh90@ms.tabrizu.ac.ir, [2]balafarila@yahoo.com, [3]l-khanli@tabrizu.ac.ir

چکیده: تاکنون روش‌های مختلفی جهت کاهش تأخیری که توسط کاربر مشاهده می‌شود، ارائه شده است. پیش‌واکشی وب یکی از روش‌های کاهش این تأخیر است. این روش از یک الگوریتم پیش‌بینی استفاده می‌کند تا فعالیت‌های آتی کاربر را پیش‌بینی کند. در این مقاله، روشی برای پیش‌بینی درخواست آتی کاربران ارائه شده است که قابل استفاده برای پیش‌واکشی وب است. در این روش، از ترکیب یک مدل مارکوف با خوشه‌بندی، مدلی جهت پیش‌بینی درخواست آتی کاربر ایجاد شده است. همچنین، از خواص زمانی دسترسی‌ها نیز جهت پیش‌بینی استفاده شده است. نتایج پیاده‌سازی بیانگر بهبود پیش‌بینی‌ها نسبت به مدل مارکوف است.

واژه‌های کلیدی: پیش‌واکشی وب، داده‌کاوی، کاربرد کاوی وب، خوشه‌بندی، مدل مارکوف

Using Clustering and Markov Model in Predicting Web Users' Next Request

S. Abdollahzadeh, Graduate student[1], M. A. Balafar, Assistant professor[2] and L. Mohammad Khanli, Associate professor[3]

1, 2 & 3- Faculty of Electrical and Computer Engineering, University of Tabriz, Tabriz, Iran,
[1]s_abdollahzadeh@ms.tabrizu.ac.ir, [2]balafarila@yahoo.com, [3]l-khanli@tabrizu.ac.ir}

Abstract: Different techniques have been proposed for reducing user perceived latency. Web prefetching is one of techniques aimed at reducing this latency. This technique uses a prediction algorithm for predicting user's future requests. In this paper, a method is proposed for predicting user's next requests. This method can be applied for web prefetching. In the proposed method, a model which is a combination of Markov model and clustering is proposed for prediction. Time features of accesses in the website are also considered in the proposed model. The result of experiments shows the improvement of proposed model with respect to the Markov model.

Keywords: Web prefetching, data mining, web usage mining, clustering, Markov model.

تاریخ ارسال مقاله: ۹۲/۱۰/۱۸
تاریخ اصلاح مقاله: ۹۳/۰۱/۱۳
تاریخ پذیرش مقاله: ۹۳/۰۱/۲۴
نام نویسنده مسئول: سیامک عبداله‌زاده
نشانی نویسنده مسئول: ایران - تبریز - بلوار ۲۹ بهمن - دانشگاه تبریز - دانشکده مهندسی برق و کامپیوتر

۱- مقدمه

باوجود کارهای تحقیقی و توسعه‌های زیاد در زیرساخت‌های وب و شبکه، تأخیر مشاهده‌شده توسط کاربر همچنان بالاست [۱]. از طرفی از دید کاربر، تأخیر می‌تواند به‌عنوان معیار اصلی کارایی وب در نظر گرفته شود. در سال‌های اخیر تلاش‌های زیادی جهت کاهش تأخیر مشاهده‌شده توسط کاربر انجام شده است. محبوب‌ترین روش‌هایی که تاکنون ارائه شده‌اند عبارت‌اند از: تکثیر جغرافیایی، حافظه‌ی نهان در وب و پیش‌واکشی وب. حافظه‌ی نهان در وب بر پایه‌ی استفاده‌ی مجدد اشیائی است که قبلاً ملاقات شده‌اند و اکنون در حافظه‌ی نهان ذخیره شده‌اند. بنابراین برخی درخواست‌ها به‌صورت محلی پاسخ داده می‌شوند و درنتیجه تأخیر مشاهده‌شده توسط کاربر کاهش می‌یابد. باوجوداینکه کارایی وب به‌وسیله‌ی حافظه‌ی نهان در وب افزایش پیدا می‌کند، ولی مزایای آن محدود است [۲]. پیش‌واکشی وب که بر اساس حافظه‌ی نهان است، یکی از روش‌هایی است که برای کاهش بیشتر تأخیر مشاهده‌شده توسط کاربر ارائه شده است [۳].

در پیش‌واکشی وب، اشیاء، قبل از تقاضای کاربر درخواست می‌شوند. برای این کار نیاز به روشی وجود دارد که با کشف الگوهای دسترسی، درخواست‌های بعدی کاربر را به‌صورت صحیح پیش‌بینی کند. پیش‌بینی اشتباه در پیش‌واکشی، موجب افزایش ترافیک شبکه و بار سرویس‌دهنده می‌شود. به همین دلیل، الگوریتم پیش‌بینی هسته‌ی اصلی پیش‌واکشی وب است [۴].

تاکنون تعداد زیادی روش جهت پیش‌بینی درخواست آتی کاربر، ارائه شده است. الگوریتم‌های اولیه از محبوبیت اشیا برای تعیین پیش واکشی شیء استفاده می‌کردند. امروزه بیشتر پژوهش‌ها، با استفاده از دنباله‌ی درخواست‌های قبلی، درخواست آتی کاربر را پیش‌بینی می‌کنند. روش‌های ارائه‌شده برای پیش‌بینی با استفاده از دنباله‌ی دسترسی معمولاً بر اساس مدل مارکوف [۶،۵،۳] هستند و یا از روش‌های داده‌کاوی [۸،۷] استفاده می‌کنند. در این مقاله یک روش پیش‌بینی با استفاده از ترکیب خوشه‌بندی و مدل مارکوف جهت استفاده در پیش‌واکشی وب ارائه شده است. به دلیل اینکه پیش‌بینی درست یک شیء (نه‌فقط صفحه) می‌تواند تأخیر مشاهده‌شده توسط کاربر را کاهش دهد، روش پیشنهادی توانایی پیش‌بینی اشیا را نیز دارد. در این روش، ابتدا نشست‌های کاربران خوشه‌بندی می‌شوند. خوشه‌بندی نشست‌ها فقط با استفاده از صفحات انجام می‌شود. سپس یک مدل مارکوف برای هر خوشه ایجاد می‌شود و از این مدل برای پیش‌بینی درخواست‌های آتی کاربران استفاده می‌شود. در مدل مارکوف سایر اشیاء نیز وجود دارند که امکان پیش‌بینی آن‌ها نیز توسط مدل فراهم می‌شود. به‌علاوه در پیش‌بینی، خواص زمانی درخواست‌ها نیز در نظر گرفته شده است. به دلیل ماهیت پویای وب، اشیاء جدیدی که به وب‌سایت اضافه می‌شوند، اهمیت بیشتری برای کاربران وب‌سایت دارند [۹].

ادامه‌ی مقاله بدین‌صورت است. بخش ۲ کارهای پیشین ارائه‌شده را در بر می‌گیرد. بخش ۳ روش پیشنهادی مقاله را ارائه می‌دهد. بخش ۴ ارزیابی کار پیشنهادی را نشان می‌دهد. بخش ۵ نتیجه‌گیری کار پیشنهادی است.

۲- کارهای پیشین

تاکنون الگوریتم‌های مختلفی جهت پیش‌بینی درخواست آتی کاربر ارائه شده است. بیشتر این پژوهش‌ها بر روی پیش‌بینی درخواست بعدی با استفاده از دنباله‌ی درخواست‌های قبلی کاربر است. معمولاً این روش‌ها از داده‌ی موجود در داده‌ی ثبت[1] دسترسی سرویس‌دهنده‌ها یا پیشکارها[2] جهت پیش‌بینی استفاده می‌کنند. به همین دلیل، این الگوریتم‌ها، نیاز به شناسایی کاربر درخواست‌کننده را دارند تا بتوانند دنباله‌ی درستی از دسترسی‌ها را داشته باشند.

DG (Dependency Graph) [۳] اغلب به‌عنوان معیاری برای کارایی دیگر الگوریتم‌ها در نظر گرفته می‌شود. الگوریتم پیش‌بینی DG یک گراف وابستگی ایجاد می‌کند که الگوی دسترسی به اشیاء را نشان می‌دهد. گراف، یک گره برای هر شیء که دسترسی به آن صورت گرفته، دارد. یک یال از گره A به گره B وجود دارد اگر و فقط اگر در یک نقطه‌ی زمانی، در طول w درخواست پس از دسترسی به A، B نیز درخواست شده باشد. w، اندازه‌ی پنجره‌ی پیش‌بینی است. وزن یال نیز تعداد دسترسی به B پس از دسترسی به A در پنجره‌ی پیش‌بینی تقسیم بر تعداد دسترسی به A است.

با تمرکز روی پیشکار، در [۵] الگوریتم معروف PPM (Prediction by Partial Match) پیشنهاد داده شد. این الگوریتم از تجمیع داده‌ی تمام کاربران برای پیش‌بینی درخواست بعدی هر کاربر استفاده می‌کند. موتور پیش‌بینی در پیشکار قرار دارد. در روش پیشنهادی [۵]، چندین مدل مارکوف برای انجام پیش‌بینی ترکیب می‌شوند. جواب بهینه هنگام استفاده از مدل مارکوف مرتبه‌ی دو حاصل شد. در [۶] یک نسخه‌ی بهبودیافته ازنظر حافظه برای PPM ارائه داده شد که زیر درخت پیش‌بینی آن بسته به محبوبیت شیء ریشه، ارتفاع متفاوتی داشت. در این روش، نشست‌های کاربران توسط یک مدل مارکوف نمایش داده می‌شوند. هر صفحه به‌عنوان ریشه می‌تواند استفاده شود و هر صفحه‌ی درخواست شده‌ی بعدی در نشست، در درختی قرار داده می‌شود که ریشه‌ی آن صفحه‌ی قبلی است. هر گره یک شمارنده نیز دارد که تعداد دفعات ملاقات آن گره از مسیر ریشه‌ی مربوطه را نگه می‌دارد.

الگوریتم پیش‌بینی DDG (Double Dependency Graph) که در [۱۰] ارائه شد، شبیه الگوریتم DG [۳] از دنباله‌ی دسترسی کاربران برای ایجاد گراف پیش‌بینی استفاده می‌کند. این پیش‌بینی کننده از قسمتی از سرآیند پاسخ به نام "Content-Type" برای تمیز بین صفحه و شیء تعبیه‌شده استفاده می‌کند. صفحات وب درواقع اشیائی هستند که به‌صورت صریح توسط کاربر درخواست می‌شوند، درحالی‌که اشیاء

تعبیه‌شده آن‌هایی هستند که توسط مرورگر وب به‌صورت خودکار درخواست می‌شوند. "Content-Type" به DDG امکان تمایز بین اشیاء یک صفحه و اشیاء صفحات دیگر را می‌دهد. درنتیجه ساختار وب‌سایت بهتر از DG در نظر گرفته می‌شود. این الگوریتم دو کلاس وابستگی را تعریف می‌کند: وابستگی بین اشیاء یک صفحه و وابستگی بین اشیاء صفحات مختلف. این دو کلاس وابستگی موجب عملکرد بهتر این الگوریتم می‌شود.

الگوریتم‌های پیش‌بینی می‌توانند در دودسته‌ی بلندمدت و کوتاه‌مدت قرار گیرند [۱۱]. در روش بلندمدت از اطلاعات آماری کلی دسترسی‌ها مثل محبوبیت اشیاء برای انتخاب اشیاء جهت پیش‌بینی استفاده می‌شود. در مقابل روش کوتاه‌مدت وجود دارد که با استفاده از دسترسی‌های اخیر کاربر، درخواست آتی کاربر را پیش‌بینی می‌کند.

الگوریتم‌های پیش‌بینی مختلفی نیز ارائه شده‌اند که از وب‌کاوی استفاده می‌کنند. کار [۷] می‌تواند در دسته‌ی کوتاه‌مدت قرار گیرد. این مقاله روشی ارائه می‌دهد که کاربران را بر اساس نرخ تناوب دستیابی به هرکدام از اشیاء، گروه‌بندی می‌کند. بدین ترتیب، ساختار سرویس‌دهنده می‌تواند طبق نیازهای گروه‌های کاربران سازمان یابد. برای کاهش سربار شبکه، الگوریتم به‌جای پیش‌بینی برای هر کاربر، با استفاده از خوشه پیش‌بینی را انجام می‌دهد. هنگامی‌که یک کاربر به سرویس‌دهنده یا پیشکار متصل می‌شود، استراتژی پیش‌واکشی URL[۲]هایی را برای خوشه‌ی دربرگیرنده‌ی کاربر، پیش‌بینی می‌کند. مدل پیشنهادی [۸] از مقایسه‌ی شباهت نشست‌ها برای پیش‌بینی استفاده می‌کند. نشست‌ها با در نظر گرفتن دنباله‌ی درخواست کاربران و زمان سپری‌شده در هر صفحه، مدل و خوشه‌بندی می‌شوند. پیش‌بینی هم با استفاده از این خوشه‌ها انجام می‌شود. مانند روش [۸]، کار پیشنهادی [۱۲] نیز ابتدا نشست‌های کاربران را خوشه‌بندی می‌کند. با استفاده از مراکز خوشه و الگوهای کشف‌شده، درخواست آتی را پیش‌بینی می‌کند.

۳- روش پیشنهادی

روش پیشنهادی دارای دو فاز برون‌خط و برخط است. در فاز برون‌خط، الگوی دستیابی کاربران شناسایی می‌شود. الگوی دستیابی کاربران، اطلاعات خوشه‌های به‌دست‌آمده در این فاز است. سپس، مدلی جهت پیش‌بینی درخواست آتی کاربر، ایجاد می‌شود. در فاز برخط از مدل ایجادشده استفاده می‌شود تا درخواست‌های آتی کاربر پیش‌بینی شوند.

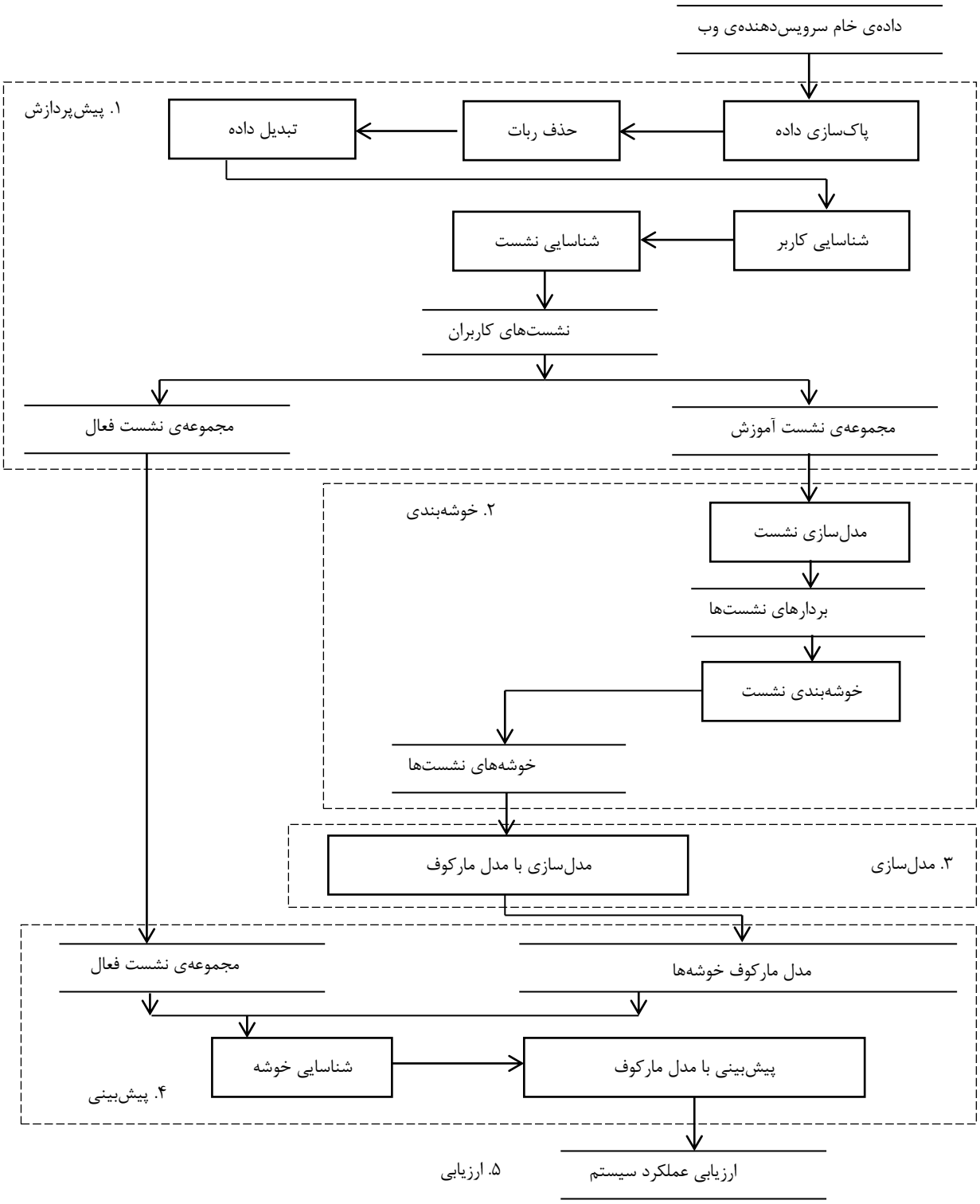
در شکل ۱ مراحل روش پیشنهادی نشان داده شده است. مراحل ۱، ۲ و ۳ فاز برون‌خط روش پیشنهادی هستند و مرحله‌ی ۴ فاز برخط را روش پیشنهادی است. ابتدا داده‌ی ثبت پیش‌پردازش می‌شود و کاربران و نشست‌ها شناسایی می‌شوند. سپس نشست‌های آموزش و نشست‌های فعال انتخاب می‌شوند. نشست فعال نشستی است که در فاز آزمایش استفاده می‌شود و هدف مقاله، پیش‌بینی درخواست‌های آن‌ها است. نشست‌های آموزش خوشه‌بندی می‌شوند و مدل مارکوف هر خوشه ایجاد می‌شود. در فاز برخط، پیش‌بینی درخواست‌های آتی نشست‌های فعال با استفاده از مدل مارکوف و اطلاعات خوشه‌ها انجام می‌شود. درنهایت عملکرد روش پیشنهادی بررسی می‌شود. توضیحات بیش‌تر نحوه‌ی انجام این مراحل در ادامه‌ی این بخش آمده است.

۳-۱- پیش‌پردازش داده

داده‌ی ورودی، داده‌ی ثبت سرویس‌دهنده‌ی وب است که در آن اطلاعات زیادی قرار دارد. اطلاعات این داده‌ی ثبت شامل دسترسی‌های صورت‌گرفته توسط کاربران به اشیاء وب‌سایت است. داده‌ی ثبت دریافتی پیش‌پردازش شده و فقط اطلاعات مفید نگه‌داشته می‌شود. پاک‌سازی داده‌ی ثبت شامل مراحل زیر است:

- حذف درخواست‌های مربوط به ربات‌ها و کاوشگران وب
- حذف رکوردهای دارای کد وضعیت خطا

درخواست‌های ذخیره‌شده ناشی از ربات‌های موتورهای جستجو یا بقیه کاوشگران، نشانگر سلیقه و الگوی پیمایش کاربران نمی‌باشند [۱۳]. به همین دلیل اطلاعات مربوط به آن‌ها از داده‌ی ثبت حذف می‌شوند. برای حذف ربات‌ها از روش معرفی‌شده در [۱۴] استفاده شده است. در این روش، واردهای مربوط به کاربرانی که فایل "robots.txt" را درخواست داده‌اند، کاربرانی که حداقل یک فایل را با روش HEAD درخواست داده‌اند و کاربرانی که هیچ درخواستی برای اشیای تعبیه‌شده در صفحات نداشته باشند، به‌عنوان درخواست ربات در نظر گرفته شده و کنار گذاشته می‌شود. سپس صفحات درخواست شده، در یک جدول در یک پایگاه داده‌ی رابطه‌ای ریخته می‌شود و برای کاهش بار محاسباتی پروسه‌ی کاربردکاوی در وب، به هر یک از این صفحات یک مقدار عددی منحصربه‌فرد به‌عنوان سمبل آن صفحه اختصاص داده می‌شود که به آن تبدیل داده می‌گوییم.



شکل (۱): مراحل روش پیش‌بینی

۳-۱-۱- شناسایی کاربران

تحلیل‌های حوزه‌ی کاربردکاوی وب نیازی به دانشی درباره‌ی هویت کاربران ندارند، ولی تشخیص کاربران مختلف از یکدیگر ضروری است. می‌توان با ترکیب فیلدهای آدرس IP و سایر اطلاعات مانند عامل کاربر (اطلاعات راجع به ماشین کاربر) و URL ارجاع‌کننده، کاربران را شناسایی کرد [۱۵]. در [۱۵]، برای شناسایی کاربران از اطلاعات موجود در داده‌ی ثبت استفاده شده است. بدین ترتیب که کاربرانی که دارای IP، مرورگر و سیستم‌عامل یکسان باشند، به‌عنوان یک کاربر در نظر گرفته می‌شوند. در این تحقیق نیز از روش معرفی‌شده در [۱۵] برای شناسایی کاربران استفاده شده است.

۳-۱-۲- شناسایی نشست‌ها

هنگام کاوش اطلاعات داده‌ی ثبت سرویس‌دهندگان، یکی از مشکلات اصلی شناسایی نشست کاربران است. منظور از شناسایی نشست، چگونگی گروه‌بندی تمام صفحاتی است که یک کاربر به‌صورت متوالی در حین یک پیمایش در وب‌سایت مرور می‌کند. در این زیربخش، از یک مکاشفه‌ی مبتنی بر زمان برای شناسایی نشست کاربران استفاده شده است (مانند [۱۳]). بدین‌صورت که یک حد آستانه برای مدت‌زمان نشست کاربران در نظر گرفته شده است. اگر فاصله‌ی درخواست بعدی کاربر از اولین درخواستش، بیش‌تر از این حد آستانه باشد، به‌منزله‌ی اتمام نشست قبلی و شروع نشست جدید است. مانند بیش‌تر روش‌های قبلی، حداکثر مدت‌زمان نشست نیز ۳۰ دقیقه در نظر گرفته شده است. پس از شناسایی نشست‌ها، مانند [۱۶، ۱۷] نشست‌هایی که در آن‌ها بیش‌تر از یک تعداد صفحه ملاقات شده‌اند، نگه‌داری می‌شوند. این مقدار در این مقاله سه صفحه برای هر نشست در نظر گرفته شده است، چراکه در نشست‌ها به‌طور متوسط سه صفحه ملاقات شده است.

۳-۲- خوشه‌بندی نشست‌ها

در این زیربخش نحوه‌ی خوشه‌بندی نشست‌های کاربران توضیح داده می‌شود. برای خوشه‌بندی، ابتدا از هر نشست، فقط درخواست به صفحات نگه‌داشته می‌شود. چراکه هدف از خوشه‌بندی، گروه‌بندی نشست‌هایی است که علایق مشترک دارند. هر نشست، دنباله‌ای از درخواست‌های سمت مشتری به سرویس‌دهنده‌ی وب است. این دنباله، شامل درخواست به صفحات وب و اشیاء تعبیه‌شده در هر صفحه است. از طرفی درخواست به صفحات نشانگر علاقه‌ی کاربر است. چراکه صفحات به‌صورت صریح توسط کاربر درخواست می‌شوند. ولی اشیاء تعبیه‌شده توسط مرورگر درخواست می‌شوند. بنابراین، جهت گروه‌بندی کاربران دارای علایق مشترک در یک گروه، فقط از درخواست‌های صریح آن‌ها برای ایجاد بردار نشست استفاده می‌شود. برای این کار، هر نشست به یک بردار Nبعدی تبدیل می‌شود. N تعداد صفحات ملاقات‌شده توسط تمام کاربران در وب‌سایت است. مقدار هر بعد برای هر نشست بیانگر میزان علاقه‌ی کاربر به صفحه‌ی مربوطه است.

در تبدیل نشست کاربران به بردار مربوطه، برای افزایش سرعت و کاهش بعد بردار، صفحاتی که در تعداد کمی از نشست‌ها مرور شده‌اند، کنار گذاشته می‌شوند. چراکه این صفحات بیانگر علاقه‌ی شخصی و یا لحظه‌ای کاربر می‌باشد و نمی‌تواند علایق گروهی را نشان دهد [۱۶،۱۸،۱۹،۲۰]. مانند کار [۱۳]، اگر صفحه‌ای در کم‌تر از ۰/۵ درصد نشست‌ها درخواست شده باشد، از بعد بردار نشست‌ها کنار گذاشته می‌شود. برای محاسبه بردار نشست، مدت‌زمان مرور صفحات داخل نشست و تعداد دفعات مرور صفحات در نشست، نشان‌دهنده اهمیت آن صفحات در نظر گرفته شده است [۱۳]. هر چه مدت‌زمان مرور یک صفحه و تعداد دفعات مرور آن بیش‌تر باشد، ارزش آن صفحه بیش‌تر بوده و مقدار متناظر آن صفحه در بردار بیش‌تر می‌شود. برای محاسبه‌ی عامل مدت‌زمان مرور صفحه از رابطه‌ی ۱ استفاده شده که در [۱۳] ارائه شده است.

$$duration(\mathrm{i},\mathrm{j}) = \frac{d(\mathrm{i},\mathrm{j}) / \mathrm{length}(\mathrm{i})}{\max_{i \in \ pages\ in\ j}(\mathrm{d}(\mathrm{i},\mathrm{j}) / \mathrm{length}(\mathrm{i}))} \quad (۱)$$

مقدار $d(\mathrm{i},\mathrm{j})$ مدت‌زمان مرور صفحه‌ی i در نشست j است. مقدار $length(\mathrm{i})$ تعداد بایت صفحه‌ی i است. برای محاسبه تعداد دفعات مرور صفحات از معیار $TF.IDF$ [۲۱] استفاده می‌کنیم. رابطه‌ی مربوط به تعداد دفعات مرور صفحه در یک نشست خاص از رابطه‌ی ۲ حساب می‌شود.

$$TF(\mathrm{i},\mathrm{j}) = \frac{numberofvisit(\mathrm{i})}{\max(\mathrm{numberofvisit(i)})} \quad (۲)$$

در رابطه‌ی فوق، $numberofvisit(\mathrm{i})$ بیانگر تعداد ملاقات صفحه‌ی i در نشست j است. مقدار IDF از رابطه‌ی ۳ حساب می‌شود.

$$IDF(\mathrm{i}) = \log(\frac{N}{N_i}) \quad (۳)$$

N برابر با تعداد کل نشست‌ها است و N_i تعداد نشست‌هایی را نشان می‌دهد که i را ملاقات کرده‌اند. معیار $TF.IDF$ هر بعد یک نشست از رابطه‌ی ۴ حساب می‌شود.

$$TF.IDF(\mathrm{i},\mathrm{j}) = \frac{TF(\mathrm{i},\mathrm{j}) \times \mathrm{IDF(i)}}{\max_{i \in \ pages\ in\ j}(TF(\mathrm{i},\mathrm{j}) \times \mathrm{IDF(i)})} \quad (۴)$$

درنهایت میانگین توافقی اعداد حاصل از دو معیار $TF.IDF$ و $duration$، محاسبه می‌شود و به‌عنوان مقدار هر بعد نشست نگه داری می‌شود. رابطه‌ی ۵، رابطه‌ی محاسبه‌ی علاقه‌ی نشست به هر صفحه را نشان می‌دهد.

$$\mathrm{int}erest(\mathrm{i},\mathrm{j}) = \frac{2 \times TF.IDF(\mathrm{i},\mathrm{j}) \times \mathrm{duration(i,j)}}{TF.IDF(\mathrm{i},\mathrm{j}) + \mathrm{duration(i,j)}} \quad (۵)$$

پس از به دست آوردن بردار هر نشست، از الگوریتم k-means++ کروی جهت خوشه‌بندی نشست‌ها استفاده می‌شود. در این الگوریتم، از k-means کروی [۲۲] برای خوشه‌بندی نشست‌ها استفاده شده است و از روش مقداردهی اولیه‌ی مراکز خوشه‌ها در k-means++ [۲۳] جهت مقداردهی اولیه‌ی مراکز خوشه‌ها استفاده شده است. برای یافتن میزان شباهت بین دو بردار، از شباهت کسینوسی استفاده شده است.

۳-۳- ایجاد مدل مارکوف

پس از اینکه نشست‌ها خوشه‌بندی شدند، یک مدل مارکوف برای هر یک از خوشه‌ها ایجاد می‌شود. مدل ایجادشده DDG است [۱۰]. این مدل، یک گراف وزن‌دار است و در آن به ازای هر شیء (هم صفحه و هم اشیاء دیگر داخل آن) یک گره وجود دارد.

دو کلاس مختلف وابستگی در نظر گرفته شده است: وابستگی بین اشیاء یک صفحه (یال‌های ثانویه) و وابستگی بین اشیاء صفحات مختلف (یال اولیه). گراف، یک گره برای هر شیء‌ای که دسترسی به آن

صورت گرفته، دارد. یک یال از گره A به گره B وجود دارد اگر و فقط اگر در یک نقطه‌ی زمانی، در طول w درخواست پس از دسترسی به A، B نیز درخواست شده باشد. w اندازه‌ی پنجره‌ی پیش‌بینی است. یال، اولیه است اگر A و B اشیاء متعلق به صفحات مختلف باشند. به‌عبارت‌دیگر، B یا یک شیء HTML است، یا کاربر به یک شیء HTML بین A و B دسترسی داشته است. اگر هیچ دسترسی به شیء HTML بین A و B وجود نداشته باشد، یال ثانویه است. یک شمارنده‌ی یال و یک شمارنده‌ی گره نیز وجود دارد. برای اضافه کردن خواص زمانی دسترسی‌ها، یک متغیر نیز در کنار هر یال وجود دارد که شمارنده‌ی گره مبدأ، هنگام آخرین ملاقات گره مقصد را نگه می‌دارد. وزن هر یال از رابطه‌ی ۶ حساب می‌شود.

$$weight(\mathrm{i},\mathrm{j}) = \frac{numberofvisit(\mathrm{i},\mathrm{j}) \times \mathrm{lastvisit}(\mathrm{i},\mathrm{j})}{count(\mathrm{i})^2} \qquad (۶)$$

عبارت $numberofvisit(\mathrm{i},\mathrm{j})$ تعداد ملاقات شیء j پس از i در پنجره w را نگه می‌دارد. $lastvisit(\mathrm{i},\mathrm{j})$ شمارنده‌ی گره i، هنگام آخرین ملاقات گره j را نگه می‌دارد. $count(\mathrm{i})$ نیز تعداد ملاقات گره i است.

۳-۴- پیش‌بینی

در مرحله‌ی پیش‌بینی، ابتدا خوشه‌ی مربوط به نشست فعال شناسایی می‌شود. برای این کار از پنجره‌ی درخواست کاربر استفاده شده است. ابتدا کاربر به تعداد پنجره‌ی موردنظر، صفحه ملاقات می‌کند. سپس نزدیک‌ترین خوشه به نشست کاربر شناسایی می‌شود. برای یافتن نزدیک‌ترین خوشه، از ترکیب الگوریتم KNN [۲۱] با مقدار ۱ برای K و اطلاعات مدل مارکوف هر خوشه، استفاده شده است.

برای یافتن خوشه‌ی موردنظر، لیست صفحات درخواست شده‌ی قبلی نشست را داریم. با استفاده از مدل مارکوف هر خوشه، ابتدا مقدار شمارنده‌ی خوشه‌ای را افزایش می‌دهیم که وزن یال خروجی از صفحات ملاقات‌شده‌ی قبلی کاربر به شیء جدید در آن‌ها بالاتر از حد آستانه است. خوشه‌هایی که شمارنده‌ی آن‌ها بیشینه است، به‌عنوان خوشه‌ی نامزد انتخاب می‌شود. شباهت نشست فعال با مراکز خوشه‌های نامزد محاسبه شده و نزدیک‌ترین خوشه به‌عنوان خوشه‌ی نشست انتخاب می‌شود. سپس با استفاده از DDG ایجادشده برای خوشه‌ی مربوطه، صفحات آتی کاربر پیش‌بینی می‌شوند. روش پیش‌بینی شبیه روش پیشنهادی [۱۰] است و دو تا حد آستانه‌ی متفاوت برای یال‌های اولیه و ثانویه در نظر گرفته می‌شود. تفاوت روش پیشنهادی با [۱۰] در اضافه کردن اطلاعات زمانی مربوط به دسترسی‌ها است.

۴- ارزیابی

برای مقایسه‌ی روش پیشنهادی با الگوریتم DDG از داده‌ی ثبت سرویس‌دهنده‌ی وب ناسا [۲۴] استفاده شده است که از یکم تا سی‌ویکم جولای سال ۱۹۹۵ جمع‌آوری شده است. اطلاعات مربوط به این داده‌ی ثبت در جدول ۱ آمده است. منظور از درخواست‌های صحیح آن‌هایی است که دارای کد وضعیت صحیح هستند و اشیاء درخواستی آن‌ها، قابلیت ذخیره در حافظه‌ی نهان مرورگر را دارند. از پانزده روز اولیه برای آموزش سیستم و از بقیه‌ی داده‌ی ثبت برای آزمایش استفاده شده است.

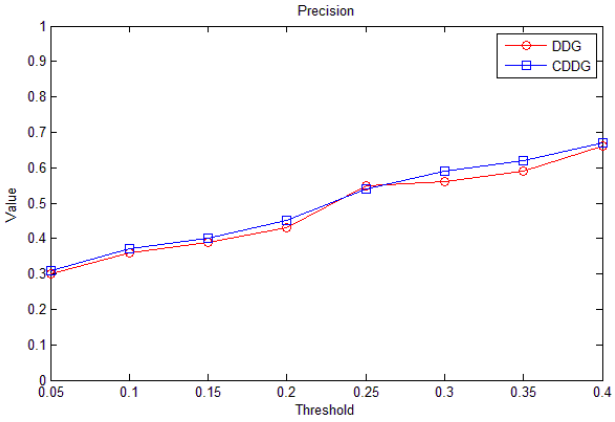
جدول (۱): اطلاعات مربوط به داده‌ی ثبت ناسا

	NASA
تعداد کل درخواست‌ها	۱۸۹۱۷۰۹
تعداد درخواست صحیح	۱۷۰۱۵۳۴
تعداد درخواست به صفحات	۵۶۱۴۰۷

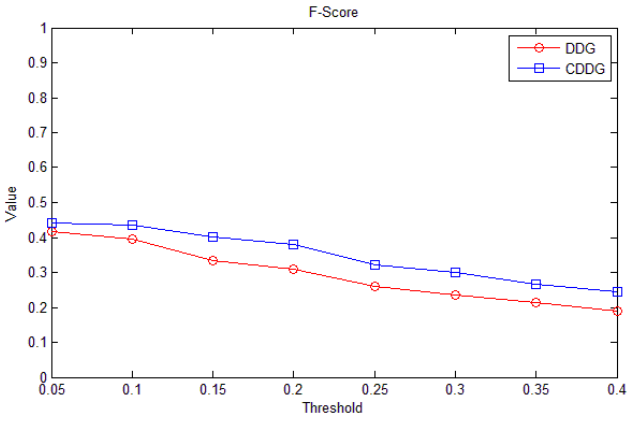
برای ایجاد مدل مارکوف برای هر خوشه مانند پیاده‌سازی [۱۰]، اندازه‌ی پنجره‌ی پیش‌بینی ۱۲ در نظر گرفته شده است. حد آستانه‌ی یال‌های ثانویه عددی ثابت و برابر با ۰/۳ در نظر گرفته شده است. حد آستانه‌ی یال‌های اولیه متغیر در نظر گرفته شده است و این عدد از ۰/۰۵ تا ۰/۴ با گام ۰/۰۵ تغییر داده شده است.

برای ارزیابی روش پیشنهادی از معیارهای دقت، یادآوری و F-score استفاده شده است. دقت تعداد پیش‌بینی درست به‌کل پیش‌بینی‌ها است. یادآوری نسبت اشیاء درخواستی کاربر است که قبلاً پیش‌بینی شده‌اند. معیار F-score میانگین توافقی دقت و یادآوری است. نتایج مربوط به هر یک از این معیارها در شکل‌های ۲، ۳ و ۴ آمده است. روش پیشنهادی با نام CDDG در شکل‌ها نشان داده شده است.

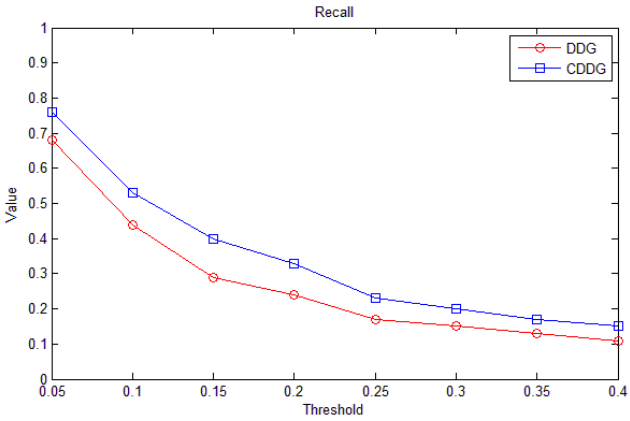
همان‌طور که در شکل‌های ۱ و ۲ مشاهده می‌شود، معیارهای دقت و یادآوری رفتاری متفاوت دارند. با افزایش حد آستانه دقت نیز بالاتر می‌رود. ولی این امر موجب کاهش یادآوری می‌شود. به دلیل اینکه این دو معیار برای ما دارای اهمیت یکسانی هستند، از معیار F-score نیز برای ارزیابی روش پیشنهادی استفاده شده است. شکل ۲ نمودار مربوط به معیار دقت را نشان می‌دهد. روش پیشنهادی در اکثر موارد (به‌غیراز آستانه ۰/۲۵) ۱ تا ۳ درصد بهتر از DDG عمل کرده است. شکل ۳ نمودار مربوط به معیار یادآوری را نشان می‌دهد. روش پیشنهادی در تمامی موارد عملکرد بهتری نسبت به DDG داشته است و توانسته بین ۴ تا ۱۱ درصد بهتر از DDG عمل کند. شکل ۴ نمودار مربوط به معیار F-score را نشان می‌دهد. روش پیشنهادی ۳ تا ۱۳ درصد بهبود نسبت به DDG داشته است. در کل با توجه به عملکرد بهتر روش پیشنهادی در هر دو معیار دقت و یادآوری، می‌توان نتیجه گرفت که روش ارائه‌شده می‌تواند عملکرد بهتری نسبت به DDG در پیش‌واکشی داشته باشد.



شکل (۲): نمودار مربوط به معیار دقت روش پیشنهادی و DDG



شکل (۳): نمودار مربوط به معیار یادآوری روش پیشنهادی و DDG



شکل (۴): نمودار مربوط به معیارF-score روش پیشنهادی و DDG

۵- نتیجه‌گیری

پیش‌واکشی وب یکی از روش‌های مؤثر جهت کاهش تأخیر مشاهده‌شده توسط کاربر است. پیش‌بینی موفق درخواست آتی کاربر موجب بهبود عملکرد پیش‌واکشی می‌شود. در این مقاله روشی جهت پیش‌واکشی ارائه شد که از ترکیب مدل مارکوف و خوشه‌بندی نشست‌ها استفاده می‌کند. همچنین از خصوصیات زمانی دسترسی‌ها در پیش‌بینی استفاده شد تا دقت پیش‌بینی‌ها افزایش یابد. نتایج نشان می‌دهند که روش ارائه‌شده، رویکرد مناسبی جهت پیش‌بینی درخواست آتی کاربر است. همچنین به دلیل پیش‌بینی صفحات و اشیاء، روش پیشنهادی قابل‌استفاده در پیش‌واکشی است. برای کارهای آتی، از روشی مبتنی بر محتوای صفحات برای خوشه‌بندی نشست‌ها استفاده خواهد شد. چراکه استفاده از محتوای صفحات و کاوش آن می‌تواند موجب شناسایی شباهت بین صفحات مختلف با یکدیگر شود. از این دانش می‌توان برای توصیه‌ی صفحات جدیدی که به وب‌سایت اضافه شده‌اند، استفاده کرد.

مراجع

[1] A. Sidiropoulos, G. Pallis, D. Katsaros, K. Stamos, A. Vakali and Y. Manolopoulos, "Prefetching in content distribution networks via web communities identification and outsourcing," World Wide Web, vol. 11, no. 1, pp. 39-70, 2008.

[2] Y. Jiang, M-Y. Wu and W. Shu, "Web prefetching: costs, benefits and performance," in Proceedings of the 7th international workshop on web content caching and distribution, Boulder, Colorado, 2002.

[3] V. N. Padmanabhan and J. C. Mogul, "Using predictive prefetching to improve World Wide Web latency," SIGCOMM ComputCommun Rev, vol. 26, no. 3, pp. 22-36, 1996.

[4] J. Domenech, B. de la Ossa, J. Sahuquillo, J. A. Gil and A. Pont, "A taxonomy of web prediction algorithms," Expert Systems with Applications, vol. 39, no. 9, pp. 8496-8502, 2012.

[5] L. Fan, P. Cao, W. Lin and Q. Jacobson, "Web prefetching between low-bandwidth clients and proxies: potential and performance," SIGMETRICS Perform Eval Rev, vol. 27, no. 1, pp.178-187, 1999.

[6] C. Xin and Z. Xiaodong, "Popularity-based PPM: an effective Web prefetching technique for high accuracy and low storage," in Proceedings of the International Conference on Parallel Processing, pp. 296-304, 2002.

[7] S. K. Rangarajan, V. V. Phoha, K. S. Balagani, R. R. Selmic and S. S. Iyengar, "Adaptive neural network clustering of web users," Computer, vol. 37, no. 4, pp. 34-40, 2004.

[8] Ş. Gündüz and M. T. Özsu, "A web page prediction model based on click-stream tree representation of user behavior," in Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 535-540, 2003.

[9] X. Li, B. Liu and P. Yu, "Time sensitive ranking with application to publication search," Link Mining: Models, Algorithms, and Applications, Springer, New York, pp. 187-209, 2010.

[10] J. Domenech, J. A. Gil, J. Sahuquillo and A. Pont, "Using current web page structure to improve prefetching performance," Computer Networks, vol. 54, no. 9, pp. 1404-1417, 2010.

[11] G. Pallis, A. Vakali and J. Pokorny, "A clustering-based prefetching scheme on a Web cache environment," Computers & Electrical Engineering, vol. 34, no. 4, pp. 309-323, 2008.

[12] M. Wan, L. Li, J. Xiao, Y. Yang, C. Wang and X. Guo, "CAS based clustering algorithm for Web users," Nonlinear Dynamics, vol. 61, no. 3, pp. 347-361, 2010.

[13] H. Liu and V. Kešelj, "Combined mining of Web server logs and web contents for classifying user navigation patterns and predicting users' future requests," Data & Knowledge Engineering, vol. 61, no. 2, pp. 304-330, 2007.

[14] J.X. Yu, O. Yuming, C. Zhang and S. Zhang, "Identifying interesting visitors through Web log classification," Intelligent Systems, vol. 20, no. 3, pp. 55-59, 2005.

[15] R. Cooley, B. Mobasher and J. Srivastava, "Data preparation for mining world wide web browsing patterns," Knowledge and information systems, vol. 1, no. 1, pp. 5-32, 1999.

[16] T. W. Yan, M. Jacobsen, H. Garcia-Molina and U. Dayal, "From user access patterns to dynamic hypertext linking," Computer Networks and ISDN Systems, vol. 28, no. 7, pp. 1007-1014, 1996.

[17] A. Banerjee and J. Ghosh, "Clickstream clustering using weighted longest common subsequences," in Proceedings of the

web mining workshop at the 1st SIAM conference on data mining, vol. 143, pp. 144-152, 2001.
[18] B. Mobasher, "Webpersonalizer: a server-side recommender system based on web usage mining," in Proceedings of the 9th Workshop on Information Technologies and Systems (WITS'99), Charlotte, NC, 1999.
[19] Y. Xie and V. V. Phoha, "Web user clustering from access log using belief function," in Proceedings of the 1st international conference on Knowledge capture, ACM, pp. 202-208, 2001.
[20] P. Kumar, P. R. Krishna, R. S. Bapi and S. K. De, "Rough clustering of sequential data," Data & Knowledge Engineering, vol. 63, no. 2, pp. 183-199, 2007.
[21] B. Liu, *Web data mining: exploring hyperlinks, contents, and usage data*, Springer, 2007.
[22] Z. Shi, "Efficient online spherical k-means clustering," IEEE International Joint Conference, pp. 3180-3185, 2005.
[23] D. Arthur and S. Vassilvitskii, "K-means++: the advantages of careful seeding," in Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, New Orleans, Louisiana, 2007.
[24] NASA webserver log, available at http://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html.

زیرنویس‌ها

[1] Log
[2] Proxy
[3] Uniform Resource Locator