توزیع مؤثر اسناد برای ایجاد توازن بار بین سرورها با استفاده از شمارش رخداد کلمات در سابقه پرس‌وجوها

سیده ریحانه تراب جهرمی و سجاد ظریف‌زاده

چکیده: هدف اصلی موتورهای جستجو، یافتن مرتبط‌ترین نتایج نسبت به پرس‌وجوی کاربر در سریع‌ترین زمان ممکن است. صفحات خزش‌شده توسط موتور جستجو بین سرورهای متعددی توزیع می‌شوند تا در هنگام جستجو بتوان از قدرت بازیابی و پردازش موازی آنها برای تولید سریع‌تر پاسخ استفاده نمود. با توجه به تعداد بسیار زیاد صفحات وب، موتورهای جستجو سیاست‌های مختلفی را برای توزیع مناسب اسناد بین سرورها انتخاب می‌کنند. در این مقاله، روش جدیدی برای توزیع اسناد پیشنهاد می‌شود که هدف آن ایجاد توازن بار کاری بین سرورها برای کاهش زمان پاسخ‌گویی موتور جستجو می‌باشد. ایده اصلی، استفاده از پرس‌وجوهای قبلی کاربران است بدین ترتیب که به هر کلمه از کلمات موجود در سابقه پرس‌وجو بر حسب تعداد رخداد روزانه آن، وزنی نسبت داده می‌شود. سپس هر سند با توجه به مجموع وزن کلمات داخل آن، وزن‌دهی می‌شود که این وزن ارتباط مستقیمی با احتمال انتخاب آن سند به عنوان پاسخ یک پرس‌وجو دارد. در نهایت، اسناد به نحوی بین سرورها توزیع می‌شوند که وزن اسناد داخل هر یک از سرورها برابر باشد. نتایج ارزیابی با استفاده از داده واقعی نشان می‌دهند که روش پیشنهادی قادر است توازن بار سرورها را مخصوصاً در زمان اوج ورود پرس‌وجوها بیش از ۲۰٪ نسبت به روش‌های گذشته بهبود بخشد.

***کلیدواژه:* توازن بار، توزیع سند، سابقه پرس‌وجو، موتور جستجو.**

۱- مقدمه

با گسترش اینترنت و افزایش چشم‌گیر استفاده کاربران از وب، موتورهای جستجو اهمیت قابل ملاحظه‌ای پیدا کرده‌اند [۱]. هر موتور جستجو شامل حجم انبوهی از اسناد یا صفحات جمع‌آوری شده از وب، میلیون‌ها کاربر و تعداد زیادی سرور برای ذخیره‌سازی توزیع‌شده اسناد و پردازش موازی پرس‌وجوهای کاربران است. انتظار کاربران از موتور جستجو دریافت مرتبط‌ترین نتایج در زمانی کوتاه (معمولاً کسری از ثانیه) می‌باشد. لیکن به دلیل حجم زیاد اسناد و پرس‌وجوها، چالش‌های متعددی برای تولید پاسخ در این زمان اندک وجود دارد. برای مثال، فرض کنید که بیشتر اسنادی که در پاسخ به پرس‌وجوهای کاربران برگردانده می‌شوند، در تعداد محدودی سرور ذخیره شده باشند و اسنادی که کمتر مورد درخواست هستند در بقیه سرورها ذخیره گردند. در این صورت با بالارفتن تعداد پرس‌وجوها، بار پردازشی برخی سرورها (برای جستجو و بازیابی اسناد از نمایه و اجرای الگوریتم رتبه‌بندی) به شدت بالا رفته و اصطلاحاً تشکیل گلوگاه می‌دهند، یعنی زمان پاسخ کل موتور وابسته به عملکرد تعداد محدودی سرور می‌شود. در این حالت، زمان دسترسی به اسناد به مرور افزایش می‌یابد و در نتیجه، موتور عملکرد نامطلوبی پیدا می‌کند [۲]. از آنجا که موتور جستجو باید در هر زمان، قابلیت پاسخ‌گویی سریع به حجم عظیمی از پرس‌وجوهای کاربران را داشته باشد، به روشی برای توزیع اسناد بین سرورها نیاز است که بتواند بین فضای ذخیره‌سازی مورد استفاده در سرورها از یک سو و میزان دسترسی و بار پردازشی آنها از سوی دیگر توازن ایجاد نماید [۳] و [۴]. تحقیقات گذشته نشان می‌دهند که استفاده از روش‌های مناسب توزیع اسناد که بر توازن بار بین سرورها تمرکز دارند، منجر به کاهش قابل ملاحظه در زمان پاسخگویی به پرس‌وجوهای کاربران می‌شود [۲].

تا کنون روش‌های مختلفی برای توزیع اسناد در موتورهای جستجو ارائه شده‌اند که در یک دسته‌بندی کلی می‌توان آنها را به دو رویکرد مبتنی بر سند و مبتنی بر کلمه تقسیم نمود [۱] و [۵]. در رویکرد اول، اسناد بین سرورها توزیع می‌شوند به این معنی که هر سند دقیقاً در یک سرور قرار می‌گیرد، البته ممکن است کپی‌های متعددی از هر سند نیز داشته باشیم. در رویکرد دوم، نمایه کلمات بین سرورها توزیع می‌گردد. در این حالت ممکن است کلمات داخل هر سند در سرورهای مختلفی قرار گیرند. در مورد مزایا و معایب هر کدام از این رویکردها در ادامه مقاله صحبت خواهیم کرد. امروزه بسته به اهدافی که سیستم‌های بازیابی اطلاعات دنبال می‌کنند، مانند تولید سریع پاسخ، دقت بالای نتایج و استفاده بهینه از منابع، از یکی از این دو رویکرد و یا ترکیب آنها استفاده می‌شود. با رشد چشم‌گیر صفحات وب و همین طور افزایش تعداد پرس‌وجوها در سال‌های گذشته، یافتن روش توزیعی که بتواند توازن بیشتری بین بار پردازشی سرورها ایجاد کند، هنوز هم مورد علاقه محققان حوزه موتور جستجو می‌باشد.

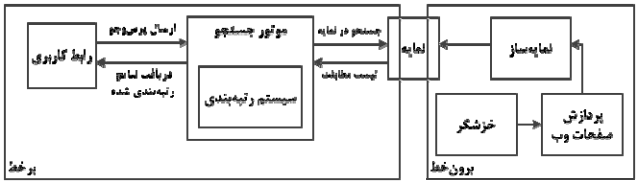
یکی از مشکلات رایج در بسیاری از روش‌های معرفی‌شده گذشته برای توزیع اسناد، عدم تفکیک بین اسناد مهم از مابقی اسناد می‌باشد. منظور از سند مهم سندی است که در پاسخ به بسیاری از پرس‌وجوهای ارسالی به موتور بازگردانده می‌شود و در نتیجه، دائماً مورد جستجو و پردازش قرار می‌گیرد. اطلاعات مربوط به پرس‌وجوهای قبلی کاربران از جمله اطلاعاتی است که در چند سال اخیر برای بهبود تصمیم‌گیری‌های مختلف در موتور جستجو مورد توجه قرار گرفته است. با استفاده از این اطلاعات می‌توان برآوردی از میزان دسترسی به اسناد مختلف در آینده برای توزیع بهتر آنها بین سرورها به دست آورد.

در این مقاله، یک روش جدید مبتنی بر سند برای توزیع اسناد معرفی می‌شود که ایده اصلی آن، استفاده از کلمات موجود در سابقه پرس‌وجوهای کاربران به منظور ایجاد توازن بار بین سرورها می‌باشد. از سابقه کلمات پرس‌وجوهای قبلی برای پیش‌بینی تعداد دسترسی احتمالی به هر سند در پرس‌وجوهای آینده استفاده می‌شود. به این صورت که به هر کلمه، بر اساس آمار استفاده روزانه از آن کلمه در گذشته، وزنی نسبت داده می‌شود.

این مقاله در تاریخ ۵ دی ماه ۱۳۹۷ دریافت و در تاریخ ۶ مرداد ماه ۱۳۹۸ بازنگری شد.

سیده ریحانه تراب جهرمی، گروه مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران، (email: reyhaneh.torab@stu.yazd.ac.ir).

سجاد ظریف‌زاده (نویسنده مسئول)، گروه مهندسی کامپیوتر، دانشگاه یزد، یزد، ایران، (email: szarifzadeh@yazd.ac.ir).



شکل ۱: عملکرد موتور جستجو در دو فاز برخط و برون‌خط.

سپس به اسناد، با توجه به وزن کلمات موجود در متن آنها وزنی داده می‌شود. در نهایت، اسناد با توجه به وزن تخصیص داده شده، به نحوی مابین سرورها توزیع می‌گردند که مجموع وزن اسناد داخل هر یک از سرورها تقریباً برابر شود. نتایج ارزیابی بر روی داده واقعی یک موتور جستجو نشان می‌دهد که روش پیشنهادی، توازن بار بین سرورها را نسبت به روش‌های گذشته به میزان قابل توجهی (بیش از ۲۰٪) بهبود می‌دهد.

ساختار مقاله در ادامه به شرح زیر است: در بخش ۲، کارهای قبلی انجام‌گرفته در این زمینه بیان می‌شود. بخش ۳ به معرفی روش پیشنهادی برای توزیع اسناد می‌پردازد. نتایج ارزیابی در بخش ۴ آورده شده‌اند و نهایتاً بخش ۵ به نتیجه‌گیری اختصاص دارد.

۲- تحقیقات گذشته

در این بخش ابتدا مروری خلاصه بر نحوه عملکرد موتور جستجو خواهیم داشت و سپس کارهای گذشته در زمینه توزیع اسناد را بررسی خواهیم کرد. عملکرد کلی موتور جستجو در شکل ۱ نشان داده شده است. در موتور جستجو دو فاز اصلی وجود دارد [۳] و [۶]. در فاز اول که فاز برون‌خط نامیده می‌شود، تمام محتوای وب توسط خزشگر، خزش می‌گردد و پایگاه داده‌ای برای موتور جستجو ساخته می‌شود که نمایه نام دارد. ساختار نمایه امکان جستجوی سریع در حجم زیادی از اسناد را فراهم می‌آورد. در فاز دیگر که فاز برخط نامیده می‌شود، عملیات جستجو برای بازیابی اسناد مرتبط با پرس‌وجوی کاربر از نمایه صورت می‌گیرد. همچنین در این فاز با اجرای الگوریتم رتبه‌بندی، اسناد مرتبط که در قالب لیست مطابقت قرار گرفته‌اند با توجه به امتیازشان مرتب می‌شوند. با توجه به حجم زیاد اسناد وب و تعداد زیاد پرس‌وجوهای کاربران، ساختار نمایه روی مجموعه بزرگی از سرورها توزیع می‌گردد و فاز برخط (شامل جستجوی نمایه و رتبه‌بندی) به صورت موازی انجام می‌گیرد.

در اغلب موتورهای جستجوی کنونی، ساختار نمایه در حافظه پویا (RAM) قرار دارد [۷]. بنابراین بخش عمده زمان پاسخ یک پرس‌وجو مربوط به زمان بازیابی اسناد مرتبط از نمایه و رتبه‌بندی آنها می‌باشد. به بیان دیگر، هرچه تعداد اسناد مرتبط با یک پرس‌وجو در یک سرور بیشتر باشد، رسیدگی به پرس‌وجو در داخل آن سرور بیشتر طول خواهد کشید. اگر توزیع اسناد مابین سرورها به صورت نامتوازن باشد به گونه‌ای که بیشتر اسناد مرتبط با پرس‌وجوها در تعداد محدودی سرور قرار داشته باشند، آن گاه این سرورها تبدیل به گلوگاه موتور شده و قادر به تولید پاسخ در زمان قابل قبول نیستند [۸].

تاکنون الگوریتم‌های مختلفی برای توزیع مناسب اسناد بین سرورها معرفی شده‌اند که این الگوریتم‌ها را می‌توان به طور کلی به دو رویکرد مبتنی بر سند و مبتنی بر کلمه تقسیم نمود.

۲-۱ توزیع مبتنی بر سند

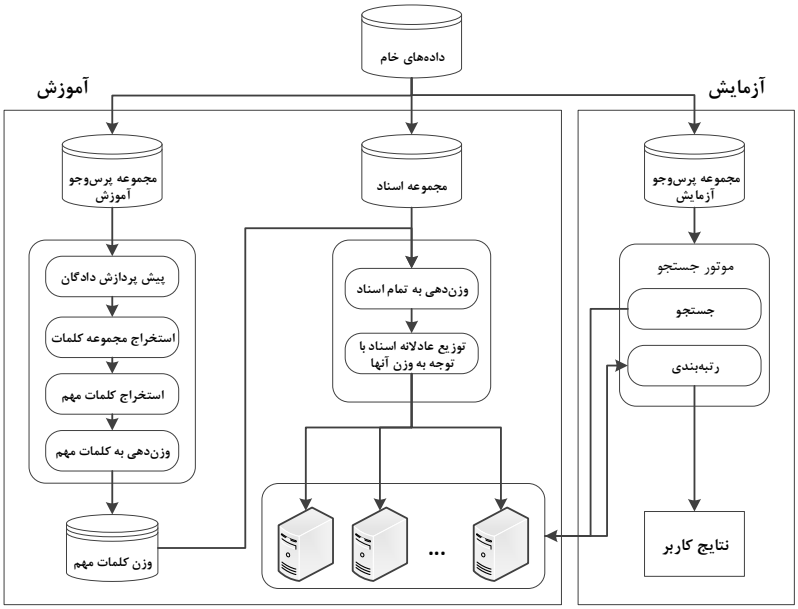
در این رویکرد، ابتدا مجموعه اسناد به زیرمجموعه‌هایی مجزا تقسیم می‌شوند و سپس این زیرمجموعه‌ها در قالب نمایه بین سرورها توزیع می‌گردند [۹]. همچنین یک مدیر مرکزی وجود دارد که پرس‌وجوها را به تمام سرورها ارسال می‌کند. هر سرور پرس‌وجوی دریافتی را در نمایه خود پردازش کرده و مجموعه‌ای از اسناد را در پاسخ به سرور مرکزی ارسال می‌کند. سرور مرکزی پاسخ‌های جمع‌آوری شده از همه سرورها را ترکیب و مرتب می‌کند و تعدادی از اسناد با ارزش‌تر را به عنوان پاسخ به کاربر برمی‌گرداند.

ویژگی مثبت این رویکرد، سادگی و مقیاس‌پذیری آن (برای افزایش اسناد و سرورها) می‌باشد. همچنین به دلیل ارسال هر پرس‌وجو به تمام سرورها، توازن بار نسبی بین سرورها برقرار می‌شود که این خود منجر به افزایش بهره‌وری منابع می‌گردد [۸]. چالش اصلی این رویکرد، تقسیم متوازن اسناد به زیرمجموعه‌ها است به نحوی که اسناد پرتقاضا در یک زیرمجموعه قرار نگیرند. نقطه ضعف دیگر این رویکرد، هزینه بالای آن از بابت تعداد دسترسی زیاد به حافظه به ازای هر پرس‌وجو است، چون ساختار نمایه در هر یک از سرورها باید جستجو شود، صرف‌نظر از این که آیا آن سرور حاوی اسناد مرتبط با پرس‌وجو هست یا خیر. در [۲] روشی مبتنی بر همین رویکرد ارائه می‌شود که در آن، اسناد به نحوی مابین سرورها توزیع می‌شوند که هر دو عامل فضای ذخیره‌سازی و بار کاری سرورها متوازن گردد. ابتدا اسناد بر حسب اندازه‌شان به تعدادی مجموعه با اندازه ثابت تقسیم می‌شوند و سپس این مجموعه‌ها بر حسب بار بین سرورها توزیع می‌گردند. برای تخمین بار هر سند نیز از مجموع احتمال رخداد کلمات داخل آن سند در سابقه پرس‌وجوها استفاده می‌شود.

برای بهبود این رویکرد، توزیع نظیر به نظیر پیشنهاد شد که در آن، هر کاربر به صورت مستقل مجموعه‌ای از اسناد مورد نیاز خود را در یک زیرمجموعه جمع‌آوری می‌کند [۱۰] تا [۱۲]. در این حالت نیز مجموعه اسنادی که توسط یک کاربر جمع‌آوری می‌شود احتمالاً با مجموعه اسناد کاربران دیگر همپوشانی دارد و بنابراین حتی در چنین سیستم پیچیده‌ای نیز ممکن است تقسیم اسناد به صورت نامتوازن انجام گیرد.

در نهایت، روش توزیع موضوعی ارائه گردید که همراه با روشی جدید از جستجو با نام جستجوی انتخابی کار می‌کند [۱۱] و [۱۳]. در روش جستجوی انتخابی، مجموعه اسناد بر اساس شباهت موضوعی یا محتوایی (بر اساس متن سند یا سابقه پرس‌وجوهای کاربران) به دسته‌های مختلف تقسیم‌بندی می‌شود و اسناد هر دسته در مجموعه مشخصی از سرورها قرار می‌گیرند. سپس برای جستجوی یک پرس‌وجو، تنها سرورهایی که حاوی اسناد مرتبط با پرس‌وجو هستند جستجو می‌گردند. در این روش، فرض می‌شود که اسنادی که در پاسخ به یک پرس‌وجو برگردانده می‌شوند، احتمالاً از نظر موضوعی مشابه هستند و در یک دسته قرار می‌گیرند. مزیت این روش کاهش هزینه جستجو و زمان تولید پاسخ است [۱۳]. از معایب این روش، پیچیدگی و عدم دقت آن در تشخیص موضوع اصلی سند و پرس‌وجو و دسته‌بندی اسناد مشابه و همین طور عدم وجود توازن بار کاری بین سرورهای موضوعی است. البته در تحقیقات بعدی تلاش گردید تا با توجه به سابقه پرس‌وجوی کاربران، بار تا حدی متوازن‌تر بین سرورهای موضوعی توزیع شود [۱۴].

اخیراً نیز دو مقاله معرفی شده‌اند که بر کاهش هزینه جستجو (یعنی تعداد اسناد مورد جستجو یا مجموع عملیات انجام‌شده برای یافتن نتایج پرس‌وجو) به جای توازن بار بین سرورها تمرکز دارند [۱۵] و [۱۶]. یعنی زمان جستجوی نمایه را به عنوان بخش اصلی زمان پاسخ فرض می‌کنند که البته اگر در نظر بگیریم که همه اسناد و نمایه‌ها در حافظه RAM قرار دارند، این فرض چندان دقیق نیست (در این حالت، زمان مربوط به اجرای الگوریتم رتبه‌بندی قابل توجه خواهد بود). این روش‌ها عمدتاً از توزیع موضوعی و اطلاعات موجود در سابقه پرس‌وجو استفاده می‌کنند تا تعداد



شکل ۲: نمای کلی روش پیشنهادی.

کمی از نمایه‌های موضوعی جستجو شوند. در نتیجه، نرخ خروجی و زمان پاسخ بهبود می‌یابد ولی توازن بار بین سرورها بسیار بدتر می‌شود که این موضوع نحوه کارکرد موتور جستجو را در زمان اوج ترافیک مختل می‌کند.

مقالات دیگری نیز معرفی شده‌اند که از احتمالات مربوط به توزیع کلمات و اسناد استفاده می‌کنند. در [۱۶] نشان داده می‌شود که توزیع تصادفی در رویکرد مبتنی بر سند منجر به نتایج مناسبی می‌شود ولی کارکرد قابل قبولی در توزیع موضوعی ندارد. این مقاله یک رویکرد ترکیبی با نام توزیع کلمات کوچک معرفی می‌کند و نشان می‌دهد کارایی آن از روش تصادفی بهتر است. البته محاسبات مربوط به این روش صرفاً برای وقتی است که اطلاعات نمایه و اسناد در حافظه دیسک (به جای حافظه پویا) ذخیره شده باشند.

۲-۲ توزیع مبتنی بر کلمه

رویکرد توزیع مبتنی بر کلمه برای حل مشکل تعداد دسترسی به حافظه، نمایه را بر اساس کلمات بین سرورها تقسیم می‌کند [۱۷] و [۱۸] و بنابراین هر سرور، نمایه کلمات خاصی را در خود دارد. برای پاسخگویی به پرس‌وجوها، مدیر مرکزی پرس‌وجو را دریافت می‌کند، از سرورهای مربوط لیست اسناد حاوی هر یک از کلمات پرس‌وجو را درخواست می‌کند و در نهایت، تمام لیست‌های به دست آمده را به صورت مرکزی پردازش و ترکیب می‌کند. بنابراین یک سرور تنها در صورتی در تولید پاسخ مشارکت می‌کند که نمایه یک یا چند کلمه از کلمات داخل پرس‌وجو را داشته باشد. به همین دلیل، تعداد دسترسی‌ها به حافظه در مقایسه با روش توزیع مبتنی بر سند کمتر است که این خود می‌تواند سرعت پاسخ‌گویی به پرس‌وجوها را بهبود دهد [۸]، [۱۷] و [۱۸]. اما مدیر مرکزی به دلیل انجام عمده پردازش‌ها گلوگاه محسوب می‌شود. به علاوه، به این دلیل که بخش عمده‌ای از پردازش مجموعه پرس‌وجوها با استفاده از تعداد کمی از ساختارهای نمایه (یعنی نمایه‌های مربوط به کلمات پراستفاده در پرس‌وجوها) صورت می‌گیرد، سرورهای فاقد این نمایه‌ها دچار گرسنگی می‌شوند. در نتیجه از برخی سرورها بسیار کم و از برخی دیگر بسیار زیاد استفاده می‌شود، مگر آن که مجدداً توزیع مناسبی برای تقسیم کلمات صورت گیرد.

برای رفع معایب گفته‌شده، روش خط لوله پیشنهاد شد که هدف آن حذف گلوگاه در مدیر مرکزی است [۱] و [۱۹]. در این روش، پردازش به صورت کلمه به کلمه انجام می‌گیرد، یعنی به جای فرستادن لیست اسناد حاوی هر یک از کلمات پرس‌وجو به مدیر مرکزی، پرس‌وجو به صورت جزئی و تک‌کلمه در هر سرور پردازش می‌شود. سپس باقیمانده پرس‌وجو و نتایج پردازش سرور فعلی، بین سرورهای بعدی دست به دست می‌شود. این روند تا زمانی که پردازش تمام کلمات پرس‌وجو تمام نشده، ادامه پیدا می‌کند. اگرچه این روش مشکل گلوگاه را حل می‌کند اما چند عیب اساسی دارد، از جمله این که نرخ خروجی پایین‌تر و زمان پاسخ بالاتری به نسبت روش توزیع سند دارد [۱]. همچنین همانند رویکرد اصلی توزیع مبتنی بر کلمه، بین بار پردازشی سرورها توازن وجود ندارد.

موفات و همکاران برای ایجاد توازن بار در روش خط لوله، روشی دیگر را پیشنهاد دادند [۸]. آنها برای توزیع بهینه نمایه کلمات و در نتیجه ایجاد توازن بین بار کاری سرورها، میزان بار کاری را به ازای هر کلمه به دست آوردند و به جای توزیع تصادفی کلمات بین سرورها، از بار آنها برای توزیع استفاده نمودند. از این منظر، روش پیشنهادی ما مشابه با مقاله فوق می‌باشد، یعنی به ازای هر کلمه و با استفاده از سابقه پرس‌وجوهای گذشته، وزنی در نظر می‌گیریم که معرف بار پردازشی آن کلمه است. لیکن بر خلاف روش موفات، روش ما مبتنی بر سند می‌باشد و برای محاسبه وزن هر کلمه از فاکتورهای بیشتری (در کنار میانگین تعداد تکرار یک کلمه) استفاده می‌کنیم که میزان پردازش مورد نیاز آن کلمه را دقیق‌تر نشان می‌دهد.

روش‌های دیگری نیز معرفی شده‌اند که با ترکیب دو رویکرد فوق سعی دارند از مزایای هر دو بهره گیرند. در [۲۰]، ابتدا اسناد و سرورها به گروه‌هایی تقسیم می‌شوند و سپس اسناد بین این گروه‌ها به روش مبتنی بر کلمه توزیع می‌گردند. به طور کلی، روش‌های توزیع مبتنی بر کلمه برای مجموعه داده‌های کوچک عملی هستند [۲۰]. با توجه به سادگی و مقیاس‌پذیری رویکرد توزیع مبتنی بر سند، معمولاً از این رویکرد در موتورهای جستجو استفاده می‌شود. اگرچه برای مشکل اصلی آن یعنی عدم توازن بار بین سرورها، باید حتماً راهکار مناسبی اندیشیده شود.

۳- روش پیشنهادی

شکل ۲ نمای کلی روش پیشنهادی را نشان می‌دهد که بر پایه رویکرد

جدول ۱: مشخصات پرس‌وجوهای موجود در دادگان.

مدت	تعداد پرس‌وجوها	تعداد پرس‌وجوها بعد از فیلتر
پنج روز	۴۸۷۳۷۱	۳۶۲۳۸۲

جدول ۲: مشخصات اسناد موجود در دادگان.

حجم (GB)	تعداد	میانگین طول متن اسناد (کاراکتر)
۳٫۶۹	۱۵۸۲۳۳۱	۱۴۵۴

توزیع مبتنی بر سند کار کرده و خود به دو فاز آموزش و آزمایش تقسیم می‌شود. در فاز آموزش، ما از مجموعه پرس‌وجوهای قبلی کاربران (با عنوان دادگان آموزش) استفاده می‌کنیم و پس از پردازش‌های اولیه، کلمات مهم را از آن استخراج می‌نماییم. سپس به هر کدام از کلمات مهم و در ادامه به هر سند وزنی می‌دهیم به طوری که هرچه وزن یک سند بیشتر باشد، احتمال بازگرداندن آن به عنوان نتیجه جستجو نیز بیشتر خواهد بود. در نهایت، اسناد طوری بین سرورها توزیع می‌شوند که مجموع وزن اسناد داخل همه سرورها تقریباً برابر شود. در فاز آزمایش، پرس‌وجوهای موجود در دادگان آموزش به موتور ارسال می‌شوند تا با توجه به نحوه توزیع اسناد در فاز آموزش، بررسی شود که بار پردازشی سرورها تا چه حد متوازن است. بدیهی است در این مرحله تغییر یا جابه‌جایی در محتویات سرورها نخواهیم داشت.

۳-۱ مدل داده

دادگان مورد استفاده در این پژوهش، پرس‌وجوهای مربوط به گزارش وقایع (لاگ) پنج روز از موتور جستجوی پارسی‌جو از روز ۱۳۹۶/۴/۳۱ تا ۱۳۹۶/۵/۴ است که مشخصات آن در جدول ۱ آمده است. پرس‌وجوهای چهار روز اول به عنوان پرس‌وجوی آموزش و پرس‌وجوهای روز آخر به عنوان پرس‌وجوی آزمایش در نظر گرفته شده‌اند. از پرس‌وجوهای آموزش برای وزن‌دهی به کلمات و اسناد و در نهایت توزیع اسناد بین سرورها استفاده می‌کنیم و از پرس‌وجوهای آزمایش برای ارزیابی کیفیت توزیع اسناد در هنگام ورود پرس‌وجوها بهره می‌گیریم. اطلاعات مربوط به هر پرس‌وجو شامل متن پرس‌وجو، زمان ارسال پرس‌وجو، آدرس آی‌پی و شناسه ارسال‌کننده پرس‌وجو و در نهایت، اطلاعات مربوط به نوع سیستم و مرورگر ارسال‌کننده می‌باشد. از آنجا که پرس‌وجوهای هرز (اسپم) [۲۱] نیز بخش قابل توجهی از پرس‌وجوهای وب را تشکیل می‌دهند که ممکن است بر تصمیمات مختلف موتور جستجو تأثیر نامطلوب بگذارند، ما با استفاده از تعریف قواعدی (برای مثال، اعمال حد آستانه روی فاصله زمانی مربوط به پرس‌وجوهای ارسال‌شده از یک آدرس آی‌پی یا اعمال حد آستانه بر روی فاصله زمانی بین دو پرس‌وجوی تکراری) تا جای ممکن پرس‌وجوهای هرز را فیلتر می‌کنیم که آمار مربوط به پرس‌وجوها بعد از اعمال فیلتر نیز در جدول ۱ مشاهده می‌شود.

تعدادی سند نیز به صورت تصادفی از بین سندهای خزش‌شده از وب، به عنوان نمونه‌ای از کل اسناد وب انتخاب و مورد استفاده قرار گرفته‌اند که مشخصات آنها در جدول ۲ آمده است. این اسناد صرفاً مربوط به صفحات فارسی خزش‌شده می‌باشند که ما از محتوای متنی آنها استفاده می‌کنیم.

۳-۲ پیش‌پردازش دادگان

قبل از پیش‌پردازش متن پرس‌وجوها، ابتدا تمام اطلاعات موجود در گزارش وقایع پرس‌وجو بررسی می‌شود و همان طور که قبلاً نیز گفته شد، تلاش می‌شود تا اثر ارسال پرس‌وجوهای هرز به موتور جستجو تا حد ممکن کاسته شود. سپس متن پرس‌وجوی کاربران پالایش می‌شود و فاصله‌ها یا علایم اضافی از پرس‌وجو حذف یا اصلاح می‌گردند. همین طور کلمات موجود در پرس‌وجو مورد ریشه‌یابی قرار می‌گیرند (برای مثال، پسوند ها یا های از کلمات جمع حذف می‌گردد). در این پژوهش از تصحیح کلمات دارای غلط املایی صرف‌نظر می‌کنیم.

۳-۳ استخراج مجموعه کلمات

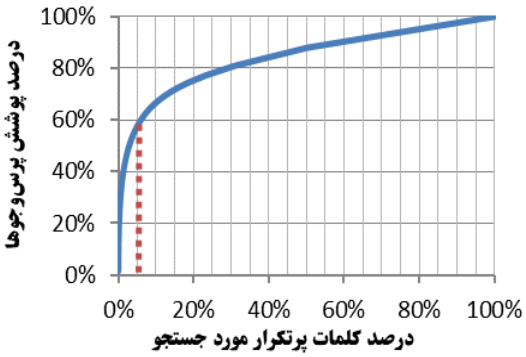
پس از اتمام مرحله پیش‌پردازش، کلمات تشکیل‌دهنده هر پرس‌وجو از هم جدا می‌شوند و هر کلمه به عنوان یک کلمه مستقل در نظر گرفته می‌شود. چون در مرحله قبل، کلمات با غلط املایی تصحیح نشده‌اند و همچنین به این علت که پردازش تمام کلمات موجود در مجموعه استخراج‌شده زمان‌بر است، مجموعه‌ای از کلمات با تکرار بالا در سابقه پرس‌وجو به عنوان کلمات مهم برای توزیع مؤثر اسناد در نظر گرفته می‌شوند. قسمت آموزش روش پیشنهادی، با استفاده از همین مجموعه کلمات انجام می‌گیرد.

۳-٤ استخراج کلمات پرتکرار از بین مجموعه کلمات

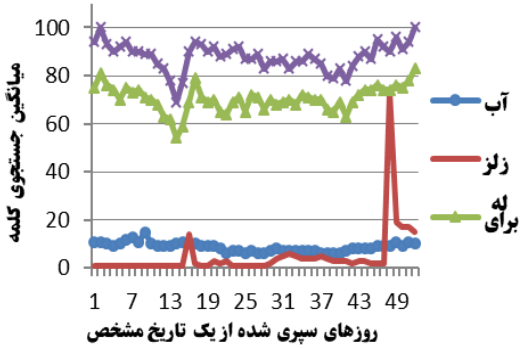
از آنجا که ترافیک پرس‌وجوهای ارسالی کاربران به موتور جستجو دائماً و به صورت لحظه‌ای تغییر می‌کند، برخی سرورها در لحظاتی دچار بار پردازشی بسیار زیادی می‌شوند. دلایل مختلفی برای پیدایش چنین باری وجود دارد [۲۲]: ۱) بروز رویدادهای غیر مترقبه مثل آتش‌سوزی یا زلزله که ناگهان مورد جستجوی زیادی قرار می‌گیرند، ۲) غیر همگون بودن الگوی جستجوی کاربران به این معنی که درصد قابل توجهی از پرس‌وجوها در ساعت محدودی در روز، مثلاً حوالی ظهر، به موتور ارسال می‌شوند به طوری که نرخ ورود پرس‌وجوها در زمان اوج ترافیک چند ده برابر نرخ میانگین می‌باشد [۷]، ۳) توزیع نامتوازن اسناد مابین سرورها به این معنی که پاسخ بیشتر پرس‌وجوها ممکن است توسط تعداد محدودی سرور تهیه شود و ۴) متفاوت‌بودن حجم محاسبات برای یافتن پاسخ پرس‌وجوهای مختلف [۲۲].

این نکته حایز اهمیت است که رسیدگی به بعضی از پرس‌وجوها نیازمند تخصیص منابع محاسباتی بسیار بیشتری در مقایسه با بقیه پرس‌وجوها است. برای مثال، پرس‌وجوهایی که با تعداد زیادی سند مطابقت داشته باشند، منجر به انجام محاسبات سنگین و زمان‌بر در داخل سرورها (برای بازیابی، امتیازدهی و رتبه‌بندی) می‌شوند. به چنین پرس‌وجوهایی که زمان پردازش طولانی دارند پرس‌وجوهای سنگین گفته می‌شود [۲۲]. هرچند ممکن است درصد پرس‌وجوهای سنگین در کل اندک باشد، ولی همین درصد اندک بر زمان پردازش پرس‌وجوهای دیگر اثر می‌گذارد و به طور کلی، نرخ خروجی سیستم را به ویژه در زمان اوج ترافیک، کاهش می‌دهد. راه حل اصولی برای پردازش پرس‌وجوهای سنگین، بهره‌گیری از ظرفیت همه سرورها برای توزیع بهتر اسناد است تا برخی سرورها دچار بار زیاد نشوند و در سیستم ایجاد گلوگاه نکنند [۱۰]. بنابراین وجود توازن بار بین سرورها، مخصوصاً در زمان اوج ترافیک، اهمیت بیشتری پیدا می‌کند.

در این بخش، کلماتی که بار زیادی به سرورها تحمیل می‌کنند، از جمله کلمات موجود در پرس‌وجوهای سنگین، استخراج می‌شوند. در این پژوهش، پرس‌وجوهایی را سنگین در نظر گرفته‌ایم که تعداد سند برگشتی در پاسخ آنها بیش از ۱٪ از کل اسناد باشد. این پرس‌وجوها در مجموع کمتر از ۱۰٪ پرس‌وجوهای ارسالی به موتور را تشکیل می‌دهند، ولی



شکل ۳: توزیع تجمعی استفاده از کلمات پرتکرار در پرس‌وجوها.



شکل ۴: میزان استفاده از کلمات مختلف در گوگل بر حسب زمان.

میانگین زمان پاسخ آنها (بر طبق نتایج به دست آمده از موتور جستجوی پارسی‌جو) تقریباً سه برابر مابقی پرس‌وجوها است.

در شکل ۳ نمودار توزیع تجمعی برای میزان استفاده از کلمات موجود در سابقه پرس‌وجو رسم شده است. در این نمودار، کلمات بر اساس تعداد تکرار در پرس‌وجوها از بیشترین به کمترین (در محور افقی) مرتب شده‌اند. محور افقی درصد کلمات پرتکرار را نشان می‌دهد و محور عمودی درصد پوشش کلمات پرس‌وجوها است. در واقع، این نمودار نشان می‌دهد که با استفاده از چه درصدی از کلمات پرتکرار، چه درصدی از کلمات همه پرس‌وجوها پوشش داده می‌شوند. برای مثال، کلمه {علی} می‌تواند ۵۰٪ از پرس‌وجوی {علی محمد} را پوشش دهد. همان طور که شکل نشان می‌دهد، تنها با در نظر گرفتن ۶٪ از پرتکرارترین کلمات، نزدیک به ۵۷٪ از کلمات تشکیل‌دهنده پرس‌وجوها پوشش داده می‌شوند. نکته قابل توجه آن است که این درصد محدود از کلمات پرتکرار می‌تواند بیش از ۹۰٪ کلمات پرس‌وجوهای سنگین را پوشش دهد، یعنی پرس‌وجوهای سنگین عمدتاً از کلمات پرتکرار تشکیل شده‌اند. بر حسب این نتایج، ما از ۶٪ کلمات با بیشترین تکرار به عنوان مجموعه کلمات پرتکرار استفاده می‌کنیم. با تمرکز بر این مجموعه کلمات، هم هزینه محاسبات آتی الگوریتم کاهش پیدا می‌کند و هم این که نیاز به اجرای مداوم الگوریتم توزیع نیست، چون مجموعه کلمات پرتکرار به ندرت تغییر می‌کنند.

۳-۵ وزن‌دهی به کلمات پرتکرار

پس از شناسایی کلمات پرتکرار، برای تخمین میانگین بار ایجادشده توسط هر یک از آنها بر سرورها، هر کلمه با توجه به چگونگی استفاده از آن کلمه در روزهای مختلف، وزن‌دهی می‌شود. به عنوان نمونه، شکل ۴ که از سابقه پرس‌وجوی کاربران گوگل به دست آمده، نسبت میزان استفاده از چهار کلمه {زلزله، آب، از، برای} را در بازه‌ای مشخص از سال نمایش می‌دهد. کلمات "برای" و "از" از جمله کلمات توقفی[1] هستند که به طور متداول در پرس‌وجوها ظاهر می‌شوند، به این معنی که میانگین تکرار بالایی هم در پرس‌وجوها و هم در اسناد دارند. بنابراین بار قابل توجهی روی سرورها ایجاد می‌کنند. از همین رو، میانگین تعداد تکرار هر کلمه در هر روز، به عنوان شاخصی مهم برای وزن‌دادن به آن کلمه لحاظ می‌شود.

علاوه بر این، مشاهده می‌شود که کلمه "زلزله" در یک بازه زمانی کوتاه به مقدار زیادی جستجو شده، در حالی که در بقیه زمان‌ها میانگین جستجوی پایینی داشته است. این کلمه ممکن است در آن بازه زمانی کوتاه، بار تعداد کمی از سرورها را به میزان زیادی بالا ببرد و در نتیجه توازن بار بین سرورها را بر هم بزند. به همین دلیل، "انحراف از میانگین" نیز به عنوان شاخصی دیگر برای وزن‌دادن به کلمه در نظر گرفته می‌شود. انحراف از میانگین، میزان انحراف معیار را بین دفعات جستجوی یک کلمه در روزهای مختلف نشان می‌دهد. ایده اصلی آن است که کلمه‌ای که در یک زمان به صورت انفجاری در پرس‌وجوها ظاهر شده، احتمال دارد که در آینده نیز مجدداً با نرخ بالا و به صورت ناگهانی ظاهر شود.

اضافه بر دو شاخص فوق، به کلمه‌ای که در تعداد بیشتری سند آمده باید قاعدتاً وزن بیشتری داده شود به این دلیل که جستجوی آن کلمه منجر به بازیابی و رتبه‌بندی تعداد بیشتری سند خواهد شد که این خود مستلزم انجام محاسبات بیشتری است. در سیستم‌های بازیابی اطلاعات، معیاری با نام فرکانس سند برای هر کلمه تعریف می‌شود که تعداد اسناد حاوی آن کلمه را نشان می‌دهد [۲۳]. ما از مقدار فرکانس سند هر کلمه به عنوان شاخص سوم برای وزن‌دهی به کلمات استفاده می‌کنیم. چون این معیار برای برخی کلمه‌ها از جمله حروف اضافه مقدار بسیار بزرگی دارد، معمولاً از لگاریتم آن در فرمول‌ها استفاده می‌شود [۲۳].

بنابراین به ازای هر کلمه t، وزن کلمه w_t از (۱) محاسبه می‌شود. در این رابطه، N_t میانگین تعداد دفعات استفاده از کلمه t در هر روز، σ_{N_t} انحراف از میانگین برای تعداد دفعات استفاده از کلمه t در روزهای مختلف و df_t مقدار فرکانس سند برای کلمه t است. کلیه این مقادیر از دادگان آموزش محاسبه می‌شوند

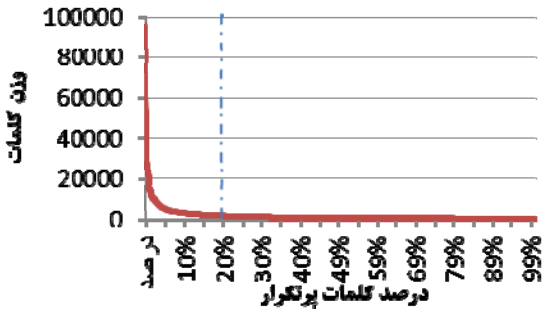
$$w_t = (\alpha \times N_t + \beta \times \sigma_{N_t}) \times \log(df_t) \qquad (۱)$$

در رابطه بالا، جمع ضرایب α و β برابر با یک می‌باشد. از این رابطه، صرفاً برای وزن‌دهی به مجموعه کلمات پرتکرار استفاده می‌شود و وزن مابقی کلمات صفر در نظر گرفته می‌شود. با توجه به ارزیابی‌های انجام‌شده، اهمیت میانگین بیشتر از انحراف از میانگین است که البته منطقی نیز می‌باشد. در نتیجه، ضریب α باید بزرگ‌تر از β انتخاب گردد. طبق نتایج ارزیابی، مقادیر $\alpha = ۰/۷$ و $\beta = ۰/۳$ بهترین حالت وزن‌دهی به کلمات را با توجه به دادگان ما نتیجه می‌دهد. در فصل ارزیابی، نحوه تنظیم این پارامترها مورد بحث قرار خواهد گرفت.

۳-۶ پیمانه‌بندی کلمات پرتکرار

از آنجا که دادگان آموزش مربوط به مدت زمان محدودی بوده و لزوماً روند جستجوی آتی کاربران را به صورت دقیق نشان نمی‌دهد، احتمال رخداد خطا در محاسبه وزن کلمات وجود دارد. یکی از روش‌های ساده برای کاهش اثرات خطا در دادگان آموزش، استفاده از پیمانه‌بندی کلمات است. اگر به نمودار توزیع وزن برای کلمات پرتکرار که در شکل ۵ آمده توجه کنید، مشخص است که تعداد کمی از این کلمات، تکرار بسیار زیادی در پرس‌وجوها دارند. برای مثال، فقط ۲۰٪ از کلمات پرتکرار دارای

1. Stop Words



شکل ۵: توزیع وزن کلمات پرتکرار.

وزن بیشتر از ۲۰۰۰ هستند و مابقی کلمات پرتکرار از وزن کمتری برخوردارند. برای کاهش خطا و همین طور برای سادگی در محاسبات، کلمات پرتکرار را به دو پیمانه "پراهمیت" و "کم‌اهمیت" تقسیم می‌کنیم: ۲۰٪ ۴کلمات با بیشترین تکرار در پیمانه اول و ۸۰٪ مابقی کلمات در پیمانه دوم قرار می‌گیرند.

در مرحله بعد، میانگین وزن کلمات هر پیمانه را در دادگان آموزش محاسبه و به جای وزن تمام کلمات آن پیمانه قرار می‌دهیم. این وزن جدید را برای کلمه t با $w_b(t)$ نشان می‌دهیم. میانگین وزن کلمات موجود در پیمانه پراهمیت تقریباً برابر با ۱۰ و میانگین وزن کلمات در پیمانه کم‌اهمیت تقریباً ۱ به دست آمده است. پس از این مقادیر برای وزن‌دهی همه کلمات پرتکرار موجود در هر یک از پیمانه‌ها استفاده می‌کنیم. لازم به ذکر است که برای تفکیک بهتر کلمات، می‌توان تعداد پیمانه‌ها را بیشتر در نظر گرفت که البته این خود می‌تواند منجر به احتمال خطای بیشتر در پیمانه‌بندی شود (چون آمار موجود برای یک کلمه در دادگان آموزش دقیقاً منطبق با دادگان آزمایش نیست). در فصل ارزیابی، در مورد تنظیم تعداد پیمانه در پیمانه‌بندی کلمات صحبت خواهیم کرد.

۳-۷ وزن‌دهی به تمام اسناد

پس از محاسبه وزن هر کلمه، وزن هر یک از سندها قابل محاسبه است. به طور خلاصه، هدف از وزن‌دهی آن است که سندهایی را که با احتمال بیشتری توسط موتور جستجو به عنوان نتیجه بازگردانده می‌شوند، تشخیص دهیم. در بخش قبل گفته شد که وزن تمام کلمات موجود در هر پیمانه، مساوی در نظر گرفته می‌شود. در این صورت، هرچه یک سند حاوی کلمات پرتکرار بیشتری (به ویژه کلمات پراهمیت بیشتری) باشد، احتمال بازگشت آن سند به عنوان نتیجه بیشتر است. برای مثال، احتمال بازگشت دو سند که یکی حاوی یک کلمه "دانلود" و دیگری حاوی یک کلمه "ایران" باشد، با این فرض که کلمه "دانلود" و "ایران" جزو کلمات پرتکرار پراهمیت هستند، برابر در نظر گرفته می‌شود. در نتیجه، هر دو سند وزن یکسانی خواهند داشت. ولی اگر یکی از این دو سند حاوی یک یا چند کلمه پرتکرار دیگر نیز باشد، در این صورت احتمال بازگشت آن سند به عنوان پاسخ بیشتر بوده و در نتیجه، وزن سند بیشتر خواهد بود. به بیان ساده، وزن سند d از طریق مجموع وزن پیمانه همه کلمات موجود در سند محاسبه می‌شود

$$w_d = \sum_{t \in d} tf_{t,d} \times w_b(t) \qquad (۲)$$

در رابطه بالا، $tf_{t,d}$ تعداد تکرار کلمه t در سند d می‌باشد و $w_b(t)$ وزن پیمانه مربوط به کلمه t است. وزن پیمانه مربوط به کلمات کم‌تکرار نیز صفر در نظر گرفته می‌شود. برای پرهیز از اضافه‌شدن بی‌جهت وزن اسناد، وزن سندهایی را که تنها از حروف اضافه تشکیل شده‌اند و کلمه پرتکرار دیگری ندارند، صفر در نظر می‌گیریم.

ارزیابی اولیه ما حاکی از آن است که فرمول وزن در (۲) به سمت سندهای طولانی متمایل است، یعنی به دلیل داشتن تعداد زیادی کلمه، وزن بیشتری برای سندهای با حجم متنی زیاد لحاظ می‌کند. این در حالی است که ممکن است این سندها کمتر در پاسخ به پرس‌وجوها بازگشت داده شوند (طبق تحقیقات گذشته، بیش از نیمی از اسناد هیچ گاه توسط موتور جستجو بازگشت داده نمی‌شوند [۲]). یعنی هرچه حجم یک سند بیشتر و دارای کلمات متنوع بیشتری باشد، طبیعی است که ارتباط سند با کلمات داخل آن کمتر می‌شود و در نتیجه، ممکن است با احتمال کمتری در پاسخ به یک پرس‌وجو بازگشت داده شود. برای کاهش اثر طول اسناد، وزن هر سند بر لگاریتم طول متنی آن تقسیم می‌شود تا وزن نهایی حاصل شود

$$w_d = \frac{\sum_{t \in d} tf_{t,d} \times w_b(t)}{\log size(d)} \qquad (۳)$$

در رابطه بالا $size(d)$ به معنای اندازه سند d بر حسب تعداد کلمات آن می‌باشد.

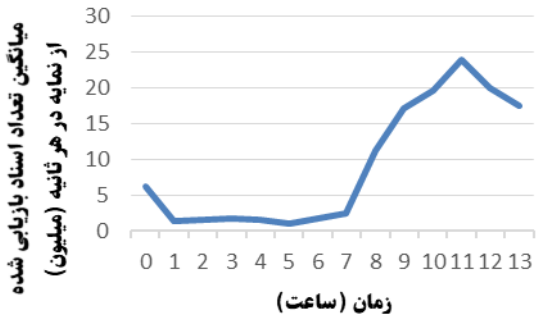
۳-۸ سیاست توزیع اسناد

حال با در دسترس بودن وزن اسناد، از روشی مشابه با "پرکردن کوچک‌ترین" [۸] برای توزیع اسناد استفاده می‌کنیم. در این روش به سرورها متناسب با سندهای داخل آنها وزنی نسبت داده می‌شود که برابر با مجموع وزن اسناد داخل آن سرور است. سندها یک به یک و بدون ترتیب خاصی پیمایش می‌شوند و برای انتساب هر سند جدید به یکی از سرورها، سروری به عنوان سرور مقصد انتخاب می‌شود که کمترین وزن کنونی را داشته باشد.

چون ما صرفاً کلمات پرتکرار را مورد ملاحظه قرار می‌دهیم، ممکن است اسناد زیادی دارای وزن صفر باشند، یعنی تعداد کلمات پرتکرار داخل آنها صفر باشد. برای مثال، تعداد زیادی از اسناد در مجموعه دادگان که به زبان‌های غیر فارسی هستند، به دلیل نداشتن کلمات پرتکرار از پرس‌وجوها که اغلب به زبان فارسی هستند دارای وزن صفر می‌باشند. در این صورت، سیاست فوق نمی‌تواند توازن را در توزیع اسناد بی‌ارزش یا کم‌ارزش مابین سرورها برقرار نماید. برای مثال، ممکن است اکثر اسناد دارای وزن صفر در یک سرور قرار گیرند، چون این اسناد اصلاً منجر به افزایش وزن سرور نمی‌شوند. در این حالت، ظرفیت نمایه برخی از سرورها خیلی سریع و به صورت نامتوازن پر می‌شود. برای رفع این ایراد، سیاست کلاسیک پرکردن کوچک‌ترین را به این صورت تغییر می‌دهیم که به ازای تمام اسناد با وزن صفر، این اسناد را به صورت مجزا و صرفاً بر اساس حجمشان به صورت عادلانه بین سرورها توزیع می‌کنیم. به این دلیل که با توزیع متوازن حجم، طول لیست اسناد برای هر کلمه در نمایه سرورها، تقریباً یکسان می‌شود.

۳-۹ ملاحظات اجرا در محیط واقعی

روند اجرای الگوریتم پیشنهادی برای توزیع اسناد در محیط واقعی به این شرح است: با در نظر گرفتن مجموعه قابل توجهی از پرس‌وجوها (مثلاً پرس‌وجوهای یک ماه گذشته)، مجموعه کلمات مهم و وزن آنها استخراج می‌شود. صفحات جدید به صورت مداوم توسط خزشگر، کشف و خزش می‌گردند یا احیاناً صفحات قدیمی به روز می‌شوند. با ورود هر صفحه، کلمات مهم آن صفحه استخراج شده و سپس آن صفحه طبق روالی که قبلاً گفته شد وزن‌دهی می‌شود. در آخر، هر صفحه به ترتیب در سرور دارای کمترین وزن مجموع (تا آن زمان) ذخیره می‌گردد و به وزن



شکل ۶. میانگین بار پردازشی سرورها در یک بازه ۱۳ساعته.

سرور انتخاب‌شده نیز افزوده می‌شود. لازم به ذکر است که مجموعه کلمات مهم و وزن پیمانه آنها با گذشت زمان چندان دچار تغییر نمی‌شود. با وجود این، وزن کلمات با در نظر گرفتن یک پنجره مثلاً یک‌ماهه دائماً محاسبه و به روز می‌شود. با توجه به تغییرات جزئی احتمالی در وزن کلمات، نیازی به به روز رسانی وزن اسناد قدیمی و سرورهای حاوی آنها نیست.

٤- ارزیابی

در این بخش، ابتدا مدل داده و معیارهای ارزیابی را بیان می‌کنیم، سپس به ارائه نتایج ارزیابی روش پیشنهادی می‌پردازیم و در نهایت، آن را با روش‌های دیگر مقایسه می‌نماییم.

چون ارزیابی واقعی میزان بار پردازشی سرورها کاری زمان‌بر، پرهزینه و غیر دقیق است، محیط ارزیابی در این تحقیق، مشابه با کارهای گذشته [۸] بر پایه شبیه‌سازی می‌باشد (عوامل متعددی نظیر وضعیت کاری سیستم عامل در بار واقعی سرورها تأثیرگذار هستند که به صورت لحظه‌ای نیز تغییر می‌کنند). ما کل اسناد موجود در دادگان را بر اساس هر یک از روش‌های توزیع به تعدادی نمایه تقسیم می‌کنیم. ساخت نمایه با استفاده از نمایه‌ساز معروف لوسین [۲۴] صورت می‌گیرد. در آزمایشات ما، تعداد نمایه‌ها برابر با ۸ بوده و حجم اسناد داخل هر یک از نمایه‌ها تقریباً برابر می‌باشد. فرض می‌کنیم که هر نمایه به صورت مجزا روی یک سرور قرار می‌گیرد. در نهایت، مجموعه پرس‌وجوهای آزمایش را یک به یک روی هر یک از نمایه‌ها جستجو می‌کنیم و بر حسب تعداد اسناد بازگشتی و تعداد تطابق‌های پرس‌وجو در داخل نمایه، بار پردازشی هر سرور را تخمین می‌زنیم. در مورد معیارهای ارزیابی در ادامه بیشتر صحبت خواهیم کرد.

٤-١ مدل داده (تقسیم دادگان)

در ابتدا، پرس‌وجوهای جمع‌آوری شده در پنج روز متوالی به دو مجموعه آموزش و آزمایش تقسیم می‌شوند. پرس‌وجوهای چهار روز اول به عنوان مجموعه آموزش و پرس‌وجوهای روز پنجم به عنوان مجموعه آزمایش در نظر گرفته می‌شوند. برای وضوح بیشتر نتایج در نمودارهای زمانی، فقط نتایج پرس‌وجوهای واردشده در نیمه اول روز پنجم نمایش داده می‌شوند که البته این نیمه حاوی ساعات اوج ورود پرس‌وجوها نیز می‌باشد (در نمودارهای غیر زمانی، نتایج کل روز پنجم در نظر گرفته شده‌اند). با این تقسیم‌بندی، پیوستگی پرس‌وجوها و روندهای مختلف جستجوی کاربران و توزیع پرس‌وجوها حفظ می‌گردد. یادآوری می‌شود که برای محاسبه پارامترهای میانگین و انحراف از میانگین تعداد هر پرس‌وجو در بازه زمانی روزانه به زمان ارسال هر پرس‌وجو و پیوستگی دادگان پرس‌وجو نیاز داریم.

در شکل ۶، تخمینی از میانگین بار پردازشی سرورها (بر حسب تعداد اسناد بازیابی‌شده از نمایه در هر ثانیه) در یک بازه زمانی ۱۳ساعته (از صفر بامداد تا ۱۳ ظهر) مشاهده می‌شود. مشخص است که در ساعات اولیه صبح میانگین بار پردازشی سرورها پایین است، چون تعداد پرس‌وجوها بسیار کم است. در نتیجه، ایجاد توازن بار برای این محدوده زمانی اهمیت چندانی ندارد ولی در بازه زمانی حدود ۱۱ تا ۱۲ ظهر که بار پردازشی سرورها حداکثر می‌باشد، این مسئله اهمیت زیادی پیدا می‌کند.

٤-٢ معیارهای ارزیابی

برای تخمین میزان بار پردازشی هر سرور در طول پردازش یک پرس‌وجو، باید ابتدا عوامل مؤثر بر میزان پردازش را مشخص کنیم. فرایند پردازش یک پرس‌وجو در هر سرور، با جستجوی نمایه آن سرور و سپس اجرای الگوریتم رتبه‌بندی انجام می‌پذیرد. به این معنی که پردازش‌ها، عمدتاً شامل دو قسمت جستجو و بازیابی نمایه و اجرای الگوریتم رتبه‌بندی است. مهم‌ترین عامل تأثیرگذار بر هر دو قسمت، تعداد سندهای بازگشتی از سرور می‌باشد [۸]. هرچه تعداد سندهای بازگشتی یک پرس‌وجو از یک سرور بیشتر باشد، بار پردازشی آن سرور نیز بیشتر خواهد بود. در کنار این عامل، تعداد رخدادهای کلمات پرس‌وجو در داخل اسناد موجود در سرور نیز بر میزان پردازش آن سرور تأثیرگذار است. بنابراین برای تخمین بار پردازشی هر سرور از دو معیار مجموع فرکانس سند و مجموع فرکانس کلمه به صورت مجزا استفاده می‌کنیم. این دو معیار در ادامه دقیق‌تر معرفی می‌شوند.

از آنجا که در رویکرد توزیع مبتنی بر سند، نتایج هر پرس‌وجو زمانی آماده می‌شوند که آخرین سرور پاسخ خود را به سرور مرکزی ارسال کرده باشد، مهم است که زمان پردازش انجام‌شده در هر سرور تقریباً برابر با زمان پردازش سایر سرورها باشد. پس لازم است تعداد سندهای بازگشتی از هر سرور به ازای مجموعه پرس‌وجوهای آموزش تقریباً یکسان باشد. برای برآورد توازن بار بین سرورها، از انحراف معیار مقادیر به دست آمده برای دو معیار فوق (یعنی مجموع فرکانس سند و مجموع فرکانس کلمه) در همه سرورها استفاده می‌کنیم. برای مثال، هرچه انحراف معیار مجموع فرکانس سند در سرورها کمتر باشد، یعنی توازن بیشتری بین بار پردازشی آنها برقرار است. در مقالات گذشته نیز از همین معیار برای ارزیابی توازن بار استفاده شده است [۸].

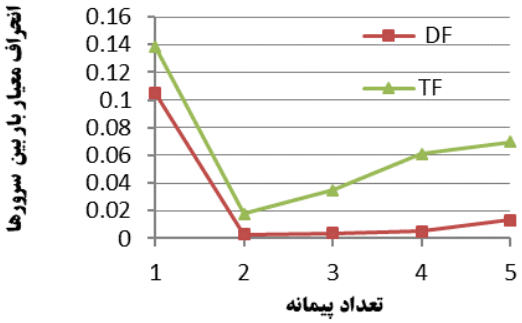
٤-٢-١ معیار مجموع فرکانس سند (DF) برای هر سرور

معیار مجموع فرکانس سند برای هر سرور به صورت جداگانه و طبق (۴) محاسبه می‌شود. به ازای هر پرس‌وجوی q، D_q^s مجموعه اسناد موجود در سرور s را که حاوی تمام کلمات پرس‌وجوی q هستند نمایش می‌دهد. به بیان ساده‌تر، D_q^s نشان می‌دهد که به ازای پرس‌وجوی q، چه اسنادی از سرور s به عنوان پاسخ برگشت داده شده‌اند. مقایسه بار پردازشی هر یک از سرورها به ازای مجموعه پرس‌وجوهای آزمایش (مجموعه Q) انجام می‌شود. بنابراین مجموع تعداد سندهایی را که به ازای کل پرس‌وجوهای مجموعه آزمایش توسط سرور s برگردانده می‌شود با DF^s نشان می‌دهیم. معیار DF^s برای سرور s نشان می‌دهد که آن سرور چه تعداد سند را برای کل پرس‌وجوهای آزمایش از نمایه خود بازیابی و رتبه‌بندی کرده است [۸]

$$DF^s = \sum_{(q \in Q)} \left| D_q^s \right| \qquad (۴)$$

٤-٢-٢ معیار مجموع فرکانس کلمه (TF) برای هر سرور

اگرچه تعداد سندهای بازگشتی از هر سرور، بار پردازشی تقریبیِ



شکل ۷: تأثیر استفاده از تعداد پیمانه‌های مختلف بر توازن بار سرورها.

ایجادشده روی آن سرور را نشان می‌دهد، اما با این معیار نمی‌توان مشخص کرد که به ازای هر سند، چه میزان پردازش انجام شده است. به همین دلیل به معیار دقیق‌تری برای بررسی بار ایجادشده بر هر سرور نیاز داریم. به طور خلاصه، هرچه در یک سندِ برگشت داده شده توسط موتور جستجو، تعداد رخداد کلمات پرس‌وجو بیشتر باشد، آن سند بار پردازشی بیشتری به نسبت سایر سندها ایجاد می‌کند. به این دلیل که در ساختار نمایه، جایگاه هر کلمه (از کلمات جستجو) در سند نیز آورده شده و برای یافتن پاسخ به پرس‌وجو و رتبه‌بندی مورد استفاده قرار می‌گیرد. بنابراین هرچه تعداد تکرار یک کلمه در سند بیشتر باشد، به پردازش بیشتری نیاز دارد. به همین دلیل برای بررسی دقیق‌تر بار ایجادشده بر هر سرور، تعداد رخداد کلمات هر پرس‌وجو را در سندهای برگشتی برای آن پرس‌وجو محاسبه می‌کنیم و این معیار را به عنوان معیار دوم در ارزیابی مورد استفاده قرار می‌دهیم. لازم به ذکر است که مشابه کارهای گذشته [۹]، در این مقاله هم فرض ما آن است که همه کلمات پرس‌وجو باید در سند بازگشتی وجود داشته باشند.

فرض کنید $TF_t(d)$ تابعی است که نشان می‌دهد کلمه t در سند d چند بار تکرار شده است. خروجی این تابع در صورت عدم رخداد کلمه در سند، صفر و در صورت رخداد کلمه، برابر با تعداد رخداد خواهد بود. هزینه پردازش یک سند برای امتیازدهی جهت رتبه‌بندی تا حد زیادی به تعداد تکرار کلمات پرس‌وجو در سند وابسته است. بنابراین بار پردازشی ایجادشده بر روی هر سرور (برای رتبه‌بندی)، به دلیل پردازش پرس‌وجوی q، به مجموع تکرار همه کلمات پرس‌وجو در اسناد بازگشت داده شده بستگی دارد. به همین دلیل، مجموع تکرار کلیه کلمات پرس‌وجوی q را در هر یک از اسناد موجود در مجموعه D_q^s محاسبه می‌کنیم و آن را با TF_q^s نمایش می‌دهیم

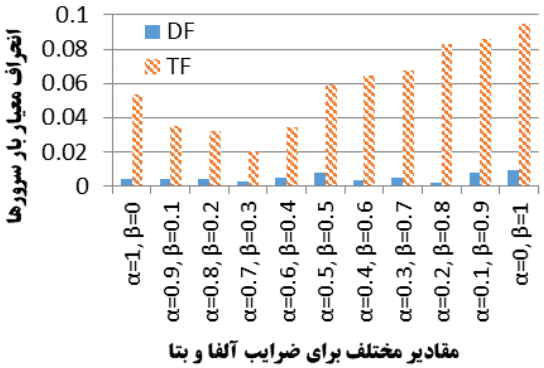
$$TF_q^s = \sum_{(d \in D_q^s)} \sum_{(t \in q)} TF_t(d) \qquad (۵)$$

در نهایت، مجموع بار پردازشی ایجادشده روی سرور s را به ازای مجموعه پرس‌وجوی آزمایش Q با TF^s نشان می‌دهیم

$$TF^s = \sum_{(q \in Q)} TF_q^s \qquad (۶)$$

۳-٤ نتایج ارزیابی

برای ارزیابی روش پیشنهادی و همین طور مقایسه آن با روش‌های دیگر، ما پرس‌وجوهای موجود در مجموعه آموزش را بر روی نمایه‌های ایجادشده توسط هر یک از روش‌های توزیع جستجو می‌کنیم و نتایج جستجو را به دست می‌آوریم. سپس بر حسب نتایج به دست آمده از هر نمایه، معیارهای مجموع فرکانس سند و مجموع فرکانس کلمه را محاسبه و برای تخمین بار پردازشی سرورها از آنها استفاده می‌کنیم (طبق فرض ما، هر نمایه معادل با یک سرور می‌باشد). در نهایت از انحراف معیار



شکل ۸: نتایج مقادیر مختلف آلفا و بتا در فرمول وزن‌دهی کلمات.

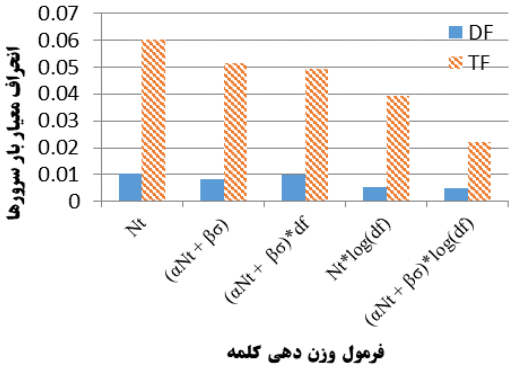
فرکانس سند و فرکانس کلمه در همه سرورها برای ارزیابی توازن بار بین آنها استفاده می‌کنیم.

ابتدا تأثیر استفاده از تکنیک پیمانه‌بندی در وزن‌دهی به کلمات پرتکرار بررسی می‌شود. نتایج ارزیابی نشان می‌دهند که استفاده از پیمانه‌بندی، انحراف معیار بار روش پیشنهادی را از نظر هر دو معیار فرکانس سند (DF) و فرکانس کلمه (TF) به طور میانگین ۹٪ بهبود می‌دهد. دلیل موفقیت پیمانه‌بندی آن است که تنها بخش محدودی از کلمات، مکرراً در پرس‌وجوها ظاهر می‌شوند و هر یک، بسته به نوع رویداد یا نیاز کاربران، در زمان‌های مختلفی اهمیت بیشتری پیدا می‌کنند. بنابراین استناد به سابقه پرس‌وجوها که مربوط به بازه زمانی محدودی است، لزوماً اهمیت و وزن کلمات را به طور دقیق نشان نمی‌دهد. در حالی که به دنبال اجرای پیمانه‌بندی، پیمانه‌ای که برای هر کلمه به دست می‌آید، چندان وابسته به بازه زمانی دادگان آموزش نیست و به ندرت دچار تغییر می‌شود.

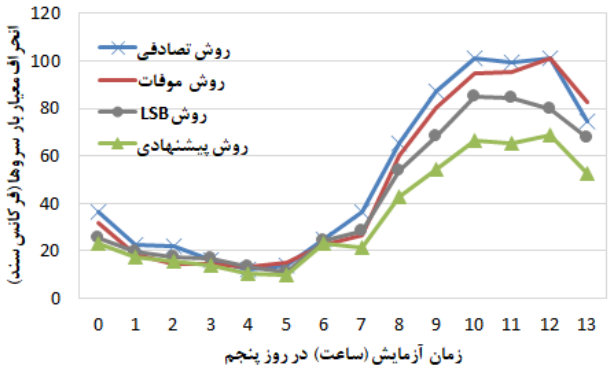
در شکل ۷، تأثیر استفاده از تعداد پیمانه‌های مختلف بر انحراف معیار بار مربوط به مقادیر فرکانس سند و فرکانس کلمه مشاهده می‌شود. با توجه به این نتایج مشخص است که پیمانه‌بندی مجموعه کلمات پرتکرار به دو دسته بهترین حالت توازن بار را بین سرورها ایجاد می‌کند. استفاده از تنها یک پیمانه، به همه کلمات پرتکرار به یک میزان اهمیت می‌دهد که البته اشتباه است و استفاده از تعداد پیمانه‌های بیشتر نیز میزان خطای پیمانه‌بندی بین کلمات موجود در دادگان آموزش و آزمایش را افزایش می‌دهد (یعنی کلمه‌ای که در دادگان آموزش در یک پیمانه قرار دارد، ممکن است با لحاظ‌کردن دادگان آزمایش در پیمانه دیگری واقع شود).

شکل ۸ میزان اثرگذاری تنظیم ضرایب α و β در (۱) را بر عملکرد روش پیشنهادی نشان می‌دهد. کمترین انحراف معیار بار سرورها با مقادیر $\alpha = ۰/۷$ و $\beta = ۰/۳$ به دست آمده و بیشترین انحراف معیار بار هم وقتی حاصل شده که یکی از پارامترهای میانگین و یا انحراف از میانگین دارای ضریب صفر باشد. ملاحظه می‌شود که تنظیم مناسب این دو ضریب، تأثیر زیادی بر عملکرد روش پیشنهادی دارد به طوری که بهترین نتیجه تا چهار برابر نسبت به بدترین نتیجه بهبود داشته است. البته همان طور که از قبل هم پیش‌بینی می‌شد، تأثیر ضریب α (یعنی ضریب میانگین) بیشتر از ضریب β (یعنی ضریب انحراف از میانگین) در فرمول وزن کلمه می‌باشد.

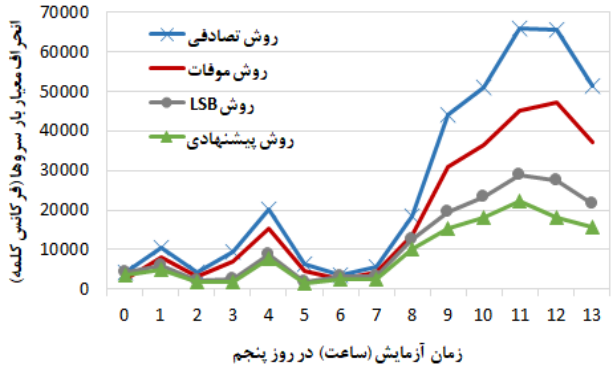
سیاست مورد استفاده در روش پیشنهادی برای توزیع اسناد، سیاست پرکردن کوچک‌ترین است. لیکن تغییر کوچکی در این سیاست اعمال کردیم بدین ترتیب که اسناد با وزن صفر به صورت عادلانه و یکنواخت بین سرورها توزیع می‌شوند. همین تغییر ساده باعث بهبود تقریباً ۱۰ درصدی در انحراف معیار مجموع فرکانس سند می‌شود (میزان بهبود برای فرکانس کلمه حدود ۲٪ است).



شکل ۹: نتایج فرمول‌های مختلف برای وزن‌دهی کلمات.



شکل ۱۰: انحراف معیار مجموع فرکانس سند در روش‌های مختلف.



شکل ۱۱: انحراف معیار مجموع فرکانس کلمه در روش‌های مختلف.

فرمول وزن‌دهی به کلمات پرتکرار در (۱) نقش اصلی را در کیفیت الگوریتم پیشنهادی ایفا می‌کند. برای بررسی تأثیر این فرمول، ما توابع مختلفی را برای وزن‌دهی انتخاب نمودیم که نتایج آنها را در شکل ۹ مشاهده می‌کنید. اجزای این توابع همان مقادیر میانگین تکرار کلمه (N_t)، انحراف از میانگین کلمه (σ) و فرکانس سند (df) می‌باشند. مقادیر α و β در هر یک از فرمول‌ها به مقادیری که بهترین نتیجه را دهد تنظیم می‌شوند. ستون اول (از راست) در واقع همان فرمول مورد استفاده در روش پیشنهادی، یعنی (۱) می‌باشد که منجر به بهترین نتیجه شده است. عدم استفاده از پارامتر انحراف از میانگین در این فرمول (ستون دوم از راست)، انحراف معیار مربوط به فرکانس سند و فرکانس کلمه را به ترتیب ۱۰٪ و ۸۰٪ بدتر می‌کند. همین طور لحاظنکردن لگاریتم فرکانس سند در این فرمول (ستون دوم از چپ) انحراف معیار سند و کلمه را به ترتیب ۳۰٪ و ۱۵۰٪ بدتر می‌کند.

در آخر، ما نتایج به دست آمده از روش پیشنهادی خود را با سه روش دیگر مقایسه می‌کنیم. یکی از معروف‌ترین تحقیقات صورت‌گرفته در زمینه توزیع اسناد، روش موفات و همکاران [۸] می‌باشد که در بخش تحقیقات گذشته معرفی گردید. ما به اختصار این روش را روش موفات می‌نامیم. روش دیگری که آن را مورد مقایسه قرار می‌دهیم، روش معرفی‌شده در [۲] است و همان طور که در بخش تحقیقات گذشته گفته شد، از رویکرد توزیع مبتنی بر سند تبعیت می‌نماید. در نتایج ارزیابی، این روش را به اختصار روش LSB می‌نامیم. بیشتر روش‌هایی که اخیراً معرفی شده‌اند در دسته جستجوی انتخابی (موضوعی) قرار می‌گیرند که با هدف پایین‌آوردن هزینه جستجو یا تعداد اسناد مورد جستجو طراحی شده‌اند. در این روش‌ها، سندهای دارای موضوع یکسان در یک مجموعه سرور ذخیره می‌شوند. با ورود یک پرس‌وجو ابتدا دسته آن پرس‌وجو تعیین شده و سپس پرس‌وجو به سرورهای مربوط به دسته مورد نظر ارسال می‌شود. عموماً این روش‌ها به توازن بار بسیار بدی منجر می‌شوند زیرا بیشتر اسناد مربوط به پرس‌وجوهای داغ همه در یک یا چند سرور مشخص ذخیره می‌گردند. بنابراین مقایسه با این روش‌ها منطقی و عادلانه نمی‌باشد. روش توزیع تصادفی یا چرخشی روشی است که تقریباً در همه مقالات اخیر به عنوان یکی از رایج‌ترین و محبوب‌ترین روش‌ها مورد ارزیابی قرار گرفته است [۱۳] و [۱۶]. در این روش، اسناد یک‌به‌یک و به صورت چرخشی (راند- رابین) به سرورها داده می‌شوند. ما نیز برای مقایسه از این روش استفاده می‌کنیم.

نتایج مربوط به انحراف معیار بار سرورها با توجه به مجموع فرکانس سند و مجموع فرکانس کلمه به ترتیب در شکل‌های ۱۰ و ۱۱ نشان داده شده‌اند. محور افقی در این دو نمودار، بازه ۱۳ساعته مربوط به پرس‌وجوهای آموزش می‌باشد. روش تصادفی بدترین عملکرد را به ویژه از نظر فرکانس کلمه داشته است که البته منطقی می‌باشد، زیرا هیچ توجهی به تأثیر احتمالی توزیع اسناد بر بار پردازشی سرورها ندارد. روش پیشنهادی توانسته است در همه ساعات، عملکرد بهتری نسبت به سه روش دیگر از خود نشان دهد به طوری که میانگین برتری آن نسبت به روش موفات از نظر فرکانس سند و فرکانس کلمه به ترتیب ۵۶٪ و ۸۴٪ و نسبت به روش LSB نیز به ترتیب ۲۰٪ و ۲۴٪ بوده است.

روش LSB که سعی در ایجاد توازن بین بار کاری سرورها دارد نیز عملکرد خوبی از خود نشان داده است. لیکن سه نقطه ضعف عمده در این روش وجود دارد که منجر به تولید نتایجی ضعیف‌تر نسبت به روش پیشنهادی شده است: ۱) وزن یک کلمه صرفاً بر حسب احتمال رخداد آن کلمه در سابقه پرس‌وجوها بیان می‌شود و به تعداد اسناد حاوی یک کلمه توجهی نمی‌شود، اما یک کلمه هرچند در تعداد پرس‌وجوهای زیادی بیاید، اگر در تعداد کمی سند موجود باشد نمی‌تواند بار کاری زیادی بر روی سرورها ایجاد نماید، ۲) در تابع وزن یک کلمه توجهی به انحراف از میانگین تکرار کلمات در سابقه پرس‌وجوها نمی‌شود و ۳) اسناد طولانی به دلیل برخورداری از تعداد بیشتری از کلمات پرتکرار، وزن بیش از حدی پیدا می‌کنند در حالی که معمولاً این سندها ارتباط کمتری با پرس‌وجوها پیدا می‌کنند و کمتر در نتایج پرس‌وجو ظاهر می‌شوند.

جالب است که برتری روش ما در ساعات اوج ورود پرس‌وجوها که حوالی ظهر می‌باشد، بیشتر خود را نشان می‌دهد. عمده دلیل این بهبود را می‌توان در وزن‌دهی مناسب به کلمات با توجه به ویژگی‌های مختلف آنها، نظیر میانگین تکرار و انحراف از میانگین یک کلمه در سابقه پرس‌وجوها خلاصه نمود. در کنار این مورد می‌توان به اصلاحات اعمال‌شده در تابع وزن‌دهی به اسناد (لحاظ‌کردن طول سند) و سیاست توزیع اسناد (لحاظ‌کردن اسناد با طول صفر) اشاره کرد.

۵- نتیجه‌گیری

در این مقاله، روش جدیدی برای حل مسئله توزیع اسناد بین سرورهای موتور جستجو معرفی گردید. روش پیشنهادی بر بهبود توازن بار پردازشی بین سرورها مخصوصاً در زمان اوج ترافیک و ورود پرس‌وجوهای سنگین تمرکز دارد. پرس‌وجوهای سنگین پرس‌وجوهایی هستند که به خاطر مطابقت با تعداد زیادی سند، حجم محاسبات بیشتری برای تولید پاسخ لازم دارند و در صورت عدم رسیدگی مناسب، منجر به تشکیل گلوگاه در موتور جستجو می‌شوند.

در این پژوهش، ما از سابقه پرس‌وجوی کاربران برای شناخت روند جستجوی کلمات استفاده می‌کنیم. با شناخت این روند و استخراج کلمات پرتکرار، به هر یک از کلمات و متعاقباً به هر یک از اسناد، وزنی تخصیص داده می‌شود که مقدار این وزن رابطه مستقیمی با احتمال بازگردانده‌شدن آن سند در پاسخ به پرس‌وجوهای ارسالی به موتور دارد. برای وزن‌گذاری یک کلمه، از پارامترهای مختلفی نظیر میانگین و انحراف از میانگین تعداد تکرار کلمه در دادگان پرس‌وجوهای آموزش و همین طور فرکانس سند کلمه بهره می‌گیریم. در نهایت، اسناد را با توجه به وزن به دست آمده، به صورت عادلانه بین سرورها توزیع می‌نماییم. نتایج ارزیابی بر روی داده واقعی یک موتور جستجو نشان می‌دهد که روش پیشنهادی توانسته است توازن بار بین سرورها را به ویژه در زمان اوج ورود پرس‌وجوها بیش از ۲۰٪ در مقایسه با روش‌های دیگر بهبود بخشد.

مراجع

[1] B. B. Cambazoglu, E. Kayaaslan, S. Jonassen, and C. Aykanat, "A term-based inverted index partitioning model for efficient distributed query processing," *ACM Trans. Web*, vol. 7, no. 3, pp. 15-23, Sept. 2013.

[2] Y. C. Ma, C. P. Chung, and T. F. Chen, "Load and storage balanced posting file partitioning for parallel information retrieval," *J. Syst. Softw.*, vol. 84, no. 5, pp. 864-884, May 2011.

[3] A. Kulkarni, A. S. Tigelaar, D. Hiemstra, and J. Callan, "Shard ranking and cut off estimation for topically partitioned collections," in *Proc. of the 21st ACM Int. Conf. on Information and Knowledge Management*, pp. 555-564, Hawaii, USA, 1-2 Nov. 2012.

[4] A. Moffat, W. Webber, J. Zobel, and R. Baeza-Yates, "A pipelined architecture for distributed text query evaluation," *Inf. Retr. Boston.*, vol. 10, no. 3, pp. 205-231, Jun. 2007.

[5] A. Kulkarni, J. Teevan, K. M. Svore, and S. T. Dumais, "Understanding temporal query dynamics," in *Proc. of the 4th ACM Int. Conf. on Web Search and Data Mining, WSDM'11*, pp. 167-176, Hong Kong, China, 9-12 Feb. 2011.

[6] K. M. Risvik and R. Michelsen, "Search engines and web dynamics," *Comput. Networks*, vol. 39, no. 3, pp. 289-302, Jun. 2002.

[7] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer Networks ISDN Syst.*, vol. 30, no. 1-7, pp. 107-117, Apr. 1998.

[8] A. Moffat, W. Webber, and J. Zobel, "Load balancing for term-distributed parallel retrieval," in *Proc. of the 29th Annual ACM SIGIR Int. Conf. on Research and Development in Information Retrieval*, pp. 348-355, Seattle, USA, 6-11Aug. 2006.

[9] D. Puppin, F. Silvestri, R. Perego, and R. Baeza-Yates, "Tuning the capacity of search engines: load-driven routing and incremental caching to reduce and balance the load," *ACM Trans. Inf. Syst.*, vol. 28, no. 2, pp. 1-36, May 2010.

[10] C. Lucchese, S. Orlando, R. Perego, and F. Silvestri, "Mining query logs to optimize index partitioning in parallel web search engines," in *Proc. of the 2nd Int. Conf. on Scalable Information Systems*, pp. 43-52, Suzhou, China, 6-8 Jun. 2007.

[11] A. Kulkarni and J. Callan, "Selective search: efficient and effective search of large textual collections," *ACM Trans. Inf. Syst.*, vol. 33, no. 4, pp. 1-33, Apr. 2015.

[12] A. Kulkarni, A. S. Tigelaar, D. Hiemstra, and J. Callan, "Shard ranking and cutoff estimation for topically partitioned collections," in *Proc. of the 21st ACM International Conf. on Information and Knowledge Management*, pp. 555-564, Hawaii, USA, 1-2 Nov. 2012.

[13] Y. Kim, J. Callan, J. S. Culpepper, and A. Moffat, "Load-balancing in distributed selective search," in *Proc. of the 39th ACM SIGIR Int. Conf. on Research and Development in Information Retrieval*, pp. 905-908, Pisa, Italy, 17-21 Jul. 2016.

[14] Z. Dai, C. Xiong, and J. Callan, "Query-biased partitioning for selective search," in *Proc. of the 25th ACM International Conf. on Information and Knowledge Management*, pp. 1119-1128, New York, USA, 24-28 Oct. 2016.

[15] Y. Kim, J. Callan, J. S. Culpepper, and A. Moffat, "Efficient distributed selective search," *Information Retrieval J.*, vol. 20, no. 3, pp. 221-252, Jun. 2017.

[16] A. Kane and F. W. Tompa, "Small-term distribution for disk-based search," in *Proc. of the ACM Symposium on Document Engineering*, New York, USA, 4-7 Sept. 2017.

[17] S. Jonassen and S. E. Bratsberg, "Impact of the query model and system settings on performance of distributed inverted indexes," in *Proc. of the 22nd Norwegian Informatics Conf., NIK'09*, pp. 143-154, Oslo, Norway, 25-27 Nov. 2009.

[18] B. Ribeiro-Neto, E. S. Moura, M. S. Neubert, and N. Ziviani, "Efficient distributed algorithms to build inverted files," in *Proc. of the 22nd Annual International ACM SIGIR Conf. on Research and Development in Information Retrieval*, pp. 105-112, Berkeley, USA, 15-19 Aug. 1999.

[19] Z. Dai, C. Xiong, and J. Callan, "Query-biased partitioning for selective search," in *Proc. of the 25th ACM Int. on Conf. on Information and Knowledge Management*, pp. 1119-1128, New York, USA, 24-28 Oct. 2016.

[20] H. Patel, Inverted Index Partitioning Strategies for a Distributed Search Engine, University of Waterloo, 2010.

[21] Z. Gyongyi and H. Garcia-Molina, *Web Spam Taxonomy*, Stanford University, 2005.

[22] J. Zhou and T. Yang, "Selective early request termination for busy internet services," in *Proc. of the 15th Int. Conf. on World Wide Web-WWW'06*, pp. 605-614, Edinburgh, Scotland, 23-26 May 2006.

[23] K. Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval," *J. Doc.*, vol. 28, no. 1, pp. 11-21, Jan. 1972.

[24] Apache Lucene, *Welcome to Apache Lucene*, Apache Foundation, 2018. [Online]. Available: http://lucene.apache.org/.

سیده ریحانه تراب جهرمی مدرک کارشناسی و کارشناسی ارشد خود را در رشته مهندسی کامپیوتر به‌ترتیب در سال‌های ۱۳۹۳ و ۱۳۹۶ از دانشگاه باهنر کرمان و دانشگاه یزد دریافت نموده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: سیستم عامل، وب‌کاوی، سرویس‌های وب و شبکه‌های کامپیوتری.

سجاد ظریف‌زاده مدرک کارشناسی و کارشناسی ارشد خود را به‌ترتیب در سال ۱۳۸۱ و ۱۳۸۴ در رشته مهندسی کامپیوتر از دانشگاه تهران دریافت نمود. در سال ۱۳۹۱، وی موفق به اخذ درجه دکتری مهندسی کامپیوتر از همین دانشگاه شد. نام‌برده دوره فرصت مطالعاتی خود را در سال ۱۳۹۰ در دانشگاه جرجیاتک آمریکا گذراند. دکتر ظریف‌زاده از سال ۱۳۹۲ در گروه مهندسی کامپیوتر دانشگاه یزد مشغول به فعالیت گردید و اینک نیز عضو هیأت علمی این دانشگاه می‌باشد. زمینه‌های علمی مورد علاقه وی شامل سرویس‌ها و کاربردهای اینترنتی، وب‌کاوی، طراحی سیستم، تحلیل داده و حجیم داده می‌باشد.