

صلى الله عليه وسلم



دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

پروژه‌ی کارشناسی
مهندسی کامپیوتر - نرم‌افزار

عنوان:

پیاده‌سازی خزنده‌ی موتور جستجوی هوشمند کسب‌وکار فارسی

نگارش:

بهنام حاتمی ورزنده

استاد راهنما:

دکتر حمید بیگی

شهریورماه ۱۳۹۲



دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

پروژه‌ی کارشناسی
مهندسی کامپیوتر - نرم‌افزار

عنوان:

پیاده‌سازی خزنده‌ی موتور جستجوی هوشمند کسب‌وکار فارسی

نگارش:

بهنام حاتمی ورزنه

استاد راهنما:

دکتر حمید بیگی

نمre:

امضای استاد راهنما:

امضای استاد ممتحن:

چکیده

فواید بسیار استفاده از اینترنت، آن را به یکی از پرتعدادترین روش‌های برقراری ارتباط و تبادل اطلاعات بدل نموده است. با افزایش روز افزون حجم دانش ذخیره شده در این شبکه‌ی جهانی، فرآیند دسترسی به بخش خاصی از مطالب که مدنظر کاربر است تبدیل به کاری دشوار شده است. بنابراین نیاز به ابزار جدید برای ساماندهی دسترسی و جستجو در این حجم انبوه از اطلاعات بیش از پیش حس می‌گردد. به همین منظور به مرور زمان، ابزارهای عامی برای جستجو در بین اطلاعات اینترنت، طراحی و پیاده‌سازی گردید. اما این ابزارهای کلی، نیازهای خاص کاربران را به طور کامل برطرف نمی‌کرد. یکی از نیازهای خاص کاربران، مقوله‌ی کارایی است. در حال حاضر برای زبان فارسی موتور جستجوی کارآمدی برای پیدا کردن کسب و کار مناسب برای متقاضیان کار وجود ندارد.

در این پروژه، سعی بر آن شد که با استفاده از سایت‌هایی که در زمینه‌ی کارایی فعالیت دارند، یک سیستم یک‌پارچه‌ی جستجو پیاده‌سازی شود. برای این کار، نیاز به استخراج آگهی‌های کارفرمایان و کارنامک‌های متقاضیان از صفحات وب احساس می‌گردید. در این پروژه به بررسی سایت LinkedIn که یکی از موفق‌ترین سایت‌های کارایی انگلیسی زبان می‌باشد پرداخته شد و شمای داده‌ای کارنامک آن، استخراج گردید. علاوه بر آن، یک خزنده و یک موتور جستجوی متن باز، به گونه‌ای بهینه‌سازی گردید که امکان استخراج و شاخص‌بندی آگهی‌های کارایی از سایت‌های فعال در این زمینه به صورت کارایی ممکن گردید. با این کار، دقت جستجو در بین آگهی‌ها بهبود یافت و اطلاعات غیرکاربردی صفحات وب، دور ریخته شد و حجم داده‌های ذخیره شده در شاخص‌ها، به میزان محسوسی کاهش پیدا کرد.

کلمات کلیدی: آگهی کسب و کار، موتورهای جست‌وجو، خزنده‌ی Nutch، شاخص‌بند Lucene، داده‌کاوی.

فهرست مطالب

۹	۱ پیش‌گفتار
۱۱	۲ معرفی مسئله
۱۱	۱-۲ تعریف دقیق مسئله
۱۲	۲-۲ کارهای مشابه
۱۴	۳ موتورهای جستجو
۱۴	۱-۳ موتور جستجوی وب
۱۵	۲-۳ انواع موتورهای جستجو
۱۵	۱.۲-۳ موتورهای جستجو مبتنی بر خزنده‌ها
۱۶	۲.۲-۳ موتورهای جستجو مبتنی بر انسان
۱۶	۳-۳ ساختار و نحوه‌ی کار موتورهای جستجو
۱۶	۱.۳-۳ جمع‌آوری اطلاعات یا خزش
۱۶	۲.۳-۳ نگهداری پایگاه‌داده یا مخزن
۱۷	۳.۳-۳ شاخص‌بندی
۱۷	۴.۳-۳ پرسمان
۱۷	۵.۳-۳ رتبه‌دهی
۱۸	۴-۳ نمونه‌ی موتورهای جستجو
۱۸	۵-۳ خلاصه‌ی فصل
۲۱	۴ موتور جستجوی Nutch
۲۱	۱-۴ مقدمه
۲۲	۲-۴ نحوه‌ی عملکرد

۲۲	عملکرد کلی	۱.۲-۴
۲۳	ساختار افزونه	۲.۲-۴
۲۳	ساختار Lucene	۳-۴
۲۳	عملکرد کلی	۱.۳-۴
۲۴	شاخص بندی	۲.۳-۴
۲۵	تحلیل گر	۳.۳-۴
۲۵	خلاصه	۴-۴

۲۶	پایه سازی	۵
۲۶	مقدمه	۱-۵
۲۶	پیش نیازها	۲-۵
۲۷	مراحل پایه سازی	۳-۵
۲۷	بررسی و انتخاب سایت های هدف	۱.۳-۵
۲۹	بررسی سایت LinkedIn	۲.۳-۵
۲۹	آماده سازی Nutch	۳.۳-۵
۳۰	پایه سازی افزونه های تجزیه کننده	۴.۳-۵
۳۱	پایه سازی افزونه ی شاخص بندی	۵.۳-۵
۳۱	خلاصه	۴-۵

۳۵	نتایج	۶
۳۵	مقدمه	۱-۶
۳۵	محیط اجرای برنامه	۲-۶
۳۶	روش به کار رفته	۳-۶
۳۶	نتایج و بحث	۴-۶
۳۶	تعداد واژه ها	۱.۴-۶
۳۷	زمان بازیابی اطلاعات	۲.۴-۶
۳۷	نتایج جستجو	۳.۴-۶
۳۸	خلاصه	۵-۶

۳۹	نتیجه گیری	۷
۳۹	خلاصه	۱-۷
۳۹	کارهای آینده	۲-۷

فهرست شکل‌ها

۱۹	۱-۳ ساختار و نحوه‌ی کار یک موتور جستجو [۱]
۲۰	۲-۳ نحوه‌ی کار خزنده [۲]
۲۳	۱-۴ ساختار خزنده‌ی Nutch
۲۴	۲-۴ ساختار Lucene [۳]
۲۷	۱-۵ مثالی از الگوی کارخانه [۱۰]
۳۲	۲-۵ شمای داده‌ای سایت LinkedIn
۳۳	۳-۵ ساختار افزونه‌ی تجزیه‌کننده
۳۴	۴-۵ ساختار افزونه‌ی شاخص‌بند

فهرست جدول‌ها

۱-۵	لیست سایت‌های فعال کاریابی به زبان فارسی	۲۸
۱-۶	پرسமான‌های به کار رفته برای بررسی دقت جستجو	۳۶
۲-۶	تعداد کلیدواژه‌های ذخیره شده بر اساس سایت	۳۷
۳-۶	مدت زمان بازیابی بر اساس سایت	۳۷
۴-۶	میزان دقت جستجو به ازای هر سایت	۳۷

فصل ۱

پیش‌گفتار

با افزایش روز افزون حجم دانش ذخیره شده به صورت دیجیتالی و در قالب‌های مختلف نظیر اخبار، صفحات وب، صفحات شخصی، مقالات علمی، کتاب‌ها، تصاویر، فایل‌های صوتی و تصویری و شبکه‌های اجتماعی، فرآیند جستجو به دنبال بخش خاصی از مطالب که مدنظر است و یافتن آن، تبدیل به کاری دشوار شده است. بنابراین نیاز به داشتن ابزار محاسباتی جدید که امکان سازماندهی، جستجو و فهم این حجم انبوه از اطلاعات را بدهد، بیش از پیش حس می‌شود.

در حال حاضر، در مورد داده‌هایی که به صورت برخط ذخیره شده‌اند می‌توان از دو طریق جستجو و پیوند^۱ صفحه، به مطالب مورد نظر دست یافت. به این صورت که می‌توان مطلب مورد نظر را در قالب واژگان کلیدی در یک موتور جستجو وارد کرد و در پاسخ به آن، مجموعه‌ای از اسناد مرتبط با عبارت جستجو را دریافت کرد. اما بعضاً ممکن است شخص جستجوکننده به جای جستجو، به دنبال یک سند خاص یا به دنبال مطالب در یک زمینه‌ی موضوعی خاص و ارتباط آن‌ها با یکدیگر باشد. در این صورت لازم است تا شخص جستجوکننده قبل از جستجو با استفاده از واژگان کلیدی، ابتدا زمینه را پیدا کرده و مطالب مرتبط با آن را مطالعه کند. این زمینه و اسناد مرتبط با آن ممکن است در گذر زمان نیز تغییر کنند. بنابراین استفاده از ساختار معنایی اسناد و طبقه‌بندی آن‌ها با استفاده از این ساختار، روشی دیگر برای کاوش و استفاده از اسناد است.

در بسیاری از مجموعه‌های اسناد، به دلیل حجم بالای مطالب، نمی‌توان به طور کامل از قوای انسانی برای خواندن همه‌ی اسناد و پیدا کردن ساختار معنایی آن‌ها و جستجو به دنبال سایر اسناد مرتبط با استفاده از واژگان کلیدی استفاده کرد. به همین منظور روش مدل‌سازی موضوع^۲ که مبتنی بر پردازش زبان طبیعی با استفاده از یادگیری ماشین است، به همراه جمع‌آوری و استخراج خودکار اطلاعات معرفی شده است.

روش مدل‌سازی موضوع، یک مدل آماری برای یافتن عناوین استفاده شده در یک مجموعه با حجم بالا از اسناد، با

^۱link

^۲topic modeling

استفاده از اطلاعات معنایی و ساختار معنایی نهان اسناد است. فرض اصلی روش‌های مدل‌سازی عناوین، تشکیل شدن هر سند از تعداد اندکی از عناوین است که در آن هر عنوان، دارای توزیع مشخص و مرتبط با موضوع از کلمات است. بنابراین کلماتی که در توزیع احتمال مربوط به هر عنوان به کار رفته در سند، با احتمال بالایی حضور داشته باشند، با احتمال بالایی جزء کلمات تشکیل‌دهنده‌ی سند نیز می‌باشند. بنابراین با استفاده از روش‌های آماری و به صورت مشابه، این الگوریتم‌ها، کلمات استفاده شده در متن را به منظور یافتن زمینه‌های معنایی اصلی به کار رفته در متن و همچنین یافتن ارتباط این زمینه‌ها و تغییرات آن در گذر زمان، بررسی می‌کنند.

فصل ۲

معرفی مسئله

۱-۲ تعریف دقیق مسئله

هدف از انجام این پژوهش، پیاده‌سازی موتور جستجوی هوشمند کسب و کار فارسی است. روزانه حجم بالایی از آگهی‌های استخدام، در فضای برخط و در قالب صفحات وب و یا صفحات شخصی افراد، منتشر می‌شوند. از طرفی تعداد این صفحات بسیار زیاد است و به روز رسانی صفحات معمولاً از سرعت بالایی (تقریباً هر روز) برخوردار است. از طرفی دیگر، اغلب این صفحات، آگهی‌هایی در همه‌ی زمینه‌های موضوعی و شغلی و همچنین شرایط مکانی نظیر شهر و استان محل کار را پوشش می‌دهند.

در حال حاضر، در چند مورد از سایت‌های فارسی که در زمینه‌ی استخدام فعالیت می‌کنند، امکان دسته‌بندی مطالب بر حسب نوع آگهی وجود دارد، اما این دسته‌بندی توسط انسان و بدون استفاده از روش‌های یادگیری انجام می‌شود و در بسیاری از موارد متأسفانه دسته‌بندی موجود، چندان کامل نیست. همچنین امکان جستجو اغلب به صورت جستجوی متنی در این سایت‌ها وجود دارد و امکان جستجو با توجه به مواردی همچون جنسیت فرد، نوع شغل و موقعیت مکانی آن وجود ندارد. همچنین تعامل آن‌ها با افراد با استفاده از روش‌هایی مانند عضویت و یا ارسال نظر و در مواردی اندک، ارسال رزومه است. اما در روش‌های تعاملی و گزینش خبرهای مرتبط با افراد نیز متأسفانه از روش‌های هوشمند استفاده نمی‌شود و این کار با استفاده از نیروی انسانی صورت می‌گیرد.

با توجه به ویژگی‌های مطرح شده برای این صفحات وب، مشاهده و جستجوی روزانه در میان حجم انبوه اخبار و آگهی‌ها، بدون استفاده از روش‌های هوش مصنوعی و تنها با استفاده از نیروی انسانی هم برای یافتن افراد متناسب با شغل و گزینش با توجه به توانایی آن‌ها و هم برای فرد متقاضی، کاری بسیار دشوار است. بنابراین می‌توان از الگوریتم‌های یادگیری در قسمت دریافت اخبار و پیمایش صفحات وب و همچنین تعامل با متقاضی و همچنین دسته‌بندی آگهی‌ها و اخبار استفاده کرد.

در این پژوهش، با استفاده از تکنولوژی‌های موجود برای بازایی و اختصاصی‌سازی آن‌ها، از سایت‌هایی که در

زمینه‌ی استخدام فعالیت دارند، آگهی‌های استخدام استخراج می‌شود و با پردازش هرکدام، اطلاعات تخصصی مورد نیاز جداسازی شده و سپس شاخص‌بندی و برای اجرای انواع پرس‌مان‌ها توسط تکنولوژی‌های موجود آماده می‌گردد. در این پژوهش، برای بازیابی اطلاعات از نرم‌افزار متن‌باز^۱ Nutch، استفاده می‌شود. این نرم‌افزار، دارای امکانات و ویژگی‌های خاص خود می‌باشد و امکان پیکربندی بالایی دارد. سپس اطلاعات بازیابی شده را به وسیله‌ی Lucene^۲، شاخص‌بندی می‌کنیم. این نرم‌افزار یکی از بهترین نرم‌افزارهای متن‌باز در این زمینه است. سپس با استفاده از Solr^۳، امکان پرس‌مان بر روی اطلاعات استخراج شده فراهم خواهیم کرد. سپس اطلاعات به دست آمده به پژوهش مکمل برای پردازش‌های بعدی داده می‌شود.

در پژوهش مکمل، از الگوریتم‌های یادگیری برای هوشمند کردن دسته‌بندی آگهی‌ها و اخبار استفاده می‌شود. این سامانه‌ی هوشمند، از اطلاعات پیمایش‌شده‌ی صفحات وب استفاده می‌کند، بنابراین ورودی مسئله تعدادی از آگهی‌های فارسی است. هدف دسته‌بندی آگهی‌ها بر اساس موضوع آن‌هاست، به گونه‌ای که هر آگهی بتواند در یک یا چند دسته با موضوع مرتبط با خود قرار بگیرد. این مسئله همانند مسئله‌ی مدل‌سازی عناوین است. به این صورت که تعدادی سند (در قالب آگهی) در اختیار داشته و هدف نهایی قرار دادن این اسناد در یک یا چند دسته و بدست آوردن این دسته‌هاست. بنابراین از دو الگوریتم LDA و PLSA که در ادامه شرح داده خواهد شد، برای حل این مسئله استفاده می‌شود. البته باید توجه کرد که در مدل‌سازی عناوین، تاپیک‌ها به صورت هوشمند نام‌گذاری نمی‌شوند. بنابراین نام‌گذاری مناسب دسته‌ها جزئی از راه حل مسئله محسوب می‌شود. در نهایت خروجی این مسئله، تعدادی موضوع با عناوینی همچون «استخدام بانک‌ها»، «استخدام نیروی انتظامی» و یا به تفکیک مکانی مانند «استخدام استان تهران» و «استخدام استان اصفهان» و همچنین اسناد مرتبط با هر یک از موضوعات می‌باشد.

۲-۲ کارهای مشابه

در زمینه‌ی کسب‌وکار هوشمند آنلاین، در زبان‌های دیگر کارهای مشابهی انجام شده است که از جمله‌ی آن‌ها می‌توان به صفحه‌ی وب Texkernel^۴ اشاره کرد. این سایت از ۶ قسمت اصلی تشکیل شده است که به صورت مجتمع در کنار یکدیگر قرار گرفته‌اند و از هر یک از این سرویس‌ها می‌توان به صورت جداگانه استفاده کرد. در زیر به اختصار به هر یک از این سرویس‌ها و ویژگی‌های آن‌ها اشاره می‌کنیم:

– قسمت استخراج که قسمت‌های مختلف رزومه را به صورت خودکار از روی کارنامک^۵ و یا صفحه‌ی کاربر در رسانه‌های اجتماعی و امکان تکمیل پروفایل کاربر به صورت اتوماتیک استخراج می‌کند.

^۱ Open source

^۲ nutch.apache.org

^۳ lucene.apache.org

^۴ lucene.apache.org/solr

^۵ www.texkernel.com

^۶ Curriculum vitae (CV)

- قسمت منابع که به صورت اتوماتیک کارنامک و اطلاعات فرد در شبکه‌های اجتماعی را جدا کرده و به صورت گرافیکی در کنار رزومه‌ی اصلی فرد قرار می‌دهد و به کاربر امکان ویرایش و اضافه یا حذف اطلاعات از کارنامک خود در پایگاه‌داده‌ی سایت را می‌دهد. پس از این مرحله اطلاعات فرد در پایگاه‌داده‌ی صفحه ذخیره می‌شود تا در مراحل بعدی مورد استفاده قرار گیرد.
 - قسمت جستجو امکان جستجو در میان رزومه‌های موجود در پایگاه‌داده برای یافتن افراد مرتبط با هر شغل و رتبه‌بندی آن‌ها را می‌دهد.
 - قسمت خوراک شغل^۷ که به صورت خودکار به صورت روزانه در سایت‌های کسب‌وکار جستجو می‌کند و آگهی‌های جدید را پردازش کرده و قسمت‌های مورد نیاز را از آن استخراج می‌کند.
 - قسمت وصل کردن که متن آگهی کار را دریافت کرده و به صورت خودکار، افراد متناسب با آن شغل بر روی پایگاه‌داده‌ها جستجو و به صورت فهرست بدست می‌آیند.
 - قسمت برداشت که به صورت خودکار، شغل‌های متناسب با توانایی و شرایط کاربر که بر روی خوراک شغل قرار دارد را به او نشان می‌دهد.
- هر یک از این بخش‌ها به صورت جداگانه قابل دسترسی و استفاده در صفحه مورد نظر هستند. اما متأسفانه هیچ یک از این بخش‌ها از زبان فارسی پشتیبانی نمی‌کند.
- کار انجام شده در این پژوهش مشابه بخش خوراک شغل است و اطلاعات مورد نیاز را از آگهی‌های فارسی استخراج می‌کند.
- از ویژگی‌های اصلی قسمت خوراک شغل سایت textkernel می‌توان به موارد زیر اشاره کرد:
- مقایسه هر آگهی با آگهی‌های دریافت شده در ۶ ماه اخیر و تشخیص شغل‌های یکتا و رفتار کارفرماها
 - به روزرسانی و بررسی وضعیت شغل‌ها از نظر باز یا بسته بودن و همچنین ظرفیت باقیمانده از شغل به صورت روزانه
 - داشتن پیوند به صفحه‌ی فرد در شبکه‌ی اجتماعی^۸ LinkedIn
- قابل ذکر است، که نوع کارنامک فرد، باید ساختاری مشابه ساختار LinkedIn باشد، به همین علت، بررسی دقیق LinkedIn برای این پروژه نیاز است.

^۷ JobFeed

^۸ www.linkedin.com

فصل ۳

موتورهای جستجو

با توجه به آمار جهانی اینترنت [۴]، در تاریخ ۳۱م مارچ ۲۰۰۸، ۱/۴۰۷ میلیارد انسان، از اینترنت استفاده می‌کردند. میزان نفوذ اینترنت به طور روز افزون در حال افزایش است. شبکه‌ی جهانی گسترده وب^۱ (که معمولاً به اختصار وب نامیده می‌شود) [۵]، یک سیستم از اسناد ابرمتن^۲ به هم متصل است که به وسیله‌ی اینترنت قابل دسترسی هستند. با استفاده از یک مرورگر، کاربر امکان مشاهده‌ی صفحات وب که دارای محتوای داده‌ای، عکس، فیلم و سایر امکانات چند رسانه‌ای را دارد و می‌تواند به وسیله‌ی لینک‌ها، بین آن‌ها جابه‌جا گردد.

همان گونه که تعداد صفحات وب، به طور روزافزون در حال افزایش است، نیاز به موتور جستجو بیشتر احساس می‌گردد. در این فصل، توضیح مختصری در مورد المان‌های پایه‌ی هر سیستم جستجویی به همراه نحوه‌ی عملکرد آن المان، مورد بررسی قرار گرفته است. سپس، نقش خزنده‌های وب^۳، که یکی از اصلی‌ترین بخش‌های اصلی هر سیستم جستجوی اینترنتی می‌باشد، مورد بررسی گرفته است.

۳-۱ موتور جستجوی وب

محتوای بسیاری از شبکه‌ی جهانی گسترده‌ی وب، قابل استفاده برای میلیون‌ها نفر است. بسیاری از افراد، دسترسی به صفحات وب را از نقاط آغازی مانند، Yahoo^۴ و MSN^۵ و... آغاز می‌نمایند. اما بسیاری از افراد نیازمند اطلاعات [۶]، برای شروع فعالیت اینترنتی خود از موتورهای جستجو شروع می‌کنند. در این حالت، کاربر یک پرسمان^۶ ارسال

^۱ World Wide Web

^۲ Hyper text documents

^۳ Web crawlers

^۴ www.yahoo.com

^۵ www.msn.com

^۶ Query

می‌نماید، که معمولاً به صورت لیستی از کلیدواژه‌ها^۷ است و در پاسخ، لیستی از صفحات وب که احتمالاً مرتبط با درخواست کاربر بوده (معمولاً صفحاتی که در آن کلیدواژه‌ها به کار رفته باشد) را دریافت می‌کند. در زمینه‌ی وب، موتورهای جستجو، در واقع به جستجوگرهایی گفته می‌شود، که در یک پایگاه داده^۸ از فایل‌های وب، جستجوی خود را انجام می‌دهد.

۲-۳ انواع موتورهای جستجو

به طور کلی، سه نوع موتور جستجو وجود دارد [۵]:

– موتورهای جستجویی که به وسیله‌ی ربات‌ها اجرا می‌شوند (معمولاً به خزنده‌ها، مورچه‌ها^۹ یا عنکبوت‌ها^{۱۰} معروفند).

– موتورهای جستجویی که بر اساس ارسال‌های کاربران عمل می‌کنند.

– موتورهای جستجویی که بر اساس تلفیق دو نوع بالا عمل می‌کنند.

دو نوع اصلی موتورهای جستجو در زیر به اختصار توضیح داده شده است:

۱.۲-۳ موتورهای جستجو مبتنی بر خزنده‌ها

چنین موتورهای جستجویی [۵]، از تعدادی عامل^{۱۱} نرم‌افزاری خودکار (که خزنده نامیده می‌شود) تشکیل شده است. این خزنده‌ها، صفحات وب را دریافت، اطلاعات و ابر تگ‌های^{۱۲} آن را استخراج می‌کنند. همچنین برای دسترسی به تمام صفحات یک وب سایت و شاخص‌بندی^{۱۳} آن‌ها، لینک‌های داخل صفحات را دنبال می‌کند. خزنده، تمام اطلاعات استخراج شده را، در یک مخزن مرکزی^{۱۴} ذخیره می‌نماید. سپس داده‌ها، در مخزن شاخص‌بندی می‌گردد. خزنده همچنین به طور متناوب به صفحات بازبایی شده مراجعه می‌نماید و در صورت تغییر صفحات وب، اطلاعات خود را به روز رسانی می‌نماید. تناوب چنین کاری توسط مدیر سیستم، تنظیم می‌گردد.

^۷Keywords

^۸Database

^۹Ants

^{۱۰}Spiders

^{۱۱}Agents

^{۱۲}Metatags

^{۱۳}Indexing

^{۱۴}Central Repository

۲.۲-۳ موتورهای جستجو مبتنی بر انسان

چنین موتورهای جستجویی [۵]، مبتنی است بر داده‌هایی که به مرور زمان به وسیله‌ی انسان، به سیستم ارسال می‌شود. این اطلاعات شاخص‌بندی می‌گردد و دسته‌بندی^{۱۵} می‌گردد. در این نوع موتورهای جستجو، تنها داده‌هایی که ارسال شده است، در شاخص‌ها ذخیره می‌شوند. چنین موتورهای جستجویی، به ندرت در مقیاس بزرگ مورد استفاده قرار می‌گیرد، اما در سازمان‌هایی که با داده‌های با مقیاس کوچک روبرو هستند، بسیار پر استفاده است.

۳-۳ ساختار و نحوه‌ی کار موتورهای جستجو

ساختار پایه‌ی هر موتور جستجویی مبتنی بر خزنده، در شکل ۳-۱ نشان داده شده است. از این رو، فازهای اصلی هر موتور جستجویی عبارتند از:

۱.۳-۳ جمع‌آوری اطلاعات یا خزش

هر موتور جستجویی که بر پایه‌ی یک خزنده کار می‌کند، منابع اطلاعاتی خود را برای ارائه‌ی خدمات تأمین می‌کند. خزنده‌ها، نرم‌افزارهای کوچکی هستند که از طریق موتورهای جستجو به سایت‌ها سر می‌زنند، دقیقاً به همان روشی که انسان‌ها لینک‌های بین صفحات را دنبال می‌کنند. معمولاً در ابتدا، یک لیست ابتدایی از آدرس وب سایت‌ها به هر خزنده داده می‌شود. خزنده باید صفحه‌ی مربوط به هر کدام را دریافت نماید. پس از آن، لینک‌های داخل این صفحات بازیابی شده را استخراج نماید و اطلاعات استخراج شده را به واحد کنترل خزنده تحویل دهد. این واحد تصمیم می‌گیرد که چه لینک‌هایی در ادامه بازیابی گردد و لیست آن‌ها را برای خزنده ارسال می‌نماید. مراحل بیان شده را می‌توانید در شکل ۳-۲ ببینید.

۲.۳-۳ نگه‌داری پایگاه داده یا مخزن

همان طور که در شکل ۳-۱ می‌بینید، تمام داده‌های یک موتور جستجو، در یک پایگاه داده ذخیره می‌شود و تمام جستجوها و عملیات داده‌ای، به کمک این پایگاه داده انجام می‌پذیرد. این پایگاه داده نیاز دارد در طول زمان با توجه به تغییرهای بیرونی بروز رسانی گردد. در مرحله‌ی بازیابی و پس از اتمام مرحله‌ی دریافت اطلاعات به وسیله‌ی خزنده، موتور جستجو باید تمام اطلاعات جدید و مفید صفحات بازیابی شده را استخراج و در پایگاه داده ذخیره نماید. در بعضی از موتورهای جستجو، یک مخزن از صفحات ذخیره شده به صورت موقت بین این دو مرحله قرار می‌گیرد. حتی بعضی مواقع، موتورهای جستجو، یک حافظه‌ی سریع نهان^{۱۶} از صفحاتی که بازیابی شده‌اند، نگه می‌دارد تا

^{۱۵}Classified

^{۱۶}Cache

بتواند مرحله‌ی شاخص‌بندی را سریع‌تر انجام دهد و همچنین امکان جستجوی ابتدایی بر روی داده‌های دریافت شده را فراهم آورد.

۳-۳.۳ شاخص‌بندی

زمانی که صفحه‌ی بازیابی شده، در مخزن ذخیره می‌شود، کار بعدی موتور جستجو، ایجاد یک شاخص برای داده‌های ذخیره شده می‌باشد. واحد شاخص‌بندی، تمام کلمات را از هر صفحه استخراج می‌نماید و آدرس صفحه‌ی مدنظر را به ازای هر کلمه‌ی استخراج شده، ذخیره می‌نماید. نتیجه کار، معمولاً یک لغت‌نامه‌ی بزرگ می‌باشد که می‌تواند آدرس تمام صفحه‌هایی را که در آن‌ها کلمه‌ی خاصی آمده‌اند را به ما بدهد. به وضوح صفحات به صفحاتی محدود می‌شود که در فاز قبلی بازیابی شده‌اند. همان‌طور که قبلاً ذکر شده بود، شاخص‌بندی متن، مشکلات و چالش‌های خاص خودش را دارد. از جمله‌ی آن می‌توان به سایز بزرگ آن و سرعت زیاد تغییرات در آن اشاره نمود. همچنین علاوه بر چالش‌های فوق‌الذکر، جستجو برای شاخص‌های نادر و کمتر رایج نیز خود چالش‌زا است. به طور مثال، واحد شاخص‌بندی، می‌تواند یک شاخص ساختاری از اتصالات بین صفحات تولید نماید.

۳-۴.۳ پرسمان

این واحد با پرسمان‌های کاربر سروکار دارد. واحد پرسمان، مسئول دریافت و پاسخ‌گویی به درخواست‌های جستجو از طرف کاربران می‌باشد. این واحد به صورت اساسی وابسته به شاخص‌های موجود و بعضی مواقع به مخزن صفحات ذخیره می‌باشد. به علت حجم زیاد وب و وارد شدن عبارات جستجوی کوتاه به وسیله‌ی کاربران در حد یک یا دو کلیدواژه، مجموعه جواب موجود، بسیار زیاد می‌باشد.

۳-۵.۳ رتبه‌دهی

به علت اینکه مجموعه سندهای مرتبط با پرسمان وارد شده‌ی کاربر، بسیار زیاد است، یکی از مهم‌ترین وظایف موتورهای جستجو نمایش مرتبط‌ترین نتایج به کاربر است. برای اجرای کارآمد چنین امری، نتایج رتبه‌دهی می‌گردند. واحد رتبه‌دهی، به همین منظور وظیفه‌ی مرتب کردن نتایج را به گونه‌ای دارد که نتایج بالاتر احتمال بیشتری داشته باشند که همان اسنادی که کاربر به دنبال آن است باشند.

پس از پیدا کردن نتایج، به وسیله‌ی واحد رتبه‌دهی به هر یک از نتایج رتبه اختصاص داده شد، نتایج نهایی جستجو به کاربر نشان داده می‌شود. این روشی است که تقریباً تمام موتورهای جستجو مطابق با آن کار می‌کنند.

۴-۳ نمونه‌ی موتورهای جستجو

تعدادی موتور جستجو در حال حاضر قابل استفاده است. در زیر لیستی از مهم‌ترین و مشهورترین موتورهای جستجو آورده شده است:

– Google^{۱۷}

– Yahoo

– MSN

– E-Bay^{۱۸}

– AOL^{۱۹}

و تعداد بسیار زیادی موتور جستجوی دیگر در دسترس هست که کاربران را برای رسیدن به اطلاعات مدنظر یاری می‌نماید.

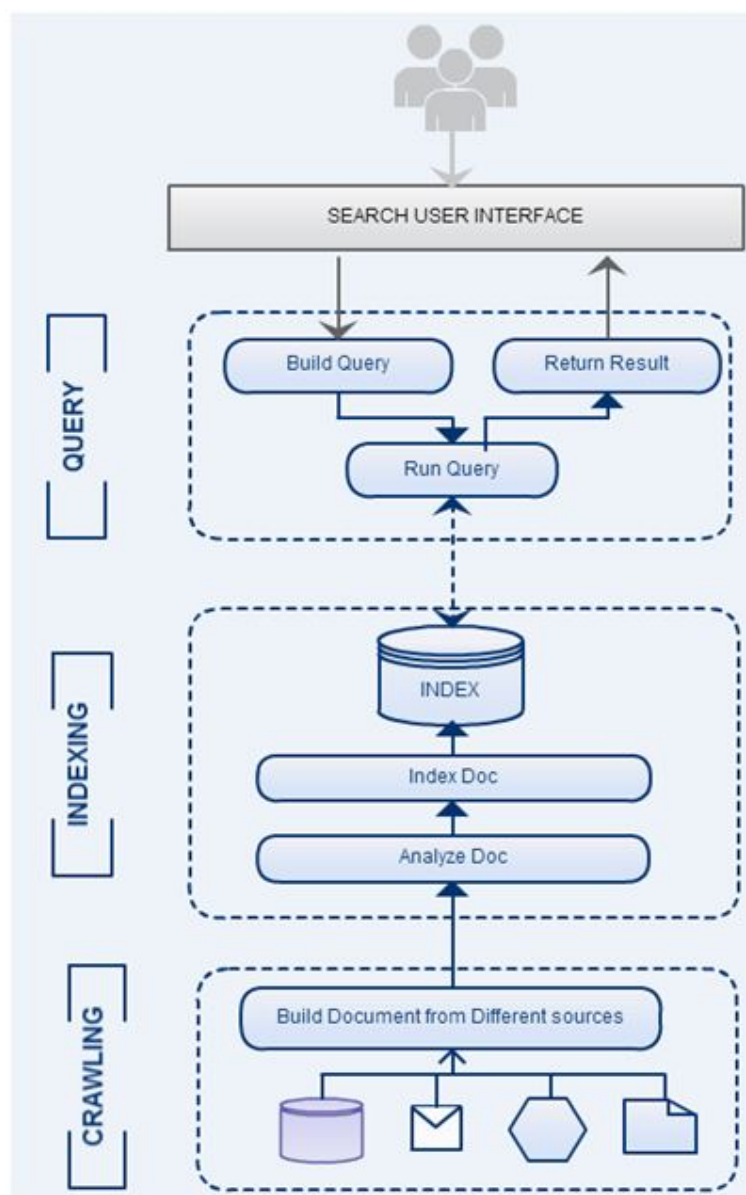
۵-۳ خلاصه‌ی فصل

موتورهای جستجو، به عنوان کلید اصلی ورود به جهان گسترده وب است. تکامل و اجزای موتورهای جستجو قسمتی مهمی از مطالعه‌ی جهان گسترده‌ی وب هستند. قسمت‌های ضروری موتور جستجو، عبارتند از خزنده، استخراج کننده، برنامه‌ریز و پایگاه داده. بعضی از مهم‌ترین موتورهای جستجوی پرکاربرد عبارتند از Google و MSN و

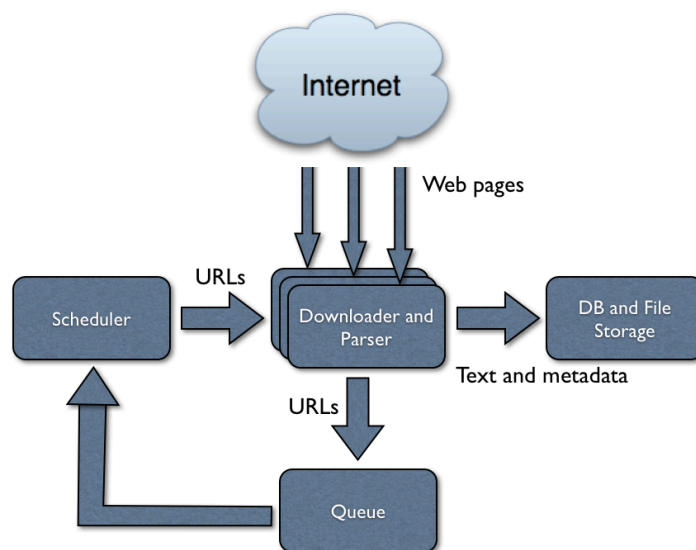
^{۱۷}www.google.com

^{۱۸}www.ebay.com

^{۱۹}www.aol.com



شکل ۳-۱: ساختار و نحوه‌ی کار یک موتور جستجو [۱].



شکل ۳-۲: نحوه‌ی کار خزنده [۲].

فصل ۴

موتور جستجوی Nutch

۴-۱ مقدمه

پروژه Nutch یک نرم افزار متن باز بر اساس زبان جاوا است که امکان خزش و جمع آوری طیف مختلفی از داده ها را، از یک شبکه ی داخلی، بخشی از اینترنت یا کل جهان گسترده ی وب را دارد [۷]. به طور کلی قبل از پیاده سازی Nutch امکان تحلیل نتایج موتورهای جستجوی معروف، در برابر پرسمانی دلخواه، وجود نداشت و نتیجه ی جستجوی آنها و نحوه ی رتبه بندی آنها، با معیاری مناسب و منصفانه قابل مقایسه نبود. یکی از دلایل این امر، وجود الگوریتم های جستجوی اختصاصی و منابع بسته^۱ در این شرکت ها بودند. البته دلیل چنین کاری، علاوه بر تمایل انحصار طلبی و رقابت، جلوگیری از سوء استفاده ی منتشرکنندگان هرزنامه ها، در بالا بردن رتبه ی یک دامنه ی خاص بود. پروژه Nutch با متن باز بودن خود، سعی در برطرف کردن این معضل نمود. یکی از اهداف این موتور جستجو، ایجاد شفافیت و افزودن جزئیات به نحوه ی رتبه بندی صفحات وب بود. علاوه بر آن، ارائه ی یک موتور جستجوی جایگزین، برای افرادی که به وسیله ی محدود موتورهای جستجوی تجاری موجود امروزی، راضی نشده اند و دارای نیازهای خاص منظره می باشند. همچنین ربات نظاره گر Nutch، این امکان را به مدیران سایت ها می دهد که قسمت هایی از سایتشان که به وسیله ی این روش جمع آوری می گردد را تحت مدیریت و کنترل خود داشته باشند.

معماری پروژه Nutch به گونه ای طراحی شده است که، هم از لحاظ حجم جستجو و هم از لحاظ سرعت، قابل گسترش و بهبود می باشد. به همین منظور، استفاده از روش های توازی در بازیابی اطلاعات، به عنوان یکی از روش های بهینه سازی از لحاظ سرعت در پیاده سازی لحاظ شده است [۷]. بخش های اصلی Nutch شامل سه قسمت اصلی واحد خزنده، واحد شاخص بندی و واسط جستجو بر روی داده ها است. خزنده Nutch به گونه ای طراحی شده که بر روی هر شبکه ی داخلی یا خارجی کار می کند. اطلاعات بازیابی شده توسط این قسمت، در یک پایگاه داده به نام WebDb برای استفاده های آتی ذخیره می شوند. خزنده، علاوه بر بازیابی و ذخیره سازی، با استفاده از نرم افزاری به

^۱Closed Source

نام Lucene برای شاخص‌بندی اطلاعات بازیابی شده، استفاده می‌کند. از شاخص‌های به دست آمده، برای بازیابی اطلاعات به وسیله‌ی واسط جستجو استفاده خواهد شد.

ویژگی بارز Nutch در برابر موتورهای جستجوی موجود، ساختار قابل گسترش آن می‌باشد. به طور مثال، Nutch برای زمانی که نیاز به بازیابی یک یا چند دامنه‌ی خاص قابل استفاده است، تا زمانی که بعضی از اطلاعات یک دامنه را می‌خواهیم از صافی عبور دهیم، قابل استفاده است. Nutch با استفاده از ساختار افزونه‌ای که توسط زبان نشانه‌گذاری^۲ به سیستم شناخته می‌شود امکان، چنین کاری را فراهم می‌آورد. چنین ساختاری که همانند ساختار Eclipse می‌باشد، این امکان را ایجاد می‌کند تا بتوان بدون تغییر بنیادی در کد، به اصلاح رفتار برنامه اقدام نمود.

۲-۴ نحوه‌ی عملکرد

۱.۲-۴ عملکرد کلی

طبق شکل ۱-۴، اجزای خزنده، شامل WebDb، لیست واکشی، بازیاب‌ها و به‌روزرسان‌ها است. WebDb یک پایگاه داده‌ی اختصاصی شده است که صفحات بازیابی شده به همراه پیوندهای ورودی و خروجی آن را در خود ذخیره می‌کند. همچنین در مورد هر صفحه، مجموعه‌ی کوچکی از اطلاعات، مانند آخرین زمان بازیابی، را ذخیره می‌کند. لیست واکشی، با استفاده از اطلاعات WebDb تهیه می‌شود. این لیست، شامل پیوند صفحاتی است که باید در این مرحله بازیابی گردد. بازیاب‌ها، با استفاده از این لیست، صفحات را بازیابی نموده و WebDb را متناسب با آن به‌روزرسانی می‌نمایند. در این مرحله، تغییر صفحه نسبت به بازیابی قبل، نیز برای کنترل تناوب بازیابی ذخیره می‌گردد. محتوای به دست آمده نیز برای جستجو استفاده می‌شود. این چرخه، به گونه‌ای طراحی شده است تا بتواند تا ابد اجرا گردد و همواره تصویری به روز از صفحات وب را ارائه دهد.

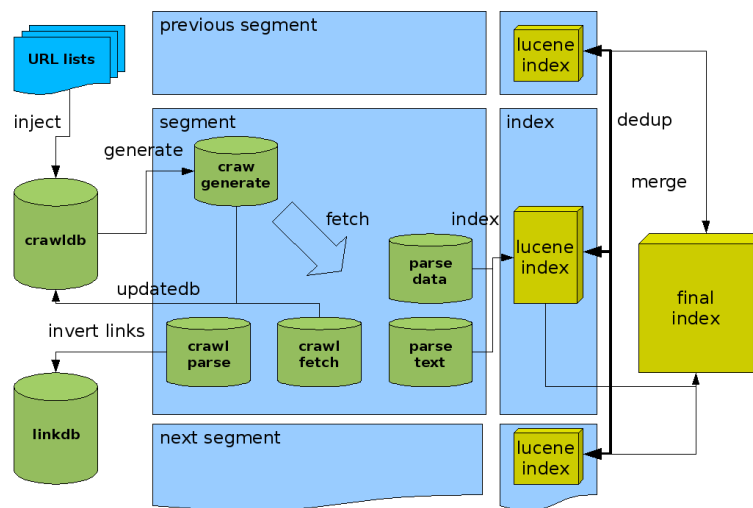
زمانی که صفحات وب، بازیابی گردید، Nutch امکان جستجو را با استفاده از Searcher خود فراهم می‌کند. در واقع پس از این مرحله، ابتدا واحد شاخص‌بندی، محتوای استخراج شده را در لیست‌های شاخص وارون^۳ ذخیره می‌نماید. هر سند، به تعدادی ناحیه‌ی شاخص‌بندی تقسیم می‌گردد و هر کدام در یک رویه‌ی جداگانه شاخص‌بندی می‌شود. در نهایت، استخری^۴ از کارگزارها^۵، ارتباط بین کاربر و واحد جستجو را فراهم می‌کند. شکل کلی این رویه در شکل ۱-۴ آورده شده است.

^۲ Markup language (XML)

^۳ Inverted Index

^۴ pool

^۵ Web Servers



شکل ۴-۱: ساختار خزنده‌ی Nutch.

۲.۲-۴ ساختار افزونه

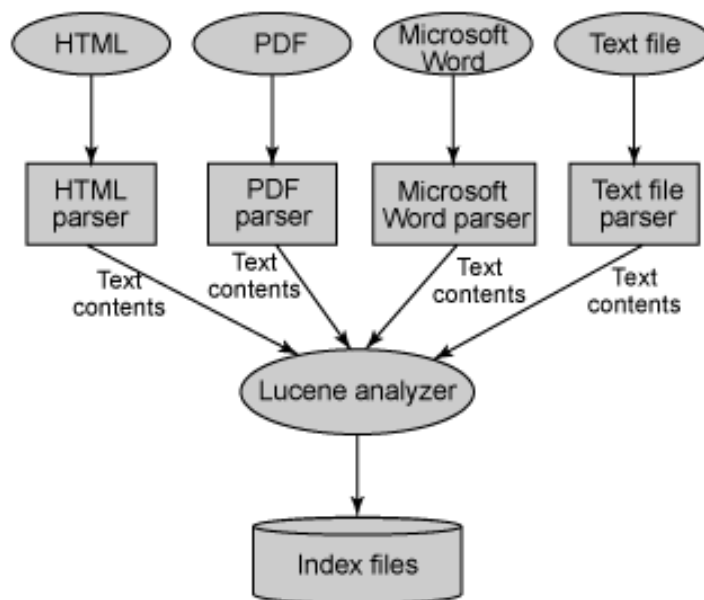
ساختار افزونه‌پذیری Nutch، کامل شبیه ساختار Eclipse می‌باشد [۸]. در واقع، Nutch یک سیستم مرکزی برای کنترل یک مجموعه از ابزارها که با یکدیگر کار می‌کنند، می‌باشد، تا بتواند یک قابلیت را به آن اضافه نمود. بعد از مطالعه‌ی ساختار Eclipse و اعمال آن به ساختار افزونه‌ی Nutch، این نتیجه حاصل شد که مهم‌ترین المان‌های افزونه‌پذیری Nutch، عبارتند از افزونه‌ها، نقاط گسترش‌پذیر و سیستم کنترل افزونه‌ها است. سیستم کنترل این امکان را به Nutch می‌دهد که بدان، کارایی اضافه شود. این کارایی به وسیله‌ی یک افزونه، به سیستم اضافه می‌گردد. هر افزونه، در واقع یک المان گذاردنی^۶ است که تعدادی نقاط گسترش را پیاده‌سازی می‌کند و این قابلیت‌ها، به وسیله‌ی سیستم کنترل مرکزی اجرا می‌شوند.

۳-۴ ساختار Lucene

۱.۳-۴ عملکرد کلی

کتابخانه‌ی Lucene به افراد، امکان افزودن شاخص‌بندی و جستجو در نرم‌افزارهای خود را می‌دهد [۹]. این کتابخانه امکان شاخص‌بندی و جستجو در هر نوع داده‌ای را، تا زمانی که قابلیت تبدیل شدن به متن را داشته باشد، می‌دهد. این بدان معنی است که این کتابخانه امکان جستجو در صفحات وب، فایل‌های PDF و Words را دارد. به این خاطر، Lucene بهترین کتابخانه‌ی برای نوشتن موتور جستجو است.

^۶Pluggable



شکل ۴-۲: ساختار Lucene [۳].

۲.۳-۴ شاخص‌بندی

شاخص‌بندی، در واقع مرحله‌ی تبدیل متن به شاخص می‌باشد. شاخص‌ها در واقع خود، داده‌ساختاری‌اند که سرعت عملیات بازیابی اطلاعات را، بهبود می‌بخشند. شاخص‌ها، جزء المان‌های اصلی Lucene می‌باشند.

برای شاخص‌بندی داده در این سیستم، داده باید به جویباری^۷ از تکه‌های متنی تبدیل گردند. بعد از آن، Lucene، داده‌ها را با تکه تکه کردن جویبار داده و اجرای عملیاتی بر روی آن، آن را برای شاخص‌بندی آماده می‌کند. به طور مثال، یکی از این عملیات می‌تواند، کوچک‌سازی حروف، برای از بین بردن حساسیت جستجو به بزرگی و کوچکی باشد. این مرحله به مرحله‌ی تحلیل معروف است. بعد از اینکه جویبارها، تحلیل گردیدند، داده‌ها آماده برای اضافه شدن به شاخص‌ها می‌باشند. مرحله‌ی شاخص‌بندی در شکل ۴-۲ متصور شده است.

این کتاب‌خانه، روشی نوآورانه برای نگهداری شاخص‌ها به کار می‌برد. این کتاب‌خانه مستقل از تعداد شاخص‌ها، برای هر شاخص از چند قطعه استفاده می‌کند. استفاده کردن از چند قطعه، اضافه کردن یک سند تازه شاخص‌بندی شده را به وسیله‌ی اضافه کردن آن به کوچک‌ترین قطعه که تازه ساخته شده است و ترکیب آن قطعه با بقیه‌ی قطعات به صورت متناوب، تسریع می‌بخشد. این رویه، اضافه کردن سند را بسیار کارا می‌کند، زیرا شاخص‌هایی که در هر لحظه تغییر می‌کند را به شدت کاهش می‌دهد.

بعضی از سیستم‌های بازیابی، برای اضافه شدن یک سند، نیاز به بروز رسانی تمام شاخص دارند، اما Lucene، به علت پشتیبانی از شاخص‌بندی افزایشی، چنین مشکلی را ندارد. چنین امری بدان معناست که Lucene امکان جستجو بر روی سند را بلافاصله بعد از شاخص‌بندی شدن آن بدون نیاز به بروز رسانی تمام شاخص، فراهم می‌سازد.

^۷Stream

۳.۳-۴ تحلیل‌گر

همان طور که در قسمت قبل، بحث شد، تحلیل یکی از مهم‌ترین مراحل، شاخص‌بندی است. این مرحله، داده‌های متنی را به یکی از اساسی‌ترین نمایش شاخص یعنی واژه‌های بنیادی^۸، تبدیل می‌نماید. این واژه‌های بنیادی برای تطابق پرسمان‌ها در مرحله‌ی جستجو با اسناد استفاده می‌شود.

۴-۴ خلاصه

در این فصل، ساختار Nutch به همراه مکانیزم افزونه‌پذیری مورد مطالعه قرار گرفت. همچنین ساختار شاخص‌بندی Lucene و تحلیل‌گر آن نیز به اختصار توضیح داده شد. در فصل آتی، پیاده‌سازی افزونه‌ها و تغییرات اعمال شده توضیح داده خواهد شد.

^۸Terms

فصل ۵

پیاده‌سازی

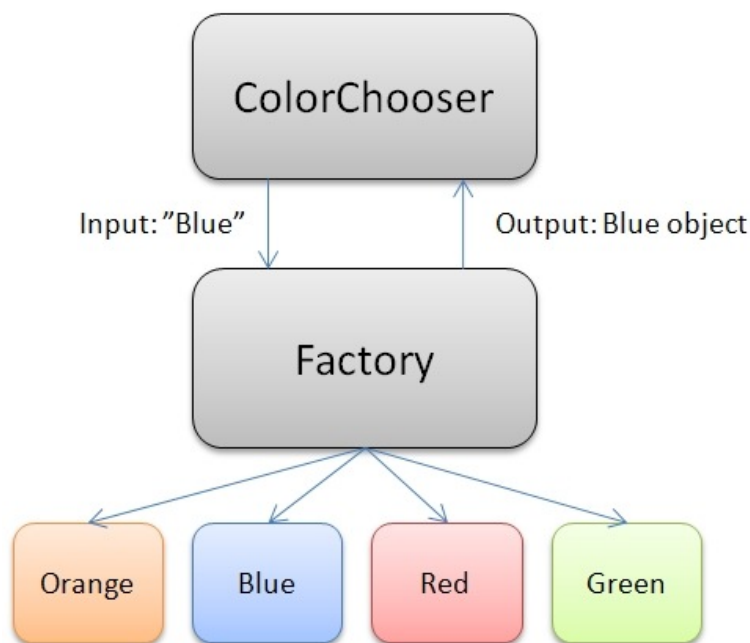
۵-۱ مقدمه

در این فصل روند طی شدن پروژه توضیح داده می‌شود. در این پروژه، در ابتدا به بررسی‌های جامع در مورد وضع سایت‌های فعال کسب‌وکار در ایران، پرداختیم. البته بررسی‌ها به نمونه‌های داخلی محدود نگردید و بررسی‌هایی در مورد نمونه‌های خارجی موفق نیز انجام شد. با بررسی دقیق‌تر، سایت‌های هدف برای استخراج اطلاعات انتخاب شد. همچنین برای امکان ایجاد مشابه فارسی LinkedIn و امکان استخراج اطلاعات طبقه‌بندی شده از LinkedIn به بررسی شمای پایگاهی این سایت، پرداخته شد و تا حد خوبی، ساختار داده‌ای این سایت پرتعداد خارجی را به دست آمد. سپس به مرحله‌ی پیاده‌سازی خزنده پرداختیم. در این مرحله، برای اجرای کار، نیاز به نوشته شدن استخراج‌گر اختصاصی برای هر دامنه احساس می‌گردید. این استخراج‌گر، از هر صفحه‌ی آگهی داده‌های مفید آن را استخراج می‌نماید. این امر، باعث بهبود دقت جستجو در مراحل بعدی می‌گردد، زیرا داده‌های غیر مرتبط به آن آگهی را عملاً حذف می‌نماید. سپس با نوشتن افزونه برای تبدیل این داده‌های استخراج شده، به ساختاری سازگار با Lucene، امکان شاخص‌بندی داده‌ها فراهم گردید. سپس با بررسی میزان تأثیر گذاری این کار بر روی دقت نتایج جستجو پرداخته شد.

۵-۲ پیش‌نیازها

برای سازگارسازی Nutch، با سایت‌های فعال در زمینه‌ی استخدام، نیاز به پیاده‌سازی افزونه‌ی خاص هر سایت، مبرم و اساسی بود. علاوه بر این، نیاز است مکانیزمی تعبیه گردد که با توجه به سایت، افزونه‌ی خاص خود را پیدا کند و آن را اجرا نماید. برای چنین کاری، از الگوی کارخانه^۱ استفاده گردید. ساختار این الگو در شکل ۵-۱ آمده است. این

^۱Factory method



شکل ۱-۵: مثالی از الگوی کارخانه [۱۰].

الگو با توجه به نوع سایت، که از روی آدرس سایت مشخص می‌شود، در بین افزونه‌هایی که خود را در سیستم ثبت نام کرده‌اند، جستجو می‌کند و در صورت پیدا کردن مورد متناسب با آدرس سایت، آن را اجرا می‌کند. چنین ساختاری، اضافه کردن یک افزونه‌ی جدید برای یک سایت جدید را بسیار راحت می‌کند و از اضافه شدن تغییرات منتشر شونده، اجتناب می‌نماید. پس از این مرحله، باید داده‌های استخراج شده را برای شاخص‌بندی به Lucene آماده نمود. این کار توسط افزونه‌ی شاخص‌بندی انجام می‌پذیرد.

۳-۵ مراحل پیاده‌سازی

۱.۳-۵ بررسی و انتخاب سایت‌های هدف

در این مرحله، با بررسی و جستجو در بین سایت‌های فارسی زبان که در زمینه‌ی استخدام و آگهی‌های مربوط به آن، فعالیت دارند، به دست آمد. به طور خاص، در مورد هر کدام، ویژگی‌های بارز استخراج شد. همچنین در این مرحله، به بررسی چند سایت خارجی که پیشروی در این زمینه، پرداخته شد. لیست سایت‌های بررسی شده در جدول ۱-۵ آمده است.

جدول ۵-۱: لیست سایت‌های فعال کاریابی به زبان فارسی

نوع جستجو	به روز رسانی	نوع تعامل با کاربر	آدرس سایت
متن ساده	هر روز	ارسال توضیح ^۲	estekhtam.com
کارنامه، آگهی، نظرات	هر روز	عضویت و ثبت کارنامه	karyab.net
کارجویان، شغل، تخصص و مکان	هر ۲ روز	عضویت و ثبت رزومه	banki.ir
مکان	هر روز	عضویت و ثبت رزومه	e-estekhdam.com
مقطع تحصیلی، رشته، جنسیت، استان و شغل	-	عضویت و ارسال کارنامه	bazarekar.ir
متن ساده	هر روز	عضویت و ارسال رزومه	estekhdamnews.com
متن ساده	هر روز	خبرنامه با ایمیل	unp.ir/jobs.php
متن ساده	هر روز	عضویت کارجویان و کارفرمایان، ثبت کارنامه و آگهی شغل	estekhdamcenter.ir
متن ساده	هر روز	دریافت آگهی و ارسال توضیح	estekhdami.org

۲.۳-۵ بررسی سایت LinkedIn

پس از بررسی سایت‌های فارسی، نوبت به بررسی سایت‌های موفق در زمینه‌ی کاریابی در زبان انگلیسی رسید. یکی از بزرگ‌ترین شبکه‌های کاریابی، شبکه‌ی اجتماعی LinkedIn می‌باشد. این شبکه در حال حاضر دارای ۲۰۰ میلیون کاربر فعال دارد. راه اندازی سایتی مشابه LinkedIn به زبان فارسی و حتی استخراج کارنامک افراد از این سایت یکی از اهداف پروژه برای پیشنهاد کار می‌باشد. به همین منظور استخراج مدل داده‌ای این سایت به عنوان اولین قدم در زمینه‌ی به دست آوردن اطلاعات طبقه‌بندی شده از این سایت می‌باشد. در شکل ۵-۲ شمای پایگاهی استخراج شده از این دامنه نشان شده است.

۳.۳-۵ آماده‌سازی Nutch

در این مرحله، باید ابتدا سایت Nutch را برای اجرا آماده‌سازی نمود. برای این کار، در این پروژه از نسخه‌ی ۲,۲ این نرم‌افزار استفاده نمودیم. این نسخه نسبت به نسخه‌های گذشته تغییرات بنیادی داشته است. به طور مثال می‌توان به ساختار ارتباط با پایگاه آن اشاره نمود. از این نسخه به بعد، با استفاده از تکنولوژی gora این امکان را ایجاد کرده است که Nutch مستقل از نوع پایگاه داده کار کند. به همین منظور در این پروژه از پایگاه داده‌ی ^۳mysql استفاده گردید. برای راه‌اندازی اولین کارهایی که انجام شد تنظیم نوع پایگاه داده‌ی آن و ستون‌های جدول پایگاهی مربوطه بود. پس از این مرحله، باید نرم‌افزار تنظیم می‌گردید. در زیر تنظیمات کلی نرم‌افزار آمده است. در اینجا اسم عامل خزنده و لیست مجاز آن‌ها را مشخص شده است. همچنین استفاده از sql نیز در اینجا مشخص شده است. تنظیمات آخر نیز تنظیمات رمزنگاری ارتباط شبکه‌ای را مشخص می‌کند.

^۳www.mysql.com

```

1 <?xml version="1.0"?>
2 <?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
3
4 <configuration>
5     <property>
6         <name>http.agent.name</name>
7         <value>EmploySpider</value>
8     </property>
9     <property>
10        <name>http.robots.agents</name>
11        <value>EmploySpider,*</value>
12    </property>
13    <property>
14        <name>storage.data.store.class</name>
15        <value>org.apache.gora.sql.store.SqlStore</value>
16    </property>
17    <property>
18        <name>parser.character.encoding.default</name>
19        <value>utf-8</value>
20    </property>
21 </configuration>

```

۴.۳-۵ پیاده‌سازی افزونه‌های تجزیه‌کننده

در این قسمت، نیاز به پیاده‌سازی یک افزونه داشتیم که در آن با توجه به نوع صفحه، پردازشگر مخصوص به آن را اجرا نماید و قسمت‌های مفید صفحه را استخراج نماید. به طور کلی ساختار هر افزونه برای شناخته شدن در Nutch باید مطابق زیر باشد.

```

myPlugin/
    plugin.xml
    build.xml
    ivy.xml

```

```
src/  
  java/  
    org/  
      employ/  
        nutch/  
          parser/  
            ...
```

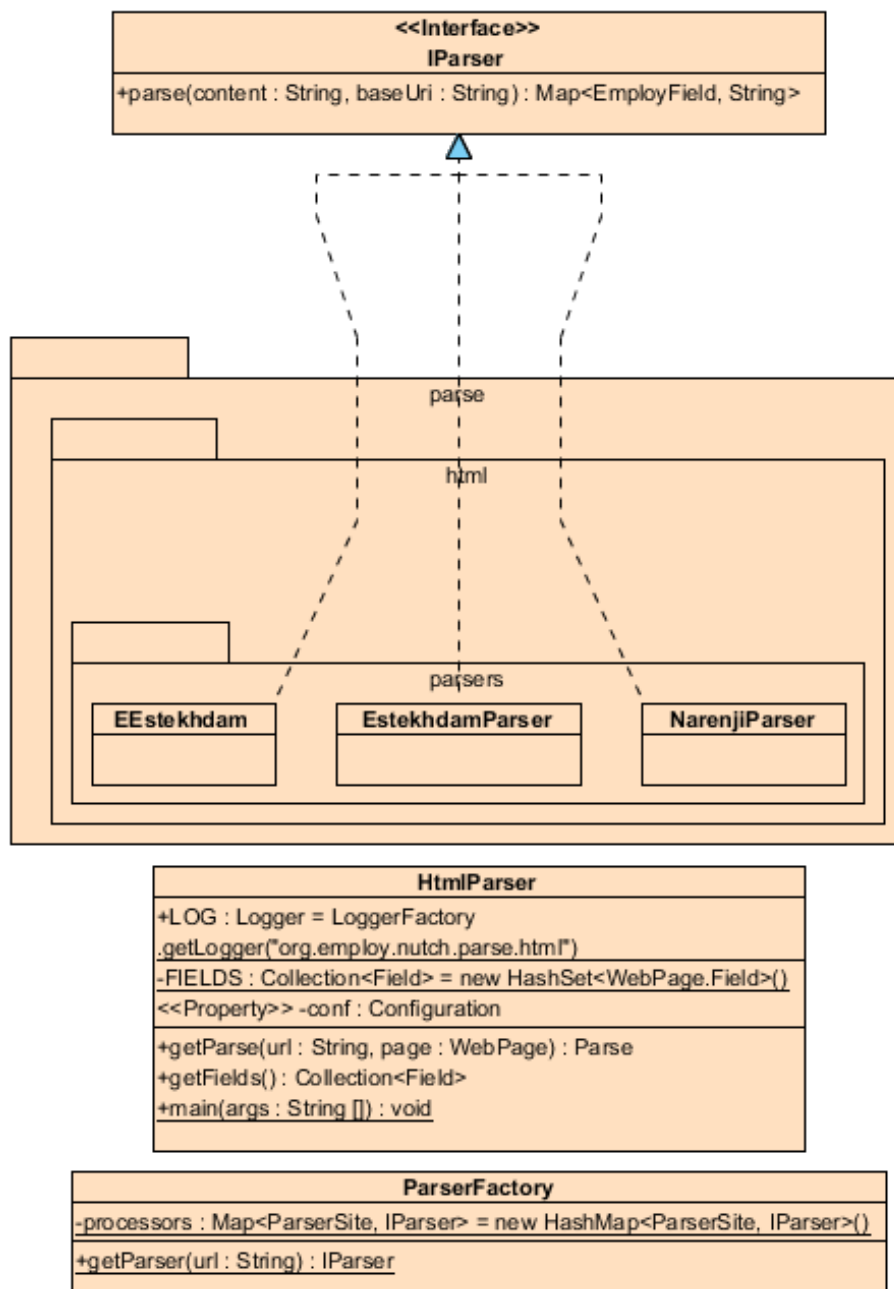
در این ساختار، سه فایل plugin.xml نوع افزونه، نقاط گسترشی که پیاده‌سازی می‌نماید و کلاس‌های قابل اجرا برای هر کدام را مشخص می‌کند. فایل build.xml برای قابل اجرا کردن افزونه برای Nutch استفاده می‌گردد و فایل ivy.xml برای مشخص کردن پیش‌نیازهای این افزونه به کار می‌رود. در شکل ۳-۵ ساختار نمودار رده‌ی افزونه‌ی تولید شده نشان داده شده است. همان طور که می‌بینید برای پشتیبانی از چند دامنه، از الگوی کارخانه استفاده شده است.

۵.۳-۵ پیاده‌سازی افزونه‌ی شاخص‌بندی

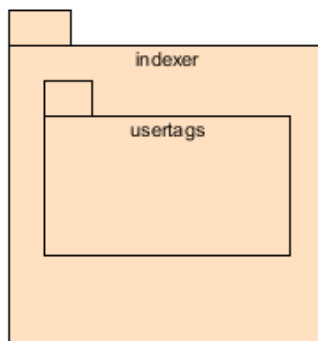
در این قسمت، ساختاری کاملاً مشابه افزونه قبلی ساخته می‌گردد، با این تفاوت که نقاط گسترش متفاوتی را پیاده‌سازی می‌کند و در مرحله‌ی متفاوتی اجرا خواهد شد. ساختار و جزئیات این افزونه در شکل ۴-۵ آمده است.

۴-۵ خلاصه

در این فصل در ابتدا به بررسی روند کارهای انجام شده پرداخته شد. سپس هر کدام از روندها به طور مفصل مورد بررسی قرار گرفت. از جمله روندهای بررسی شده، می‌توان به بررسی جامع سایت‌های فارسی اشاره نمود. همچنین ساختار افزونه‌های Nutch نیز مورد بررسی قرار گرفت. در فصل آتی به بررسی ویژگی‌های کدهای پیاده‌سازی شده پرداخته خواهد شد.



شکل ۵-۳: ساختار افزونه‌ی تجزیه کننده.



UserTagIndexingFilter
-LOG : Log = LogFactory .getLog(UserTagIndexingFilter.class) <hr/> -FIELDS : Set<Field> = new HashSet<WebPage.Field>() <<Property>> -conf : Configuration
+filter(doc : NutchDocument, url : String, page : WebPage) : NutchDocument +getFields() : Collection<Field>

شکل ۴-۵: ساختار افزونه‌ی شاخص‌بند.

فصل ۶

نتایج

۱-۶ مقدمه

در این فصل، قصد بررسی نتایج حاصل از انجام بخش پیاده‌سازی را در مقایسه با اجرای Nutch به صورت خام داریم. نرم‌افزار Nutch در هنگام بازیابی اطلاعات صفحات، تمام اطلاعات را نگه می‌دارد، که بدون حذف قسمت تبلیغات و کنار صفحه‌ها و تیترا بالای صفحه، حجم اطلاعات بسیار بالا است. اما با استفاده از تجزیه‌گر موجود، هم در حجم اطلاعات ذخیره شده صرفه جویی گردید، هم دقت جستجو در مراحل بعدی بهبود یافت. همچنین حذف منابع غیر مرتبط از سایت با حذف کردن صفحات غیر مرتبط با آگهی نیز حذف گردید. در زمینه‌ی شاخص‌بندی، با کاهش حجم اطلاعات شاخص، هم موتور جستجو سریع‌تر نتایج خود را ارائه می‌دهد، هم اینکه کلمات نامرتبط را حذف می‌کند که دقت جستجو را افزایش می‌دهد.

۲-۶ محیط اجرای برنامه

محیط اجرای برنامه، سیستم عامل اوبونتو^۱ نسخه ۱۳/۰۴ بر روی لپ‌تاپ مک^۲ سال ۲۰۱۰ می‌باشد. مشخصات سخت‌افزاری شامل پردازنده ۲ هسته‌ای $T9300$ با سرعت $2/2\text{GHz}$ ، با حافظه‌ی 4GB و فضای دیسک ذخیره‌سازی 40GB می‌باشد.

در تمام مراحل از Nutch نسخه‌ی ۲/۲ و Lucene نسخه‌ی ۴/۴ استفاده گردید و به 3GB Nutch حافظه اختصاص داده شد.

^۱Ubuntu

^۲Macbook

جدول ۶-۱: پرسمان‌های به کار رفته برای بررسی دقت جستجو

پرسمان
استخدام بانک
استخدام شهرداری
استخدام نفت
استخدام بانک کشاورزی
استخدام وزارت دفاع
استخدام همشهری
استخدام نیروی انتظامی

۳-۶ روش به کار رفته

سه آزمایش برای جمع‌آوری داده توسط محیط بیان شده ارائه شد. اولین آزمایش با استفاده از Nutch برای بازیابی اطلاعات ۳ سایت، تا عمق ۵ و ۵۰ صفحه‌ی اول در هر عمق انجام گردید. سایت‌های مورد آزمایش عبارتند از:

– www.estekhtam.com

– www.e-estekhdam.com

– www.estekhdami.org

در آزمایش اول، برای سه سایت بالا، تعداد کلیدواژه‌های ذخیره شده را با حالت عادی مقایسه می‌گردد. در آزمایش دوم، برای سه سایت بالا، مدت زمان میانگین بازیابی را با ۵ بار اجرای هر کدام، اندازه‌گیری می‌شود. در آزمایش سوم، به صورت دستی برای هر سه سایت، تعداد صفحات مرتبط در ۲۰ صفحه‌ی پاسخ داده شده به عنوان صفحه‌ی مرتبط با هفت پرسمان استخدای پرکاربرد که از سایت google استخراج شده بود، را به دست آورده شده است. لیست پرسمان‌ها در جدول ۶-۱ آورده شده است.

۴-۶ نتایج و بحث

۱.۴-۶ تعداد واژه‌ها

در آزمایش اول، همان طور که می‌بینید هنگامی که جای استفاده از افزونه‌ی Nutch از افزونه‌ی طراحی شده استفاده گردید، به علت کاهش قسمت‌های تحت درگیر سایت، تعداد کلیدواژه‌ها به میزان ۵۵ درصد تا ۷۵ درصد کاهش پیدا کرده است. جزئیات هر سایت در جدول ۶-۲ آمده است.

جدول ۶-۲: تعداد کلیدواژه‌های ذخیره شده بر اساس سایت

تعداد کلیدواژه در حالت عادی	تعداد کلیدواژه بهبود یافته	آدرس سایت
۵۵۲۴	۲۲۰۲	www.estekhtam.com
۲۷۴۷	۱۲۶۶	www.e-estekhdam.com
۲۲۲۳	۴۹۲	www.estekhdami.org

جدول ۶-۳: مدت زمان بازیابی بر اساس سایت

زمان بازیابی در حالت عادی (s)	زمان بازیابی بهبود یافته (s)	آدرس سایت
۳۶۲/۹۲	۳۷۵/۲	www.estekhtam.com
۱۰۴/۵۶	۱۰۵/۱۲	www.e-estekhdam.com
۲۸۲/۰۴	۳۱۶/۳۷	www.estekhdami.org

۶-۲.۴ زمان بازیابی اطلاعات

در آزمایش دوم، همان طور که می‌بینید هنگامی که از افزونه‌ی نوشته شده استفاده می‌گردد، زمان بازیابی اطلاعات به علت افزایش پردازش‌های انجام شده افزایش می‌یابد. به همین منظور نیاز به قدرت پردازشی و حافظه‌ی بیشتری برای کار نیاز دارد. جزئیات در جدول ۶-۳ آورده شده است.

۶-۳.۴ نتایج جستجو

در آزمایش سوم، نتایج یک جستجوی خاص و دقت هر کدام قبل و پس از استفاده از افزونه با استفاده از Lucene مورد بررسی قرار گرفت. قابل توجه است که در روش به کار گرفته شده، سایت‌هایی که در خود از چند آگهی اطلاعات دارند دیگر در رده‌های بالا دیده نمی‌شود، به خصوص که هر کدام از صفحات، لیستی غیر مرتبط از آگهی‌ها را، در کنار خود دارند. جزئیات در جدول ۶-۴ آورده شده است.

جدول ۶-۴: میزان دقت جستجو به ازای هر سایت

دقت جستجو در حالت عادی	دقت جستجو بهبود یافته	آدرس سایت
۷	۱۳	www.estekhtam.com
۱۲	۱۸	www.e-estekhdam.com
۱۱	۱۶	www.estekhdami.org

۵-۶ خلاصه

در این فصل، سعی بر آن داشتیم تا تأثیرات افزونه‌ی نوشته شده را بر قسمت‌های مختلف سیستم جستجو بررسی کنیم و ویژگی‌های مثبت و منفی آن را استخراج نماییم. در کل از نظر دقت جستجو و حجم موتور جستجو، استفاده از افزونه نوشته شده، مفید بوده است. اما استفاده از این افزونه نیاز به قدرت پردازشی بالاتری دارد که سخت افزار قوی‌تری را می‌طلبد.

فصل ۷

نتیجه‌گیری

۱-۷ خلاصه

استخراج مطالب اصلی صفحات وب، یکی از چالش‌برانگیزترین کار برای به دست آوردن اطلاعات مفید از صفحات وب می‌باشد. یکی از روش‌های اجرای این کار، نوشتن تجزیه‌گر اختصاصی برای یک سایت می‌باشد. اما به علت تنوع سایت‌ها، امکان اجرای دقیق کار، برای طیف وسیعی از سایت‌ها وجود ندارد، اما امکان اجرای این کار برای مجموعه‌ای محدود از سایت‌ها قابل اجرا است. به همین منظور استفاده از تجزیه‌گر اطلاعات برای مجموعه‌ای از سایت‌های کاریابی پیشنهاد گردید. گرچه این کار، دقت نوشتن تجزیه‌گر اختصاصی برای هر سایت را ندارد، اما از استخراج‌کننده‌های مطالب اصلی و کلی، معمولاً بسیار کاراتر می‌باشد. از طرفی روش به کار رفته صرفه جویی در حافظه‌ی مصرفی و افزایش سرعت جستجو را تضمین می‌کند.

آزمایش‌های انجام شده، نشان داد که استفاده از این روش معایب و فواید خاص خود را دارد. از جمله فواید آن بهبود دقت جستجو می‌باشد. زیرا استخراج مطالب مفید هر صفحه و حذف مطالب نامرتب و تبلیغ گونه در هر صفحه در دقت جستجو تأثیر بسزایی دارد. از طرفی، اجرای افزونه باعث کاهش سرعت اجرای بازیابی می‌گردد و تجزیه‌ی هر صفحه، حافظه‌ی مصرفی آن لحظه را به علت داده‌ساختار به‌کارگرفته شده بالا می‌برد. در مقابل آن، کاهش حجم حافظه‌ی ذخیره‌شده‌ی مصرفی، نیز به شدت محسوس بود. این ویژگی در شاخص‌بندی یک نکته‌ی مثبت است.

۲-۷ کارهای آینده

در گام بعدی، نیاز داریم افزونه‌ی نوشته شده را برای سایت‌های مختلف موجود سازگار کنیم. همچنین سطح اطلاعات سیستم را با سازگار کردن به استخراج کارنامک افراد از سایت‌های ارائه دهنده‌ی خدمات در این زمینه بهبود بخشیم. همچنین امکان بهبود سرعت استخراج اطلاعات، را با تغییر مکانیزم استخراج و استفاده از الگوهای زبان منظم انجام

دهیم. همچنین افزونه می‌تواند برای سایت‌هایی که سازگار نیست یک الگوریتم عادی برای استخراج محتوای اصلی استفاده نماید.

سپاس

از استاد بزرگوارمان، دکتر حمید بیگی که با کمک‌ها و راهنمایی‌های بی‌دریغشان، ما را در انجام این پروژه یاری داده‌اند، تشکر و قدردانی می‌کنیم.

هم‌چنین از آقای محمود نشاطی، به عنوان سرپرست پروژه و یاری‌دهنده در این مسیر صمیمانه سپاس گزاریم.

Bibliography

- [1] Search in the cloud. *www.ektron.com/Blogs/Udaiappa-Ramachandran/Search-In-The-Cloud/* (accessed in Sat, 02 Nov 2013 07:35:48 GMT). 7, 19
- [2] Web crawlers. googlebot. *http://www.milkaddict.com/web-crawlers-googlebot/* (accessed in Sat, 02 Nov 2013 07:42:27 GMT). 7, 20
- [3] Realtime full-text search. *http://blog.andreamostosi.name/tag/solr/* (accessed in Sat, 02 Nov 2013 07:48:26 GMT). 7, 24
- [4] World internet usage and population statistics available at. *http://www.internetworldstats.com/stats.htm*. 14
- [5] Sandeep Sharma. Web-crawling approaches in search engines. *Thapar University, Patiala*, pages 9–20, 2008. 14, 15, 16
- [6] Andreas Paepcke Arvind Arasu, Junghoo Cho. Searching the web. *Computer Science Department, Stanford University*. 14
- [7] Konchady. *Building Search Applications: Lucene, LingPipe, and Gate*. Mustru Publishing, 2008. 21
- [8] A. Bolour. Notes on the eclipse plug-in architecture. *http://www.eclipse.org/articles/Article-Plug-in-architecture/plugin_architecture.html*. 23
- [9] O. Gospodnetic and E. Hatcher. *Lucene In Action*. Greenwich: Manning Publications Co., 2005. 23
- [10] What is the factory method pattern. *http://blog.fossmo.net/post/What-is-the-factory-method-pattern.aspx* (accessed in Sat, 02 Nov 2013 07:52:18 GMT). 7, 27

abstract

Many advantages of the Internet have turned it into one of the most popular ways to communicate and exchange ideas. Due to increasing data provided in the World Wide Web, accessing particular information, which matches user needs, becomes challenging. So we need to have some kinds of tools for managing access points and searching through this massive source of data. As a result, general software for searching through the Internet's data implemented. However, these general software don't fulfill users for special needs. For example, one of these needs is employment. As we know, there aren't any efficient Persian search engines for finding proper work for applicants.

This project, by using active Persian employment websites, develops an integrated search engine. In the middle, we need to extract employment advertisements and applicants resumes from these websites. In addition, we need to study successful English employment websites such as LinkedIn and extract their database schema for saving resumes. Finally, we enhance an open source search engine to extract Persian Employments advertisements efficiently. As a result, we get more accurate and robust search experience among these advertisements and decrease index size which is very important in large scale search engines.

Keywords: Advertisement, Search Engines, Nutch, Lucene, Data Mining



Sharif University of Technology
Computer Engineering Department

B.Sc. Thesis
Computer Engineering - Software

Title:

Implementing Inteligent Crawler Of Persian Business Search Engine

By:

Behnam Hatami Varaneh

Supervisor:

Dr. Hamid Beigi

Aguest 2013