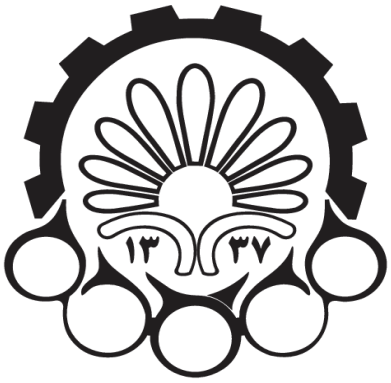


دانشگاه صنعتی امیرکبیر

دانشکده مهندسی کامپیوتر

درس:

بازیابی اطلاعات

تعریف پروژه – مرحله اول تا چهارم

نیمسال اول

سال تحصیلی ۹۸–۹۹

۱- مقدمه

هدف نهایی این پروژه ایجاد یک موتور جستجو برای بازیابی اخبار از وب‌سایت‌های خبری فارسی است. کاربر پرسمان خود را وارد کرده و سامانه خبرهای مربوط به آن را از میان خبرهای منتشر شده در چندین وب‌سایت خبری بازیابی و به کاربر ارائه می‌کند. پروژه در چهار مرحله تعریف شده است که عبارتند از:

- **مرحله اول:** ایجاد واسط کاربری به همراه یک مدل بازیابی اطلاعات ساده
- **مرحله دوم:** تکمیل مدل بازیابی اطلاعات و ارائه قابلیت‌های کارکردی پیشرفته تر
- **مرحله سوم:** استفاده از روش های خوشه‌بندی و دسته بندی برای بهبود نتایج
- **مرحله چهارم:** خزش و جستجوی آنلاین

۲- مجموعه داده پروژه

مجموعه داده مورد استفاده در این پروژه مجموعه‌ای از خبرهای واکشی شده از چند وب‌سایت خبری فارسی است که در قالب یک فایل اکسل در اختیار شما قرار خواهد گرفت. هر سطر این فایل حاوی یک خبر خواهد بود. برای هر خبر اطلاعات زیر در فایل مذکور وجود دارد:

- تاریخ و ساعت انتشار خبر
- عنوان خبر
- خلاصه خبر
- متن اصلی خبر
- کلمات کلیدی
- نام خبرگزاری یا سایت انتشار دهنده خبر
- لینک عکس مربوط به خبر

۳- مرحله‌ی اول پروژه

در این مرحله از پروژه شما می‌بایست واسط کاربری سامانه بازیابی اطلاعات خود را ایجاد و از یک مدل ساده بازیابی اطلاعات به منظور پردازش مجموعه اسناد ورودی و واکشی اخبار مرتبط با پرسمان کاربر استفاده کنید. در ادامه به بررسی ویژگی‌های مد نظر در واسط کاربری و مدل بازیابی اطلاعات سامانه در این مرحله پرداخته می‌شود.

۳-۱ واسط کاربری

واسط کاربری موتور جستجوی شما می‌تواند یک برنامه‌ی تحت وب، موبایلی یا دستکتاپی باشد. واسط کاربری طراحی شده می‌بایست تا حد ممکن ساده و در عین حال دارای ظاهری حرفه‌ای باشد. دو کارکرد اصلی این واسط کاربری عبارتند از:

- دریافت پرسمان کاربر
- ارائه نتایج جستجو به کاربر

در بخش اول، پرسمان کاربر در قالب یک متن آزاد دریافت می‌گردد. حداقل عملگرهای قابل استفاده در این بخش علامت «!» برای NOT، عملگر "" برای تعیین یک عبارت، کلمه کلیدی :source برای تعیین منبع خبر و کلمه :cat برای تعیین دسته خبر است. در این مرحله نیازی به پیاده‌سازی بخش‌های مربوط به منبع و دسته خبر در موتور جستجو نیست اما در واسط کاربری می‌بایست این عبارات دریافت و پردازش شوند.

بعد از ورود عبارت پرسمان و فشردن دکمه جستجو، نتایج بازیابی شده به کاربر نمایش داده می‌شود. واسط کاربری طراحی شده می‌بایست خبرهای مرتبط با درخواست کاربر را به صورت مرتب شده و صفحه‌بندی شده به کاربر نمایش دهد و قابلیت مرتب سازی نتایج بر حسب میزان ارتباط و زمان انتشار را داشته باشید. در صفحه نتایج، می‌بایست موارد زیر در مورد هر خبر مرتبط نمایش داده شود:

- عکس خبر
- عنوان خبر
- زمان و منبع انتشار خبر
- تکه متن(های) دربرگیرنده عبارت جستجو در متن خبر با برجسته کردن عبارت پرسمان

با کلیک بر روی هر یک از نتایج برگردانده شده، می‌بایست اطلاعات کامل خبر در یک صفحه مناسب به کاربر نمایش داده شود.

۳-۲ مدل بازیابی اطلاعات

دو وظیفه اصلی مدل بازیابی اطلاعاتی که در این مرحله پیاده‌سازی می‌شود عبارتند از:

- شاخص‌گذاری مجموعه اسناد
- اجرای پرسمان کاربر بر روی شاخص اسناد

برای شاخص‌گذاری اسناد لازم است بخش‌های زیر پیاده‌سازی شوند:

- واکشی خبر
- استخراج توکن
- ریشه‌یابی کلمات
- همسان‌ساز کلمات
- حذف کلمات پرتکرار

- ایجاد شاخص معکوس مکانی (شامل فرهنگ لغات و لیست‌های پست‌ها)

در بخش جستجو، پرسمان کاربر با در نظر گرفتن دو عملگر یاد شده در بالا بر روی شاخص معکوس مکانی ایجاد شده اجرا و نتایج به گونه‌ای که قبلا گفته شد به کاربر نمایش داده می‌شود. جستجوی مد نظر در این مرحله جستجوی دودویی است. به عنوان مثال برای عبارت جستجویی مانند "بازیابی اطلاعات" امیرکبیر !درس می‌بایست تمام اسنادی که در آنها عبارت «بازیابی اطلاعات» و کلمه «امیرکبیر» آمده باشد و کلمه «درس» نیامده باشد بازگردانده شود.

۴- مرحله‌ی دوم پروژه

در این مرحله می‌بایست بخش‌های پردازش اسناد (شامل نرمال‌سازی، استخراج توکن، هم‌سان‌سازی کلمات و ریشه‌یابی کلمات) تکمیل شود. هم‌چنین مدل بازیابی اطلاعات نیز باید بتواند نتایج جستجو را بر اساس ارتباط رتبه‌بندی کند. مدل بازیابی اطلاعات این کار را با مدل‌سازی اسناد در فضای برداری انجام می‌دهد. به این صورت که برای هر سند یک بردار عددی استخراج می‌شود که بازنمایی آن سند در فضای برداری است. سپس با داشتن یک پرسمان از کاربر ابتدا آن را به فضای برداری برده و سپس با استفاده از یک معیار شباهت مناسب، فاصله‌ی بردار عددی پرسمان را با تمام اسناد در فضای برداری محاسبه کرده و در نهایت نتایج خروجی را بر اساس شباهت مرتب‌سازی می‌کنیم. در ادامه جزییات بخش‌های مختلف پردازش اسناد، نحوه‌ی محاسبه‌ی بازنمایی برداری اسناد و معیار شباهت ارایه شده است.

۴-۱- بخش نرمال‌سازی متن

این بخش از سامانه، وظیفه‌ی یکسان‌سازی کاراکترها و پردازش مناسب کاراکترهای غیر الفبایی را برعهده دارد. برخی کاراکترهای فارسی دارای تنوع هستند و گاه با نسخه‌های عربی خود جابجا می‌شوند. به عنوان نمونه کاراکترهای «ك» و «ک» دو شکل نوشتاری از یک کاراکتر هستند. این مسئله برای استخراج دیکشنری و مقایسه کلمات مناسب نیست و باید متن وروی به شکلی استاندارد نرمال شود. در این بخش شما باید یک لیست از تمام کاراکترهای موجود استخراج کنید و در صورت نیاز هر کدام از آنها را به یک کاراکتر استاندارد نگاشت کنید. همچنین کاراکترهای غیر الفبایی مثل اعراب‌ها، انواع نیم‌فاصله، انواع جداکننده، کاراکترهای غیرمعمول، ایموجی‌ها و علامت‌های نشانه‌گذاری مثل نقطه، ویرگول و ... را به طور مناسب برای موتور جستجو پردازش کنید.

۴-۲- بخش استخراج توکن

این بخش از موتور جستجو وظیفه‌ی تکه‌تکه کردن متن ورودی را برعهده دارد طوری که هر تکه از آن یک کلمه (ترم) با معنی و کامل باشد. در طراحی و پیاده‌سازی این بخش باید نکات زیر مد نظر قرار داده شوند:

- افعال به هر شکلی که در ورودی ظاهر شدند، یک کلمه در نظر گرفته شوند. به عنوان مثال اگر «می‌توانسته‌ام» در ورودی به شکل «می توانسته ام» یا هر شکل دیگری ظاهر شده باشد باید یک کلمه در نظر گرفته شود.

- اگر یک کلمه با علامت‌های جمع به کار رفته بود، یک کلمه در نظر گرفته شود. مثلا «درخت ها» یک کلمه است.
- عبارت‌های ترکیبی پرکاربرد که تکه‌هایشان غالباً در کنار هم ظاهر می‌شوند باید یک تکه در نظر گرفته شوند. به عنوان مثال عبارت‌های «فی‌مابین»، «چنان چه»، «بنا بر این»، «علی ای حال»، «مع ذلک» و ... همه یک کلمه محسوب می‌شوند و به هر شکلی که در سند ظاهر شدند باید یک تکه در نظر گرفته شوند. یک لیست ۲۰ تایی از این عبارات تهیه کنید و برای تشخیص این عبارت‌ها از آن استفاده کنید. (آیا می‌توانید روشی خودکار برای استخراج این نوع عبارات پیشنهاد کنید؟)

۴-۳- بخش هم‌سان‌سازی کلمات

برخی کلمات در زبان فارسی به چند شکل نوشته می‌شوند. مثلاً «زغال» و «ذغال» یکی از این کلمات است. در این بخش از موتور جستجو، شما باید این کلمات را تشخیص داده و یک نسخه استاندارد از آنها را حفظ کنید و همیشه دیگری را به نسخه‌ی استاندارد تبدیل کنید. یک لیست ۲۰ تایی از این کلمات پیدا کنید و برای تشخیص و تبدیل از آن استفاده کنید. مورد دیگری که می‌بایست در بخش هم‌سان‌سازی پوشش داده شود قالب اختصاری کلمات است. مثلا ممکن است به جای خبرگزاری جمهوری اسلامی از عبارت خبرگزاری ج.ا. استفاده شود. فهرستی از این اختصارات در اسناد ورودی تهیه کنید و در بخش هم‌سان‌ساز سامانه بازیابی اطلاعات خود از آن استفاده نمایید.

۴-۴- ریشه‌یابی کلمات

در این بخش از سامانه کلمات به ریشه‌شان تبدیل می‌شوند تا در ادامه‌ی پردازش از نسخه‌ی ریشه‌ی کلمه استفاده شود تا تفاوت در صرف کلمه باعث نشود یک سند در نتیجه‌ی جستجو ظاهر نشود. به عنوان مثال اگر پرسمان ورودی «روش‌های پختن عدسی» باشد، ما می‌توانیم اسنادی که در آنها کلمه‌های «روش»، «پختم» و «عدس» ظاهر شده‌اند هم در نتایج جستجو بیاوریم. در طراحی و پیاده‌سازی بخش ریشه‌یاب کلمات باید نکات زیر در نظر گرفته شوند:

- فعل‌ها به ساده‌ترین شکل خود تبدیل شوند.
- کلمات جمع به ساده‌ترین شکل خود تبدیل شوند.
- پیشوندها یا پسوندهای چسبیده به کلمات حذف شوند.

مثال‌های جدول زیر را در نظر بگیرید:

ورودی	ریشه	ورودی	ریشه
می‌روم	رو	**درختان**	درخت
گفتند	گفت	**کتابم**	کتاب
می‌خواهید	خواه	**عادلانه‌ترین**	عادلانه
رفته است	رفت است	**جعبه‌ای**	جعبه
شوند	شو	**خانه‌هایمان**	خانه

۴-۵- مدل بازیابی اطلاعات

در این بخش باید تمام اسناد را با استفاده از روش وزن‌دهی tf-idf به بردارهای عددی تبدیل کنید. برای این کار باید وزن هر کلمه در هر سند را محاسبه کرده و در نهایت برداری شامل وزن‌های تمام کلمات آن سند، بازنمایی آن سند در فضای برداری خواهد بود. محاسبه‌ی وزن هر کلمه t در یک سند d با داشتن مجموعه‌ی تمام اسناد D با استفاده از معادله‌ی زیر محاسبه می‌شود:

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D) = \left(1 + \log\left(f_{t,d}\right)\right) \times \log\left(\frac{N}{n_t}\right)$$

که در آن $f_{t,d}$ تعداد تکرار کلمه‌ی t در سند d و n_t تعداد سندهایی است که کلمه‌ی t در آنها ظاهر شده است. توضیحات بیشتر این روش در فصل ۶ کتاب آمده است. برای مطالعه‌ی بیشتر می‌توانید به این لینک نیز مراجعه کنید.

شما باید در این مرحله از پروژه برای تمام سندها بردار متناظرشان را استخراج کنید و سپس با داشتن پرسمان کاربر، بردار مخصوص پرسمان را نیز استخراج کنید. سپس با استفاده از معیار شباهت سعی کنید اسنادی را که بیشترین شباهت (کمترین فاصله) را به پرسمان ورودی دارند پیدا کنید. سپس آنها را به ترتیب شباهت نمایش دهید. معیارهای فاصله‌ی مختلف می‌تواند برای این کار در نظر گرفته شود که ساده‌ترین آنها شباهت کسینوسی بین بردارها است که زاویه‌ی بین آنها را محاسبه می‌کند. این معیار به صورت زیر تعریف می‌شود:

$$similarity(a,b) = \cos(\theta) = \frac{a.b}{\|a\|\|b\|} = \frac{\sum_{i=1}^{N} a_i b_i}{\sqrt{\sum_{i=1}^{N} a_i^2}\sqrt{\sum_{i=1}^{N} b_i^2}}$$

در این مرحله از پروژه نیز پرسمان می‌تواند شامل عملگرهایی که در مرحله‌ی یک تعریف شده بودند باشد. پس توجه کنید که باید شاخص معکوس مکانی را برای این عملگرها حفظ کنید. علاوه بر شاخص معکوس مکانی، شاخص‌گذاری مبتنی بر tf-idf هم انجام دهید که برای مرتب‌سازی نتیجه‌ی جستجو کاربرد دارد.

در انتهای کار برای نمایش یک صفحه از نتایج پرسمان فقط کافیست K سندی انتخاب شوند که بیشترین شباهت را به پرسمان داشتند. ساده‌ترین راه حل برای این کار مرتب‌سازی تمام اسناد براساس شباهت‌شان با پرسمان است که هزینه زمانی این کار از مرتبه‌ی O(nlogn) است که با فرض زیاد بودن تعداد اسناد می‌تواند باعث زیاد شدن شدید زمان پاسخ موتور جستجو شود. برای حل این مسئله از پشته (heap) استفاده کنید و برای نمایش هر صفحه تنها K سند با بیشترین شباهت را از آن بیرون بکشید. توجه کنید که ساختن پشته از مرتبه‌ی زمانی O(2n) و استخراج K سند با بیشترین مقدار از مرتبه‌ی O(logn) است و در مجموع این تکنیک می‌تواند حدوداً مشکل زیاد بودن زمان پاسخ را حل کند. توجه کنید که اسناد با امتیاز صفر نیازی نیست در پشته ریخته شوند. برای شناسایی این اسناد و حذف آنها در مرحله اول از تکنیک Index elimination استفاده کنید.

۴-۶- بخش‌های امتیازی مرحله‌ی دوم

در ادامه بخش‌های امتیازی پروژه معرفی شده‌اند. امتیاز پیاده‌سازی هر کدام از این موارد در ادامه ذکر شده است. توجه کنید که هر گروه می‌تواند حداکثر ۱۰ امتیاز اضافه بر نمره پروژه دریافت کند و پیاده‌سازی موارد بیشتر منجر به نمره اضافه نخواهد شد. در صورت پیاده‌سازی هر یک از این بخش‌ها، توضیحات کامل مربوط به آن را در گزارش‌تان بیاورید. برای بخش‌هایی که به موتور جستجو اضافه می‌کنید باید در مورد مزیت استفاده از تکنیک، تاثیر آن در موتور جستجو و مقایسه‌ی آن با نسخه‌ی بدون استفاده از تکنیک را در گزارش ذکر کنید.

- **۵ امتیاز: بررسی Heaps' law و Zipf's law در اسناد ورودی**
- **۵ امتیاز: وزن‌دهی متفاوت اسناد و پرسمان به روش‌های زیر و مقایسه نتایج آنها برای پرسمان‌ها**

Recommended tf–idf weighting schemes

weighting scheme	document term weight	query term weight
1	$f_{t,d} \cdot \log \frac{N}{n_t}$	$\left(0.5 + 0.5\frac{f_{t,q}}{\max_t f_{t,q}}\right) \cdot \log \frac{N}{n_t}$
2	$1 + \log f_{t,d}$	$\log\left(1 + \frac{N}{n_t}\right)$
3	$(1 + \log f_{t,d}) \cdot \log \frac{N}{n_t}$	$(1 + \log f_{t,q}) \cdot \log \frac{N}{n_t}$

- **۵ امتیاز: افزایش سرعت پردازش پرسمان با روش Inexact top K document retrieval (جزییات در فصل ۷ کتاب)**
- **۵ امتیاز: افزایش سرعت پردازش پرسمان با روش Champion lists (جزییات در فصل ۷ کتاب)**

۵- مرحله‌ی سوم پروژه

تعداد سندهایی که در یک موتور جستجو شاخص‌گذاری می‌شوند بسیار بسیار زیاد است. در سال ۲۰۱۳، سی تریلیون سند در گوگل شاخص‌گذاری شده بود و ماهانه ۱۰۰ میلیارد جستجو روی آنها انجام می‌شد. در این بخش از پروژه می‌خواهیم مقیاس موتور جستجویی که در دو مرحله‌ی گذشته طراحی و پیاده‌سازی شده است را بزرگ کنیم. اسناد ورودی این بخش ۱۰۰ هزار خبر هستند که باید در موتور جستجو شاخص‌گذاری شده و مورد پرسمان قرار بگیرند.

در مرحله‌ی دوم پروژه، بردار ویژگی اسناد با استفاده از وزن‌دهی tf-idf استخراج شدند. در هنگام جستجو نیز شباهت کسینوسی بردار ویژگی پرسمان (که به صورت مشابه استخراج شده) را با تمام اسناد محاسبه کرده و شبیه‌ترین اسناد را به عنوان نتیجه پرسمان انتخاب می‌کردیم. با افزایش حجم اسناد ورودی مقایسه پرسمان با تمام اسناد به صورت کارا و در زمان مناسب امکان‌پذیر نیست. برای حل این مسئله در این مرحله از خوشه‌بندی بردارهای ویژگی اسناد استفاده می‌کنیم تا بردار ویژگی پرسمان را به جای مقایسه با تمام اسناد فقط با اسناد یک (یا چند) خوشه مقایسه کنیم.

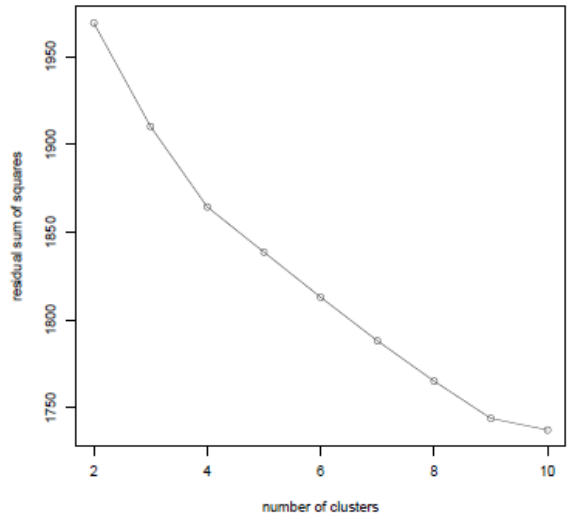
در مرحله‌ی دوم پروژه تمام بخش‌های موتور جستجو اعم از نرمال‌سازی کلمات، استخراج توکن، ریشه‌یابی و هم‌سان‌سازی کلمات به صورت کامل پیاده‌سازی شده‌اند. در این مرحله از همان بخش‌ها استفاده می‌شود و تمرکز اصلی روی بهبود مدل بازیابی اطلاعات است که خوشه‌بندی را در مرحله شاخص‌گذاری انجام داده و پرسمان را روی خوشه‌های مختلف جستجو کند.

علاوه بر خوشه‌بندی، دسته‌بندی اخبار نیز در این مرحله از پروژه بایستی پیاده‌سازی شود. به این معنا که هر خبر به یکی از دسته‌های سیاسی، ورزشی، اقتصادی و ... نگاشت شوند تا در هنگام جستجو بتوان مشخص کرد نتایج از کدام دسته‌های خبری باشند.

هم‌چنین در این مرحله استخراج اخبار مشابه نیز باید پیاده‌سازی شود. مشابه بودن دو خبر بر اساس محتوا و زمان انتشار آنها مشخص می‌شود. سپس در هنگام نمایش نتایج جستجو، برای هر خبر که در لیست نتایج ظاهر می‌شود باید اخبار مشابه با آن نیز به صورت مرتب (بر اساس شباهت یا زمان انتشار) نمایش داده شوند.

۵-۱- مدل بازیابی اطلاعات

در این بخش باید با استفاده از الگوریتم k-means بردارهای ویژگی که برای اسناد در فضای برداری استخراج شده است را به صورت مسطح (flat) خوشه‌بندی کنید. جزییات الگوریتم k-means را می‌توانید در بخش ۱۶,۴ کتاب مطالعه کنید. در این الگوریتم تعداد خوشه‌ها باید به عنوان پارامتر الگوریتم از قبل مشخص باشد اما در دادگان اخباری که در اختیار داریم مقدار بهینه برای این پارامتر را نمی‌دانیم. برای پیدا کردن مقدار بهینه برای این پارامتر تعدادی k پیشنهادی در نظر بگیرید و با تمام آنها الگوریتم خوشه‌بندی را انجام دهید و برای هر کدام معیار RSS که به نحوی نشان‌دهنده‌ی کیفیت خوشه‌بندی است را استخراج کنید. شیب نمودار مشابه شکل زیر خواهد شد. در این حالت مقدار بهینه k مقادیری است که در نمودار شکستگی (زانو) پیدا می‌کند. شکستگی نمودار جایی است که شیب آن تغییر ناگهانی می‌کند مثلا در شکل زیر مقادیر ۴ و ۹ شکستگی نمودار هستند. (بخش ۱۶,۴,۱ از کتاب)



با توجه به محدودیت توان محاسباتی نمی‌توانیم تمام مقادیر k را مورد بررسی قرار دهیم. ۳۰ مقدار پیشنهادی برای k در نظر بگیرید و بازه‌ی قابل قبولی از تمام kهای ممکن و مفید را پوشش دهید. در گزارش خود توضیح دهید چرا این مقادیر را انتخاب کرده‌اید و نمودار RSS را برای تمام این مقادیر رسم کنید. سپس با توجه به نمودار RSS بهترین مقدار k را انتخاب کنید تا از آن خوشه‌بندی در موتور جستجو استفاده کنید.

بعد از انتخاب یک خوشه‌بندی مناسب، برای پاسخ‌گویی به یک پرسمان، ابتدا بردار ویژگی آن را همانند قبل استخراج کنید. سپس شباهت کسینوسی آن را با تمام مراکز خوشه‌ها (centroidها) محاسبه کرده و خوشه با بیشترین شباهت را انتخاب کنید. سپس شباهت کسینوسی بردار پرسمان با تمام سندهای آن خوشه را نیز محاسبه کرده و از میان آنها شبیه‌ترین سندها به پرسمان را انتخاب کنید و به عنوان نتیجه جستجو برگردانید. تمام تکنیک‌هایی که در مراحل قبل پروژه پیاده‌سازی کرده‌اید (همانند index elimination و ...) در این مرحله نیز قابل استفاده هستند. برای پاسخ‌گویی به پرسمان‌های ترکیبی با عملگرهای not ، and و عبارت نیز شاخص‌های معکوس مکانی را هم‌چنان نگه دارید.

توجه کنید لزومی بر اینکه فقط یک خوشه را برای جستجو انتخاب کنیم وجود ندارد. به این معنی که بعد محاسبه‌ی شباهت بردار پرسمان با مراکز خوشه‌ها، می‌توانیم b خوشه با بیشترین شباهت را انتخاب کرده و جستجو را در تمام اسناد آنها انجام دهید. این کار خصوصاً زمانی موثر است که تعداد خوشه‌ها زیاد باشد و در نتیجه احتمالا تعداد اسناد در یک خوشه کم شده باشد. انتخاب مقدار b و تعداد خوشه‌ها با هم مرتبط هستند و بهترین مقادیر آنها مقادیری است که یک تعادل بین سرعت پاسخ‌گویی و کیفیت نتایج است.

۵-۲- دسته‌بندی خبرها

همانطور که در ابتدای تعریف پروژه اعلام شد، موتور جستجوی طراحی شده می‌بایست قابلیت تعیین دسته خبر را در زمان وارد کردن پرسمان به کاربر بدهد. این قابلیت با استفاده از کلمه کلیدی cat ارائه می‌گردد. به عنوان مثال زمانی که کاربر عبارت «مهاجم cat:sport» را برای جستجو وارد می‌کند می‌بایست کلمه «مهاجم» در میان خبرهای ورزشی جستجو شود و در صورت ورود عبارت «مهاجم cat:politics» می‌بایست این کلمه در میان خبرهای سیاسی جستجو شود. برای این منظور با استفاده از روش‌های دسته‌بندی اسناد متنی ارائه شده در درس، دسته هر خبر را تعیین و ذخیره کنید تا در زمان جستجو بتوان از آن استفاده کرد. دسته‌های خبری مد نظر عبارتند از:

- Science: علمی و دانشگاهی (دانشگاهی، پژوهشی، علم و فناوری، آموزشی و ...)
- Culture-Art: فرهنگی و هنری (دین و اندیشه، ادبیات و کتاب، سینما و تئاتر، موسیقی، میراث باستانی، گردشگری)
- Politics: سیاسی (سیاست داخلی، مجلس، دولت، دفاعی امنیتی، حقوقی قضایی و ...)
- Economy: اقتصادی (اقتصاد کلان، تولید و تجارت، انرژی، عمران و اشتغال، قیمت ارز و طلا و خودرو و ...)
- Social: اجتماعی (جامعه، شهری، خانواده، آموزش و پرورش، محیط زیست، حوادث و انتظامی، سلامت، پزشکی، طب‌سنتی و ...)
- International: بین‌الملل (آسیا، خاورمیانه، غرب، فرامنطقه‌ای)
- Sport: اخبار ورزشی
- Multimedia: چندرسانه‌ای (عکس، ویدئو، گرافیک، صوت)

برای دسته‌بندی اسناد از الگوریتم‌های K نزدیک‌ترین همسایه با مقادیر مختلف K و روش بیز ساده استفاده کنید. برای این کار به داده‌ی برچسب‌خورده نیاز است. به تعداد مورد نیاز از اسناد (حداقل ۱۰۰۰ سند) را به صورت دستی برچسب بزنید (دسته‌ی آنها را مشخص کنید) و از آنها در مدل KNN برای مشخص کردن دسته‌ی یک سند تست استفاده کنید. این برچسب‌زنی توسط خود تیم توسعه دهنده پروژه انجام می‌شود.

سعی شده است که دسته‌ها جامع باشند، به این معنی که بتوان تمام اسناد را در یکی از این دسته‌ها جای داد. اما اگر لازم بود دسته‌ای اضافه شود یا یک دسته را به دو دسته‌ی کوچک‌تر شکسته شود، این کار را انجام دهید (اگر چه این کار توصیه نمی‌شود). اگر در برچسب‌زنی اسناد و تعریف دسته‌های مشخص شده دچار سردرگمی شدید به یکی از خبرگزاری‌های معتبر (مثل ایسنا) مراجعه کنید و زیردسته‌های خبری آن را بررسی کنید که چه نوع اخباری در آن زیردسته قرار گرفته‌اند و بر اساس آن دسته‌ی اسناد خودتان را مشخص کنید. بعد از برچسب‌زنی برای اینکه مطمئن شوید از تمام دسته‌ها داده‌ی برچسب‌خورده وجود دارد، تعداد سند موجود در هر دسته (در داده‌ی برچسب خورده) را محاسبه کرده و در گزارش خود ذکر کنید. وقتی از الگوریتم KNN استفاده می‌کنیم آیا تفاوت در تعداد داده‌ی برچسب‌خورده در دسته‌های مختلف مشکلی در دسته‌بندی ایجاد نمی‌کند؟ اگر یک دسته (برچسب) در میان داده‌های برچسب‌خورده خیلی زیاد باشد و از یک دسته داده‌ی کمی وجود داشته باشد، مشکلی ایجاد می‌شود؟

۵-۳- استخراج و نمایش اخبار مشابه در نتایج

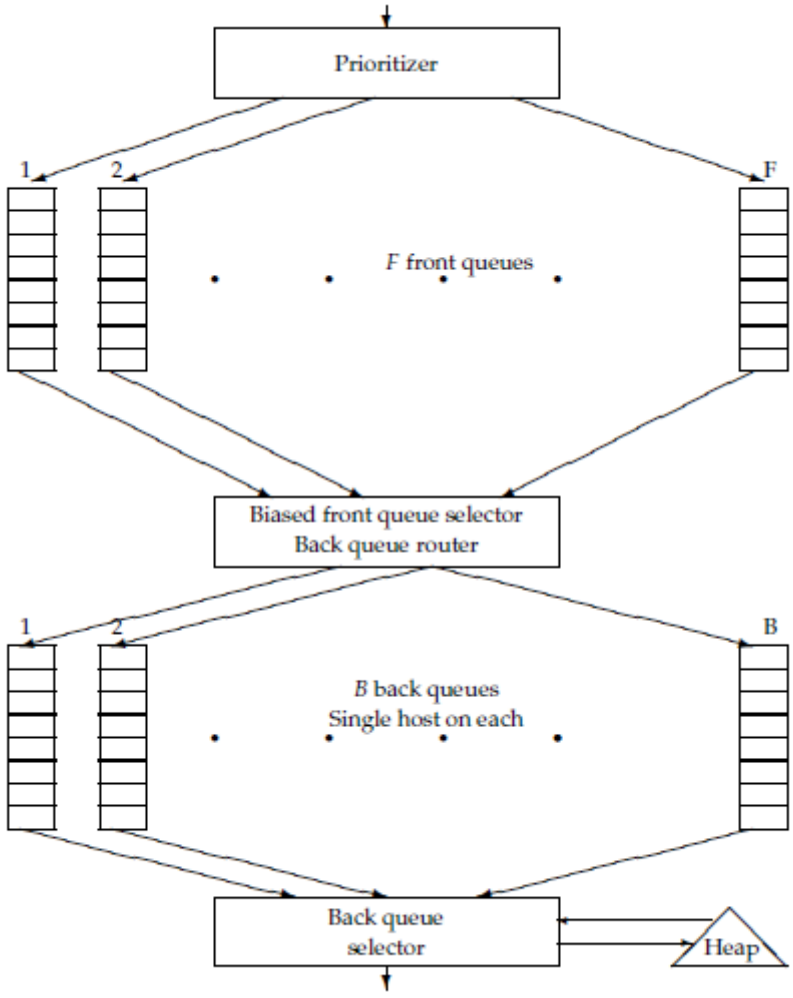
همچنین در هنگام پردازش اخبار می‌بایست اخبار مشابه با یک خبر نیز استخراج شوند و در نتایج ارائه شده به کاربر به صورت یکجا نمایش داده شود. به این صورت که در لیست نتایج اخبار و در صفحه‌ی خبر عنوان، تصویر، خلاصه و منبع خبر اصلی نمایش داده شده و عناوین و منبع سایر خبرهای مشابه نیز در زیر آن نمایش داده شود. خبرهای مشابه بر اساس محتوا و زمان انتشار آنها شناسایی می‌شوند. برای این منظور می‌توان از توابع شباهت معرفی شده در بخش خوشه‌بندی و آستانه‌گذاری روی آن استفاده کرد.

۵-۴- ارزیابی موتور جستجو

با استفاده از خوشه‌بندی و با انتخاب مقدار مناسب برای تعداد خوشه‌ها، سرعت پاسخ‌گویی به پرسمان بهبود می‌یابد. در مقابل ممکن است کیفیت نتایج جستجو تحت تاثیر خوشه‌بندی قرار گیرد. برای بررسی این موضوع ۱۰ پرسمان که انتظار دارید نتایج قابل پیش‌بینی داشته باشند را انتخاب کنید. نتایج این ۱۰ پرسمان را در حالت‌های بدون خوشه‌بندی و با خوشه‌بندی از نظر کیفیت نتایج و سرعت پاسخ‌گویی به پرسمان مورد مقایسه قرار دهید. ارزیابی کیفی موتور جستجو براساس ارتباط شهودی پرسمان با اسناد نتیجه انجام می‌شود و تعریف معیار عددی برای این کار نیاز نیست. فقط کافی‌ست به صورت شهودی نتایج را در حالت‌های مختلف با هم مقایسه کنید. سعی کنید پرسمان‌هایی که انتخاب می‌کنید هدفمند باشد به طوری که نتایج قابل پیش‌بینی داشته باشند.

۶- مرحله‌ی چهارم پروژه (امتیازی)

در این مرحله از پروژه می‌خواهیم یک خزنده‌ی وب به موتور جستجو اضافه کنیم و آن را به یک موتور جستجوی آنلاین تبدیل کنیم. با استفاده از این خزنده می‌خواهیم RSSهای خبرگزاری‌ها را بخوانیم، لینک اخبار را از آنها استخراج کنیم و محتوای آنها را خوانده و شاخص‌گذاری کنیم. برای این کار لینک‌های RSS را به ماژول اولویت‌بندی می‌دهیم که شامل صف‌هایی بر اساس معماری Mercator (شکل ۲۰,۳ کتاب) است. سپس بر اساس صف‌های front یک RSS را انتخاب می‌کنیم و تمام لینک‌های جدید موجود در آن را به صف‌های عقبی وارد می‌کنیم. با خواندن لینک‌های خبر از صف‌های عقبی اقدام به دریافت محتوای این لینکو استخراج محتوای متنی خبر، تاریخ انتشار خبر و اطلاعات مورد نیاز دیگر می‌کنیم. با این اطلاعات، خبر را در تمام بخش‌های موتور جستجو (شامل شاخص‌گذاری، فضای برداری، دسته‌بندی اخبار، استخراج اخبار مشابه و ...) اضافه می‌کنیم.



یک لیست ۲۰ تایی از خبرگزاری‌های مطرح تهیه کنید و آدرس تمام RSSهای آنها را در یک فایل (یا دیتابیس) ذخیره کنید. این لینک‌ها را به صورت متناوب بر اساس نرخ به روز شدن‌شان در یکی از صف‌های جلویی قرار دهید. با خواندن هر کدام از آنها از سر صفش با خواندن فایل RSS لینک‌های اخبار جدید آن را که باید در صف‌های عقبی به منظور خزش قرار داده شوند، استخراج کنید. توجه کنید که در هر بار خواندن فایل‌های RSS با مقایسه لینک‌های موجود در آن و لینک‌های خزش شده، لینک‌های جدید را تشخیص داده و فقط آنها را خزش کنید (duplicate url elimination). پس از استخراج لینک‌های

جدید از RSS با استفاده از جدول مسیریابی صفحه‌های عقبی (back) لینک‌های آن را به یکی از صف‌های عقبی منتقل کنید. با با در نظر گرفتن اطلاعات موجود در پشته (heap) برای هر یک از صف‌های عقبی، در زمان مناسب درخواست دریافت محتوای لینک خبر به سایت مقصد آن ارسال می‌شود. جزییات بیشتر این فرآیند در بخش ۲۰,۲ کتاب آمده است. توجه کنید که فرآیند توضیح داده شده در کتاب کمی با تعریف این مرحله از پروژه متفاوت است. در این مرحله از پروژه لینک‌های RSS در صف‌های جلویی قرار می‌گیرند و در زمان انتخاب باياس شده از صف‌های جلویی لینک‌های جدید موجود در هر RSS استخراج شده و لینک اخبار در صف‌های عقبی قرار می‌گیرند.

پس از دریافت محتوای خبر باید عنوان، خلاصه، متن، زمان انتشار خبر و سایر اطلاعات مورد نیاز برای پردازش از صفحه‌ی html خبر استخراج و ذخیره شوند. در ادامه محتوای خبر به ماژول شاخص‌گذاری موتور جستجو داده شود تا پس از عبور از ماژول‌های نرمال‌سازی و همسان‌سازی متن، استخراج کلمات، ریشه‌یابی، حذف کلمات پرتکرار و ... در نهایت به دیکشنری، شاخص معکوس مکانی و فضای برداری tfidf اسناد، خوشه‌بندی و ... اضافه شود. همچنین دسته‌بندی و استخراج اخبار مشابه نیز برای آن خبر انجام شود. این فرآیند باید طوری صورت بپذیرد که پس از شاخص‌گذاری خبر بتوان آن را در نتایج جستجو به طور کامل مشاهده کرد. این خبر ممکن است مستقیماً در نتیجه‌ی یک جستجو ظاهر شود یا در لیست اخبار مشابه یک خبر در نتیجه جستجو نمایش داده شود.

ماژول اولویت‌بندی وظیفه‌ی قراردادن فایل‌های RSS مختلف روی صف‌های جلویی را برعهده دارد. برای این کار نیاز است تخمینی برای میزان به‌روزشوندگی هر یک از لینک‌ها داشته باشد تا صف متناسب با آن را پیدا کرده و لینک را در آن صف قرار دهد. برای پیاده‌سازی این ماژول یک نرخ تولید خبر به هر منبع (هر لینک RSS یا هر خبرگزاری) اختصاص دهید. مقدار این نرخ نشان می‌دهد منبع مربوطه به صورت میانگین در هر چند دقیقه یک خبر تولید می‌کند. این مقدار را در ابتدا ۱۰ دقیقه قرار دهید و با خواندن هر بار RSS مربوط به آن، این مقدار را به روز کنید. ساده‌ترین راه به روزرسانی این است که در صورتی که منبع خبر جدیدی نداشت، نرخ را با مقدار ثابتی جمع کنید و در صورتی که خبر جدیدی در آن یافت شد یک مقدار ثابت از آن کم کنید. این کار را می‌توان به صورت هوشمندتری نیز انجام داد (مثلا اختلاف زمانی آخرین باری که این منبع را خوانده‌ایم تا کنون تقسیم بر تعداد اخبار جدید). فرمول پیشنهادی برای این به روز رسانی به شکل زیر است. نحوه کار این فرمول را در نظر بگیرید و مشکلات آن را در گزارش خود شرح دهید. برای حل مشکلات آن راه حل بهتری برای به روز رسانی این نرخ پیشنهاد کنید.

$$r^{new} = \begin{cases} r^{old} + 1, & n = 0 \\ r^{old}, & n = 1 \\ r^{old} - 1, & n > 1 \end{cases}$$

در این فرمول n نشان‌دهنده‌ی تعداد اخبار جدید در منبع است.

توجه کنید که این مرحله از پروژه را کاملا جداگانه از مرحله‌ی سوم پیاده‌سازی کنید. به این معنی که ابتدا فاز سوم پروژه را انجام داده و یک نسخه‌ی کارا از آن ذخیره کنید. سپس این مرحله را به عنوان نسخه دیگری از موتور جستجو توسعه دهید.

۷- نکات مهم

- پروژه به صورت گروهی انجام می‌شود.
  - گروه‌ها می‌توانند ۱ الی ۳ نفره باشند.
  - نمره کلیه اعضای گروه الزاما یکسان نخواهد بود.
  - هر یک از اعضای گروه می‌بایست بر تمام بخش‌های پروژه تسلط کامل داشته باشد.
- برای تحویل پروژه می‌بایست برنامه اجرایی به همراه گزارش کتبی تحویل داده شود. گزارش کتبی می‌بایست نحوه پیاده‌سازی کلیه قسمت‌های مدل بازیابی اطلاعات را مشخص کند.