دسته‌بندی متون

پروژه مبانی برنامه‌سازی

زمستان ۹۵

فاز ۲

در این فاز شما مانند فاز قبل تعدادی فایل RSS دریافت کرده و با پردازش آن‌ها را در حافظه خود ذخیره می‌کنید و علاوه بر آن یک پیش‌پردازش روی خبرها نیز انجام می‌دهید.

زمان‌بندی

مهلت انجام این فاز از ۱۶ تا ۳۰ دی است.

ساختار فایل‌ها

شما باید در این فاز تنها یک فایل برای انجام پیش‌پردازش روی خبرهایی که دریافت می‌کنید اضافه کنید. یعنی تمام کدهای این فاز شما در یک فایل خواهد بود و باید این فایل را بعلاوه دو فایلی که از فاز قبلی ساخته‌اید در کوئرا آپلود کنید.

به‌روز رسانی داده‌ها

در ابتدا باید مجموعه هایurl جدیدی که روی سایت قرار داده شده‌است را مانند فاز قبل با فرستادن یک درخواست GET به fop-project.ir/news/get-urls دریافت کنید فقط برای اینکه url های فاز ۲ را بگیرید باید در قسمت query-params درخواست خود phase را برابر ۲ قرار دهید.
برای روشن شدن موضوع به این نکته دقت کنید:شما در قسمت های قبل در قسمت data درخواست POST خود اطلاعاتی میفرستادید مثلا answer=salam & member=۱ یا یک رشته json برای مشخص کردن خبر. در درخواست GET قسمت data وجود ندارد و شما باید داده های خود را در قسمت query-params وارد کنید که بر خلاف درخواست post به انتهای url می‌چسبد. یعنی الان آدرسی که شما باید به آنcURL بزنید برابر است با www.fop-project.ir/?phase=۲ احتمالا در هنگام سرچ کردن با گوگل هم دیده اید گوگل عبارت شما را در انتهای url جستجو میچسباند و از همین روش استفاده میکند. درواقع با ظاهرشدن ؟ در انتهای url، قسمت query-params شروع میشود.url ها را دریافت و مانند فاز قبل فایلهای rss مربوط به هر خبر را مجددا دریافت و ذخیره کنید(دیگر نیازی به فرستادن اطلاعات پردازش شده به سایت نیست. url های خبر در این فاز جدید است و ربطی به فاز قبل ندارد).
پی نوشت: احتمالا الان متوجه شدید چرا درخواست های جتسجوی گوگل از نوع get است ولی وقتی مثلا در یک صفحه شما لاگین میکنید درخواست post میفرستد و رمز عبور و نام کاربری شما به انتهای url نمیچسبد! این ناشی از تفاوت فسلفی get و post است که اولی برای دریافت اطلاعات است و دیگری برای فرستادن اطلاعات و انجام یک کار یا تغییر روی سرور است.
برای ساده‌تر شدن کار شما فایل‌های RSS ساده‌تر شده‌اند و هر یک دربردارنده تقریبا ۱۰۰ خبر هستند. به تغییرات فایل‌های RSS دقت کنید. مثلا اکنون هر خبر تنها یک دسته‌بندی دارد.

دسته‌بندی‌های هدف

اگر کمی در فایل‌های RSS بگردید متوجه این نکته می‌شوید که تنها ۷ دسته‌بندی برای خبرها وجود دارد. هدف ما در این فاز این است که با انجام عملیات‌هایی روی خبرهای دسته بندی شده‌مان، بتوانیم به اطلاعاتی برسیم که بتوان با آن ها یک خبر که دسته‌بندی آن را نمیدانیم گرفته و بگوییم که دسته‌بندی آن چیست.
چیزهایی که می‌توان بیش از همه روی آن‌ها حساب کرد، توضیحات(description) و تیتر(title) هر خبر هستند که هر کدام شامل تعدادی کلمه هستند که به نوعی نماینده متن خبر محسوب می‌شوند. در

واقع ما قصد داریم تا با انجام پیش‌پردازش روی این بخش‌ها(تیتر و توضیحات) به اطلاعات مفیدی راجع به ارتباط کلمه‌ها و دسته‌بندی خبرها برسیم تا بتوان با بررسی کلمه‌ها موجود در تیتر و توضیحات یک خبر که دسته‌بندی آن را نمی‌دانیم دسته‌بندی آن را مشخص کرد.

استفاده از داده‌ساختارهای مختلف

برای این که برنامه شما در زمان معقولی اجرا شود، باید از داده‌ساختار های مختلف مانند درخت پیشوندی(که در سوال اول تمرین ۹تان نیز باید آن را پیاده‌سازی کنید) در مراحل آتی استفاده کنید(اگر نگاهی به شیوه کار این داده‌ساختار بیاندازید دلیل کاهش زیاد زمان اجرا هنگام استفاده از آن را می‌فهمید).
البته باز هم اجباری برای استفاده از این داده‌ساختار ندارید و می‌توانید از روش های دیگری مانند درهم‌سازی نیز استفاده کنید اما استفاده از ترای(همان درخت پیشوندی) به شما توصیه می‌شود! نیز فراموش نکنید که سرعت اجرای برنامه شما نیز یکی از پارامترهای نمره‌دهی به پروژه است.

مدل‌سازی و حل مسئله

مسئله ما این است که می‌خواهیم تعدادی خبر که دسته‌بندی آن‌ها را نمی‌دانیم گرفته و با بررسی توضیحات آن‌ها دسته‌بندی‌شان را مشخص کنیم. همان طور که قبلا هم گفتیم برای این کار یک پیش‌پردازش روی متن خبرهایی که دسته‌بندی آن ها را می‌دانیم انجام می‌دهیم تا به اطلاعاتی راجع به ارتباط بین کلمه‌ها موجود در متن و دسته‌بندی خبر دست‌یابیم.
(دقت کنید که چون به متن خبرها دست‌رسی نداریم با تیتر و توضیحات آن‌ها کار می‌کنیم.
در ادامه هر جا به کلمات موجود در خبر اشاره کردیم منظور کلمات تیتر و توضیحات خبر در فایل‌های RSS است.)
برای این کار هر خبر را مانند بسته‌ای از کلمه‌ها مدل می‌کنیم! یعنی فقط این اطلاعات را ذخیره می‌کنیم که چه کلمه‌های در تیتر و توضیحات این خبر با این دسته‌بندی آمده‌اند و هر کدام چند بار. تا بتوانیم با بررسی تعداد تکرار یک کلمه در خبرهای مربوط به یک دسته بندی، تعداد تکرار آن در خبرهای مربوط به بقیه دسته‌بندی ها و ... به اطلاعات مفیدی راجع به کلمه‌ها و ارتباطشان با دسته‌بندی‌ها برسیم.
البته در این مدل‌سازی اطلاعاتی از خبر مانند توالی کلمه‌ها از دست می‌رود اما در عوض میتوان بسیار ساده‌تر با کلمه‌ها موجود در هر خبر کار کرد.

بررسی ارتباط کلمه‌ها و دسته‌بندی‌ها

کلمه‌های هر خبر(کلمه‌ها موجود در تیتر و توضیحات خبر) هر کدام تا حدی مرتبط با موضوع و دسته‌بندی خبر هستند. مثلا اگر دسته‌بندی یک خبر business باشد، در تیتر و توضیحات آن خبر کلمه‌های مثل stocks و bank و sales وجود دارند که ارتباط زیادی با دسته‌بندی دارند و هم‌چنین کلمه‌های مانند for ، new و the وجود دارند که ربط خاصی به دسته بندی خبر ندارند. همین‌طور ممکن است نام یک کمپانی گمنام نیز در این کلمه‌ها آمده باشد و چون در جای دیگری نیامده کمکی به ما در دسته‌بندی سایر خبرها نمی‌کند. در مجموع به نظر می‌رسد که کلمه‌های هستند که در تشخیص دسته‌بندی یک خبر از بقیه مفید تر هستند و می‌توان با ملاک قرار دادن آن‌ها دسته‌بندی یک خبر را حدس زد. یعنی اگر یک خبر که دسته‌بندی آن را نمی‌دانیم به ما بدهند، می‌توانیم با بررسی بودن یا نبودن و تعداد تکرار آن کلمات در آن خبر حدس مناسبی درباره دسته‌بندی آن خبر بزنیم.
اگر بخواهیم علمی‌تر و عملی‌تر(!) صحبت کنیم ما n (که n یک متغیر است) کلمه‌ی موثر در تشخیص دسته‌بندی خبرها را پیدا می‌کنیم(توضیحات بیشتر در بند بعدی داده شده است). سپس هر خبر را با یک بردار به طول n مدل می‌کنیم، برداری که مؤلفه i ام آن نشان دهنده وزن کلمه i ام از n کلمه موثر یادشده

در خبر مورد نظر است. واضح است که دو خبر با موضوع مشابه برداری شبیه به هم دارند. سپس میتوان با مقایسه بردار یک خبر که دسته بندی آن را نمیدانیم با بردار هر دسته‌بندی(برای هر دسته‌بندی نیز می‌توان مشابها یک بردار تعریف کرد) میزان ارتباط آن خبر با دسته بندی را تخمین بزنیم.
اگر چیز زیادی از پاراگراف قبلی نفهمیدید، نگران نشوید! چون در این فاز تنها باید آن n کلمه مؤثر در تشخیص دسته‌بندی را پیدا کنید و بقیه کار مانند به دست آوردن وزن یک کلمه در یک خبر و به دست آوردن بردار هر دسته‌بندی در فاز بعد مفصل‌تر شرح داده‌شده و پیاده‌سازی می‌شود.

پیش‌پردازش

نکته:همان‌گونه که قبلا نیز اشاره کردیم در ادامه هر جا صحبت از وجود یک کلمه در یک خبر بود منظور همان وجود آن در تیتر و توضیحات مربوط به هر خبر در فایل‌های RSS است.
همان‌طور که گفته شد باید در ابتدابا انجام عملیاتی کلمه‌ها تیتر و توضیحات خبر ها را در داده‌ساختارهایی که طراحی کرده‌اید ذخیره کنید. برای انجام این کار ابتدا علامت‌های نگارشی و حروف غیر از حروف الفبای انگلیسی را از تیتر و توضیحات هر خبر حذف کرده و در ادامه تمامی بزرگ را به حروف کوچک زبان انگلیسی تبدیل کنید تا بتوان بهتر و ساده‌تر آن‌ها را در داده‌ساختارها ذخیره کرد.
حال چگونه باید n کلمه مؤثر در تشخیص دسته‌بندی ها را پیدا کرد و اصلا این عدد n باید چند باشد؟ در واقع عدد n نیز یکی از پارامترهای مسئله است و می‌توانید با تغییر دادن آن و بررسی کلمه‌ها به دست آمده و نتایج قسمت‌های بعد، آن را تغییر داده و بهتر کنید(فراموش نکنید که در طی مراحل آتی از داده ساختارها که در اول متن فاز نیز به آن اشاره کرده بودیم استفاده کنید...).
برای پیدا کردن این کلمه‌ها باید در ابتدا کل کلمه‌ها موجود در تیتر و توضیحات خبرها را بررسی کنیم. با انجام سه گام زیر روی کلمه‌ها به n کلمه موثر مورد نیاز می‌رسیم:

- حذف کلمه‌ها بسیار کم‌تکرار: گر یک کلمه در تعداد بسیار کمی از خبرها آمده باشد ملاک معتبری برای مقایسه محسوب نمی‌شود. مثلا اگر ۲۰۰۰ خبر داشته باشید. کلمه‌ای که کم‌تر از ۳ بار تکرار شده باشد اهمیت چندانی ندارد. می‌توانید با تغییر کران‌بالای تعداد تکرار(عدد ۳ در این مثال) و بررسی خروجی مقدار مناسب آن را پیدا کنید.
- حذف کلمه‌ها بسیار پرتکرار: همینطور اگر یک کلمه در تعداد بسیار زیادی از خبرها آمده باشد(مثلا در بیش از سه‌چهارم خبرها)، نمی‌تواند ملاک خوبی برای مقایسه باشد. این کار برای حذف کلمه‌های مانند the و and و for انجام می‌شود. می‌توانید با تغییر کران‌پایین این تعداد و بررسی این که چه کلمه‌های حذف می‌شوند مقدار مناسب آن را پیدا کنیدس.
- انتخاب n کلمه با تاثیر بیشتر در بین کلمه‌ها باقی‌مانده (کلماتی که وجود یا عدم وجودشان در یک خبر تاثیر زیادی در حدس ما نسبت به دسته‌بندی آن خبر دارد).

در نظر داشته‌باشید که تغییر هر یک از کران‌ها و متغیرهای بالا تاثیرات مثبت و منفی خاص خودش را دارد و باید با تغییر دادن مقادیر آن‌ها و بررسی خروجی به مقدار بهینه آن‌ها برسید. حال باید معیاری برای مقایسه تاثیر یک کلمه در مشخص کردن دسته‌بندی یک خبر تعیین کنیم. یعنی تابعی طراحی کنیم که با قرار دادن کلمه در آن تابع آن تابع به ما عددی بدهد که نشان‌دهنده اهمیت آن کلمه در زمینه تعیین دسته بندی باشد(دقت کنید که این تابع باید به طور کلی اهمیت یک کلمه را در تعیین دسته‌بندی خبر مشخص کند، نه این که فقط اهمیت آن کلمه را نسبت به یک دسته‌بندی معین مشخص تعیین کند. یعنی شما باید در تابع خود به نحوی نسبت کلمه دریافتی را با تمام دسته‌بندی ها مشخص کنید، نه فقط یک دسته‌بندی).
طراحی این تابع به عهده شماست! شما باید با بررسی این که چه عواملی وجود داشتن یا نداشتن یک کلمه در یک خبر را برای تعیین دسته بندی آن مهم می‌کنند و بررسی آن عوامل در خبرهایی که به شما

داده‌شده است اهمیت کلمه ها را تعیین کنید و n عدد از مهم‌ترین آن‌ها را نیز بیابید. اما برای راهنمایی به فاکتورهای زیر درباره تابع مد نظر که یک کلمه را به عنوان ورودی می‌گیرد توجه کنید:

- تعداد خبرهای هر دسته‌بندی.
- تعداد خبرها با دسته‌بندی c که کلمه ورودی در آن‌ها هست، به ازای هر دسته‌بندی c موجود. همین‌طور تعداد خبرهایی که دسته‌بندی آن‌ها c است و کلمه ورودی در آن ها هست نسبت به تعداد کل خبرهایی که کلمه w در آن‌ها هست، باز هم به ازای هر دسته‌بندی c موجود.
- تعداد خبرها با دسته‌بندی c که کلمه ورودی در آن‌ها نیست، به ازای هر دسته‌بندی c موجود. همین‌طور تعداد خبرهایی که دسته‌بندی آن‌ها c است و کلمه ورودی در آن ها نیست نسبت به تعداد کل خبرهایی که کلمه w در آن‌ها نیست، باز هم به ازای هر دسته‌بندی c موجود.

می‌توانید این راهنمایی‌ها را به عنوان سرنخ داشته باشید، از خلاقیت خودتان استفاده کنید و با بررسی نتیجه‌ای که به ازای استفاده از تابعی که طراحی کرده‌اید به دست می‌آید(n کلمه‌ی نهایی) تابع مناسب را برای انتخاب کلمات موثر بیابید! می‌توانید این نکته را هم در نظر داشته باشید که چون n کلمه‌ی موثر برای تعیین دسته‌بندی خبر های ورودی استفاده می‌شوند باید در آن‌ها چند کلمه کلیدی مربوط به هر کدام از دسته‌بندی‌ها باشد...