Web-based Personalized Hybrid Book Recommendation System

۱ مقدمه

در این مقاله سعی شده مدلی از رکامندر سیستم هایبرید بر پایه وب برای معرفی کتاب، نشان داده شود. این سیستم با بررسی ابعاد مختلف پیشنهاد کتاب، کارایی بهتر از دو سیستم تک بعدی collaborative و content based دارد.

۲ سیستم پیشنهاد شده

این سیستم از ۳ بخش سرورمرکزی، و انبارداده و اینترنت تشکیل شده‌است.

در سرورمرکزی تمام عملیات‌های محاسباتی انجام خواهندشد. این سرور به یک انبارداده‌ای متصل است که شامل ۲ جدول است. در جدول اول اطلاعات یوزر و ویژگی‌های demographic مربوط به آن، در جدول دوم اطلاعات کتاب‌ها مانند اسم، عکس و ...، در جدول سوم اطلاعات بدست‌آمده از web scraping و در جدول چهارم امتیازهای افراد به کتاب‌ها قرار دارند. جدول چهارم شامل ویژگی مهمی به نام timestamp هست که به سیستم کمک میکند تا کتاب‌های قدیمی را از دور خارج کند و همگام با تغییر سلایق مردم پیش برود.

در بک‌اند این سیستم میتوان از Apache Mahout و یا MySQL استفاده کرد. درصورت استفاده از Mahout، با استفاده از ویژگی Hadoop آن، میتوان در ابعاد بزرگتری نسبت به MySQL کار کرد اما سرعت دسترسی آن نسبت به MySQL بسیار کمتر است.

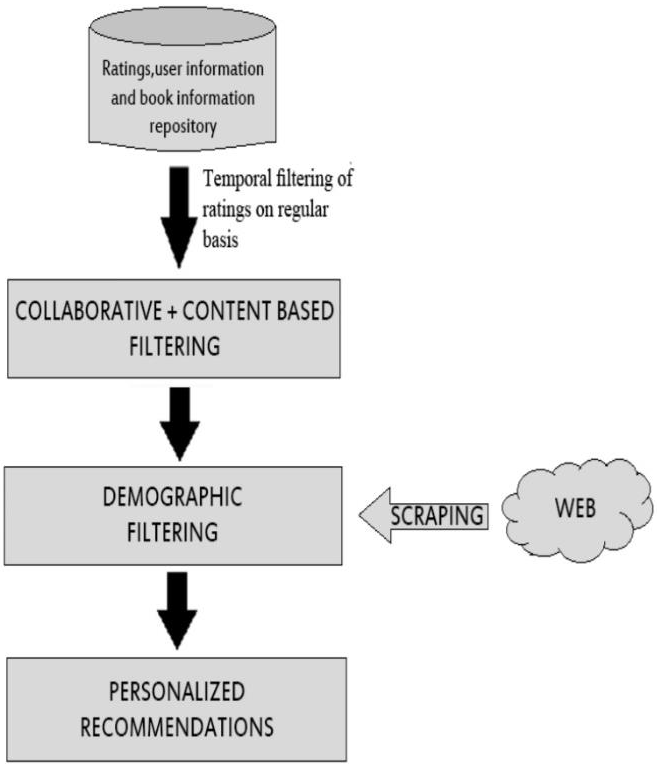
از متدهای فیلترینگ demographic،collaborative item-item, user-user و content-based در ساخت این سیستم استفاده شده.

میتوانید در شکل ۲-۱، ساختار و روند کار این سیستم را مشاهده نمایید.

۳ متدهای فیلترینگ

در این سیستم، از ترکیبی از ۳ متد فیلترینگ collaborative، content based و demographic استفاده شده.

ابتدا به معرفی این ۳ متد در سیستم پیشنهاد شده میپردازیم.



شکل۲-۱: روند کار سیستم پیشنهادی

۳-۱ Collaborative-based filtering

در Apache Mahout کلاس‌های PearsonCorrealtionSimilarity و GenericUserBasedRecommender وجود دارند که در پیاده‌سازی مقادیر شباهت user-user و item-item به ما کمک میکنند.

۳-۲ Content-based filtering

در این متد علاوه بر نمرات داده‌شده توسط کاربران، از اطلاعات دیگری مانند نویسنده، ژانر، هزینه، تعداد صفحات و ... نیز برای پیدا کردن بهترین پیشنهادات استفاده میشود.

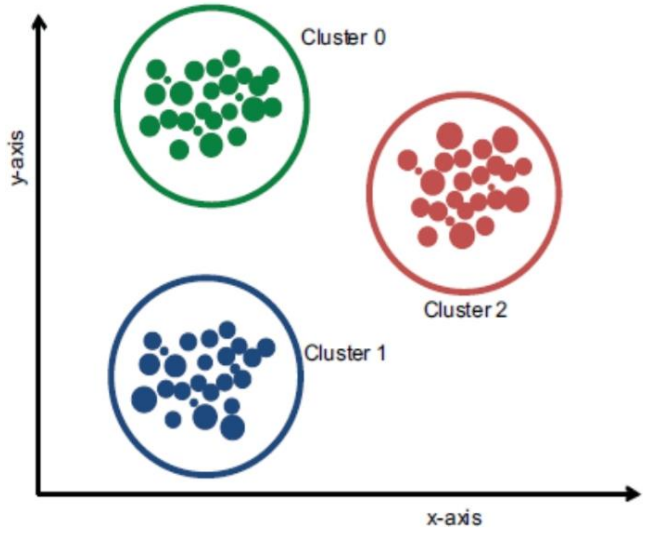
مهمترین قدم در این بخش، پیدا کردن متد یادگیری است زیرا علاوه بر بازده، به دلیل بزرگی داده‌ها، باید به فاکتورهایی مانند زمان و فضا توجه کرد.

در Apache Mahout، کلاسی برای پیاده‌سازی Content-based filtering وجود ندارد اما کلاس‌هایی وجو دارند که میتوانند برای پیاده‌سازی الگوریتم‌های موردنیاز، به ما کمک کنند.

۳-۳ Demographic-based filtering

تکنیک کلاسترینگ میتواند در جهت افزایش بهره‌وری، به سیستم کمک کند. در این تکنیک، به جای استفاده از همه یوزرها به عنوان training set، یوزرها را بر اساس ویژگی‌هایی مانند سن، جنسیت و ... به دسته‌هایی تبدیل میکند و سپس از این دسته‌ها به عنوان training set استفاده میشود. در شکل ۳-۱ میتوانید ساختار گرافیکی این تکنیک را ببینید.

در Apache Mahout، کلاس‌هایی مانند SequenceFile، EuclideanDistanceMeasure و ManhattanDistanceMeasure وجود دارند که به ما در پیاده سازی این متد کمک میکنند.

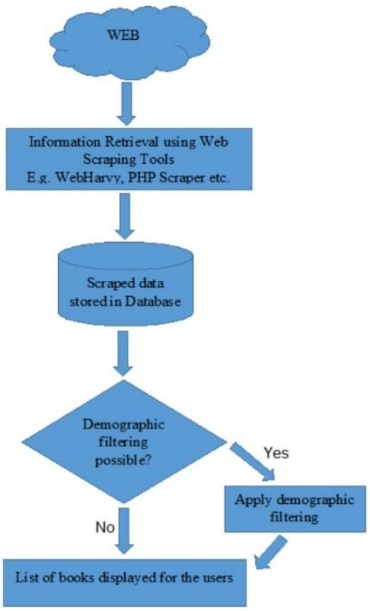


شکل۳-۱: تکنیک کلاسترینگ

۴ Web Scraping

به عمل جمع‌آوری اطلاعات از سطح اینترنت Web Scraping گفته میشود. در شکل ۴-۱ میتوانید فلوچارت عملیات Web Scraping را مشاهده نمایید.

وبسایت‌های فعال در حوزه E-Commerce میتوانند در سیستم، به ما کمک زیادی بکنند. به عنوان مثال با انجام Web Scraping بر روی وبسایت آمازون، میتوان اطلاعاتی از قبیل پرفروش‌ترین کتاب‌ها، کتاب‌ها با بیشترین امتیاز و ... را به دست آورد و برای حل مشکلاتی از قبیل cold start و gray sheep users، از آنها استفاده کرد.



شکل۴-۱: فلوچارت web scraping

۴ اعتماد

این بخش یکی از مهمترین بخش‌های سیستم پیشنهادی است زیرا در صورتی که کاربر به سیستم اعتماد نداشته باشد، دیگر ابعاد سیستم بی‌فایده خواهند بود. برای این موضوع میتوان در صفحه‌ای از وبسایتمان، متن یا ویدیویی با موضوع روش کار این سیستم در اختیار کاربران بگذاریم تا با درک روند کار سیستم، اعتمادشان افزایش یابد.

روش دیگر استفاده از mirror behavior است. در این روش، کاربران میتوانند پروفایل‌های دیگر کاربران را مشاهده نمایند و کاربران مشابه خود را تشخیص دهند. این روش باید بصورت ۲طرفه انجام شود و درصورتی که کاربری اجازه مشاهده پروفایلش را به دیگران ندهد، خود نیز دسترسی به پروفایل دیگران نداشته باشد.

در صورت افزایش امنیت سیستم، کاربران به سیستم راحتتر اعتماد میکنند. برای جلوگیری از حملاتی مانند نظرها و امتیازات جعلی، میتوان از متدهایی مانند کپچا، بررسی آیپی کاربر و ... استفاده کرد.