به نام خدا

گزارش کار پروژه درس داده کاوی

موضوع :

سیستم های توصیه گر

تهیه کنندگان:

علی نجفی، آراز قلی پور شیلابین

استاد مربوطه:

دکتر مریم شعاران

دانشگاه سراسری تبریز

چکیده

با توجه به رشد سریع سیستم های اطلاعات و وجود حجم زیادی از اطلاعات در شبکه جهانی نیاز به سیستم های که شبکه جهانی را برای کاربر سفارش کننده احساس شده است از طرفی نیاز روزافزون کاربران به وب مفهومی نیز بر اهمیت این موضوع افزوده است بنابراین سیستم هایی توصیه گر به عنوان راه حلی برای این معضل معرفی شدند این سیستم ها به طور هوشمند با استفاده از تکنیک های هوش مصنوعی به شناسایی علایق کاربران خود در فضای اینترنت پرداخت و پیشنهادی متناسب با اولویت ها و محدودیت های فعلی به کاربر ارایه می دهند. سیستم های توصیه گر برای حل مشکل سربار اطلاعاتی در اینترنت به وجود آمده اند. سیستم های توصیه گر از دانش علایق کاربر که از گذشته گردش وی در وب به دست آمده برای پیدا کردن کالاها یا صفحات وب مورد علاقه وی استفاده می کند. در این بخش به بررسی collaborative filtering که خود به روش های user-base و item-base و content-base تقسیم میشود، می پردازیم و از پایگاه داده MovieLens استفاده میکنیم.

بررسی پایگاه داده

پایگاه داده MovieLens شامل یک ۶۰۴۰ کاربر میباشد که در آن کاربران به فیلم هایی که قبلا مشاهده نموده اند که تعداد آن ها ۳۹۵۲ می باشد، امتیاز بین صفر تا ۵ را داده اند. به طول مثال :

	userId	movieId	rating
0	1	1	4.0
1	1	3	4.0
2	1	6	4.0
3	1	47	5.0
4	1	50	5.0
5	1	70	3.0

کاربر شماره ۱ به فیلم های ۱، ۳، ۶ امتیاز ۴ داده است.

همچنین هر فیلم در ۱۶ ژانر یا سبک مختلف دسته بندی شده اند که نمونه ای را در شکل زیر می توان مشاهده کرد.

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy

تقسیم بندی داده های آموزش و آزمون

برای مطمین شدن از عملکرد درست سیستم توصیه گر ۵ درصد از داده برای داده آزمون جدا کرده ایم تا عملکرد سیستم را روی این داده ها که در زمان آموزش آن ها را ندیده است را براورد کنیم.

سیستم های توصیه گر مبتنی بر Collaborative Filtering

پالایش اطلاعات یا الگوهای موجود در داده توسط همکاری چند عامل، صورت از داده یا چند داده مختلف و غیره را پالایش گروهی می‌نامند. انگیزه پیدایش پالایش گروهی، بر اساس این ایده است که معمولاً مردم بهترین پیشنهادها را از کسانی دریافت می‌کنند که مشابه سلیقه خودشان باشند. پالایش گروهی به کاوش روش‌هایی می‌پردازد که افراد باعلاقه‌های یکسان را باهم تطابق می‌دهند و بر این اساس به ارائه توصیه می‌پردازد. این روش با استفاده از نظرات دیگر کاربران که علاقه‌مندی‌های خود را به اشتراک گذاشته‌اند، به مردم کمک می‌کند تا انتخاب‌های خود را انجام دهند. این بدین معنا است که مدل‌های مبتنی بر پالایش گروهی، در حقیقت از قدرت نظرات کاربران بر اقلام برای ایجاد توصیه استفاده می‌کنند. اصلی‌ترین چالش موجود در سیستم‌های پالایش گروهی مسئله خلوت بودن ماتریس نظرات کاربران است. پالایش گروهی خود به سه روش تقسیم می شود که به ترتیب عبارت اند از : کاربرمحور، آیتم‌محور ، محتوامحور.

پالایش گروهی کاربر محور

در این روش سعی بر این است که کاربر ها با سلایق یکسان شناسایی شده و از روی آن توصیه مناسب برای کاربر انجام شود.

داده ای که از آن در این روش استفاده می شود به صورت ماتریس (Z) امتیاز دهی می باشد که ابعاد این ماتریس به صورت N x M می باشد که در آن N تعداد کاربران و M تعداد فیلم ها است. به این صورت که Z[10][1] امتیاز کاربر ۱۱ ام که به فیلم دوم داده است را نشان میدهد.

ما از الگوریتم KNN یا K امین نزدیک ترین همسایه برای شناسایی کاربر های شبیه به هم استفاده کرده ایم که در این الگوریتم از معیار cosine similarity برای محاسبه نزدیک ترین همسایه ها استفاده شده است که فرمول آن به صورت زیر است:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

به این ترتیب شبیه ترین K کاربر به کاربر U انتخاب می شوند.

حال هدف این است که میخواهیم میزان امتیازی که کاربر u به فیلم m می دهد را تخمین بزنیم که برای این کار ابتدا کاربر u که برداری از امتیازات این کاربر به فیلم هایی که دیده است را داده است را به مدل KNN ورودی میدهیم تا K کاربری که بیشترین شباهت را دارند و فیلم m را دیده اند را پیدا کنیم.

در این جا K کاربر که به کاربر u نزدیک هستند و به فیلم m هم امتیاز داده اند پیدا شده است و میزان شباهت هر کاربر به کاربر u نیز در دسترس میباشد حال برای محاسبه امتیاز تخمینی که کاربر u به فیلم m می دهد از فرمول امتیاز وزن دار استفاده میکنیم که به صورت زیر است:

$$score(u,m) = \frac{\sum_{j \in U_k} sim(u,j) \times r(j,m)}{\sum_{j \in U_k} sim(u,j)}$$

U_k ، مجموعه k نزدیک ترین کاربران به کاربر u می باشد که به فیلم m امتیاز داده اند.

$r(j,m)$ ، امتیازی است که کاربر j به فیلم m داده است.

$sim(u,j)$، میزان شباهت کاربر u به کاربر j می باشد.

پالایش گروهی آیتم محور

در این روش سعی بر این است که آیتم هایی که شباهت یکسانی دارند شناسایی شده و بر حسب سلیقه کاربر که قبلا چندین آیتم را امتیاز دهی کرده است آیتم جدید را توصیه کنیم.

داده ای که از آن در این روش استفاده می شود به صورت ماتریس (Z) امتیاز دهی می باشد که ابعاد این ماتریس به صورت M x N می باشد که در آن N تعداد کاربران و M تعداد فیلم ها است. به این صورت که Z[10][1] امتیاز کاربر دوم که به فیلم ۱۱ ام داده است را نشان میدهد.

ما از الگوریتم KNN یا K امین نزدیک ترین همسایه برای شناسایی آیتم های شبیه به هم استفاده کرده ایم که در این الگوریتم از معیار cosine similarity برای محاسبه نزدیک ترین همسایه ها استفاده شده است که فرمول آن به صورت زیر است:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

به این ترتیب شبیه ترین K فیلم به فیلم m انتخاب می شوند.

حال هدف این است که میخواهیم میزان امتیازی که کاربر u به فیلم m می دهد را تخمین بزنیم که برای این کار ابتدا فیلم m که برداری از امتیازات این فیلم را که سایر کاربران به آن داده اند را به مدل KNN ورودی میدهیم تا K فیلمی که بیشترین شباهت را دارند را پیدا کنیم..

در این جا K فیلم که به فیلم m نزدیک هستند و به کاربر u هم به آن ها امتیاز داده است پیدا شده است و میزان شباهت هر فیلم به فیلم m نیز در دسترس میباشد حال برای محاسبه امتیاز تخمینی که کاربر u به فیلم m می دهد از فرمول امتیاز وزن دار استفاده میکنیم که به صورت زیر است:

$$score(u,m) = \frac{\sum_{j \in M_k} sim(m,j) \times r(u,j)}{\sum_{j \in M_k} sim(m,j)}$$

M_k ، مجموعه k نزدیک ترین فیلم ها به فیلم m می باشد که کاربر u به آن ها امتیاز داده است.

$r(u,j)$ ، امتیازی است که کاربر u به فیلم j داده است.

$sim(m,j)$، میزان شباهت فیلم m به فیلم j می باشد.

پالایش گروهی محتوا محور

در این روش سعی بر این است که آیتم هایی که شباهت یکسانی دارند شناسایی شده و بر حسب سلیقه کاربر که قبلا چندین آیتم را امتیاز دهی کرده است آیتم جدید را توصیه کنیم. تفاوت این روش با روش آیتم محور در این است که به جای استفاده از ماتریس امتیازی برای پیدا کردن نزدیک ترین فیلم از ژانر فیلم ها استفاده می شود و برحسب ژانر فیلم ها فیلم ها دسته بندی میشوند.

داده ای که از آن در این روش استفاده می شود به صورت ماتریس (Z) می باشد که ابعاد این ماتریس به صورت M x G می باشد که در آن M تعداد فیلم ها و G تعداد ژانرها است. به این صورت که Z[10][1] مقدار صفر یا یک میباشد. اگر یک باشد به این معنا است که فیلم یازدهم سبک یا ژانر دوم را دارا میباشد در غیراینصورت به آن سبک یا ژانر تعلق ندارد.

ما از الگوریتم KNN یا K امین نزدیک ترین همسایه برای شناسایی آیتم های شبیه به هم استفاده کرده ایم که در این الگوریتم از معیار cosine similarity برای محاسبه نزدیک ترین همسایه ها استفاده شده است که فرمول آن به صورت زیر است:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

به این ترتیب شبیه ترین K فیلم به فیلم m انتخاب می شوند.

حال هدف این است که میخواهیم میزان امتیازی که کاربر u به فیلم m می دهد را تخمین بزنیم که برای این کار ابتدا فیلم m که برداری از امتیازات این فیلم را که سایر کاربران به آن داده اند را به مدل KNN ورودی میدهیم تا K فیلمی که بیشترین شباهت را دارند را پیدا کنیم.

در این جا K فیلم که به فیلم m نزدیک هستند و به کاربر u هم به آن ها امتیاز داده است پیدا شده است و میزان شباهت هر فیلم به فیلم m نیز در دسترس میباشد حال برای محاسبه امتیاز تخمینی که کاربر u به فیلم m می دهد از فرمول امتیاز وزن دار استفاده میکنیم که به صورت زیر است:

$$score(u,m) = \frac{\sum_{j \in M_k} sim(m,j) \times r(u,j)}{\sum_{j \in M_k} sim(m,j)}$$

M_k ، مجموعه k نزدیک ترین فیلم ها به فیلم m می باشد که کاربر u به آن ها امتیاز داده است.

$r(u,j)$ ، امتیازی است که کاربر u به فیلم j داده است.

$sim(m,j)$، میزان شباهت فیلم m به فیلم j می باشد.

نتایج

ما برای محاسبه خطای سیستم از تابع خطای منهتن استفاده کرده ایم که فرمول این تابع به صورت زیر می باشد.

$$L(x, \hat{x}) = |x - \hat{x}|$$

$$ML = \frac{1}{m}\sum_{i} L(x_i, \hat{x}_i)$$

$\hat{x}$ مقدار تخمین زده شده ، x مقدار واقعی و m تعداد داده های آزمون می باشد.

نتایج به صورت زیر می باشد.

K	User-base	Item-base	Content-base
10	0.893	0.856	0.873
20	0.869	0.831	0.870
50	0.810	0.781	0.869

با توجه به نتایج به دست آمده کاملا واضح است که با افزایش K میزان خطا در هر سه روش کاهش پیدا میکند و در بین سه روش بحث شده item-base بهترین نتیجه را به ارمغان می آورد. از دلایل بهتر عمل کردن این روش می توان به ثابت بودن آیتم ها اشاره کرد چرا که هر فیلم در نهایت فیلم می باشد در حالی که در روش کاربرمحور ممکن است علایق کاربر تغییر پیدا کند.