Recommender Systems

سیستم های پیشنهاد دهنده

Salim.Namvar@Gmail.Com

Salim Namvar

2016 , August

به نام خدا

پروژه پایانی: **RECOMMENDER SYSTEMS**

درس : **داده کاوی**

استاد : **دکتر وحید صیدی قمشه**

دانشجو : **سلیم نامور**

فهرست

۱– مقدمه :

همان طور که همه می دانیم، امروزه مبحث سیستم های پیشنهاد دهنده یا همان Recommender Systems (RecSys) یکی از موضوعات مهم در دنیای داده کاوی می باشد. به عنوان مثال شرکت های تبلیغاتی که با گسترش بستر فضای مجازی اکثر رویکرد های خود را معطوف به این حوزه داشته اند، با استفاده از سیستم های پیشنهاد دهنده به موفقعیت های عظیمی دست یافته اند. این روند به آرامی به عنوان یکی از الزامات مهم وب سایت های بزرگ در اینترنت شناخته می شوند که باعث گشته شرکت های بزرگ جهت پیشبرد اهداف خود همچون فروش کالای بیش تر، ارائه خدمات بیش تر و ... به مشتریان، در این زمینه هزینه های زیادی را صرف کنند.

به عنوان یک مثال ساده از کاربرد سیستم های پیشنهاد دهنده می توان به این موضوع اشاره کرد که شرکت های تبلیغاتی در بستر اینترنت از RecSys استفاده می کنند تا اجناسی که مشتریان ممکن است به آن کالا ها علاقه داشته باشند را به آن ها پیشنهاد دهند. این اجناس می توانند انواع مختلف کالاهای مصرفی افراد که روزانه در این وب سایت ها خرید می کنند باشد. یکی از این نمونه وب سایت ها، NetFlix می باشد که فیلم ها را بر اساس ویژگی های که دارند به کاربران پیشنهاد می دهند. یا می توان به وب سایت Amazon اشاره کرد که کتاب های مورد علاقه کاربران را به آن ها پیشنهاد می دهد.

راه های مختلفی برای پیشنهاد محصولات وجود دارند. از جمله : از کالاهای که بیش ترین خرید را شامل می شوند استفاده کنیم و آن ها را به کاربران پیشنهاد دهیم، یا می توان پیشنهاد دادن را بر اساس ویژگی کالا ها انجام داد، و همچنین می توان این کار را بر اساس خرید های قبلی که کاربر انجام داده است نیز انجام داد. در بین همه این راه حل ها "پالایش گروهی" یا همان Collaborative Filtering (CF) یکی از بهترین تکنیک ها در این زمینه می باشد که توسط [گولد برگ و دیگران در سال ۱۹۹۲] ارائه شده است.

اساسا سیستم های پیشنهاد دهنده را می توان در سه نوع مختلف دسته بندی کرد: پالایش گروهی (CF)، سیستم های پیشنهاده دهنده مبتی بر محتوا (Content-based) و رویکرد مختلط یا همان Hybrid که تلفیقی از این دو روش می باشد. سیستم های پیشنهاده دهنده مبتی بر محتوا از طلاعات پروفایل کاربران و کالا ها استفاده می کند. به همین دلیل این روش در بعضی موارد همچون پیشنهاد دادن اقلام مولتی مدیا همانند فیلم و صوت یک چالش سختی می باشد تا اطلاعات آن ها را بدست آوریم.

در رویکرد پالایش گروهی (CF) کلیدی ترین موضوع اقدامات کاربر که قبلا با سیستم انجام داده است می باشد. این روش CF از رای های که کاربران قبلا به کالا ها داده اند استفاده می کند تا رای های کاربران جدید را به اقلام مختلف را پیش بینی کند. ایده ی که در پشت این روش نهفته است این می باشد که دو کاربری که قبلا

اقلام مشابه را خریده اند یا رای داده اند در آینده نیز ممکن است آن دو کاربر اقلام مشابه را مورد استقبال قرار دهند.

در این پروژه نیز ما از روش پالایش گروهی یا همان Collaborative Filtering برای پیشنهاد محصولات به کاربران استفاده خواهیم کرد. در ابتدا در اقلام را بر اساس اطلاعاتی که در پروفایلشان وجود دارد به کمک روش خوشه بندی فازی، گروه بندی می کنیم. سپس از روش پالایش گروهی استفاده کرده تا رای های کاربران نسبت به اقلام مختلف را پیش بینی کنیم.

۲– الگوریتم پیشنهادی :

الگوریتم پیشنهادی ما در دو فاز خلاصه می شود:

- فاز اول : خوشه بندی (Clustering) : در این فاز، اقلام موجود بر اساس اطلاعات پروفایلشان خوشه بندی می شوند.
- فاز دوم : پالایش گروهی مبتنی بر اقلام (Item-Based Collaborative Filtering) : در این فاز، روش پالایش گروهی بر روی هر خوشه اعمال می گردد تا رای های کاربران به اقلام مختلف پیش بینی شوند.

دید اجمالی بر روی الگوریتم:

- مرحله اول : خوشه بندی بر روی تمامی ایتم ها اعمال می شود.
- مرحله دوم : روش پالایش گروهی بر روی هر خوشه اعمال گردیده تا رای های کاربران جدید در ماتریس کاربر-ایتم پیش بینی شوند.
- مرحله سوم : در این مرحله برای جلوگیری از مشکل Cold Start راه حل پیشنهادی انجام می شود. این مشکل دارای دو حالت کلی است :
  - زمانی که کاربر جدیدی به سیستم اضافه می شود : کاربران جدید می بایست تا یک آستانه در نظر گرفته شده به اقلام مختلف رای دهند تا اقلام پیشنهادی را دریافت کنند. این آستانه یا Threshold را باید تعیین نمود.
  - زمانی که ایتم جدیدی به سیستم اضافه می شود : رای ایتم جدید NI توسط کاربر U برابر با NI(U) می باشد که مشخص کننده میانگین رای های کاربر U در آن خوشه مورد نظر می باشد.

خوشه بندی تنها فاز پیش پردازش می باشد. به صورتی که این الگوریتم می بایست در دوره های متناوب و منظم پس از اضافه شدن تعدادی مشخصی از ایتم های جدید به سیستم، اجرا شود.

۲–۱– فاز اول – خوشه بندی :

۲–۱–۱– خوشه بندی K-Means :

برای یادآوری ابتدا به بررسی و توضیح نحوه کارکرد این نوع خوشه بندی پرداخته و پس از آن این روش را در با یک مثال بر روی مجموعه داده های مقاله به صورت واضح تر بیان خواهیم کرد. اگر فرض کنیم یک سری داده به صورت داریم و می خواهیم آن ها را خوشه بندی کنیم :

$$X = (x_1, x_2, x_3, \dots, x_n) \in R^d$$

یک سری مراکز خوشه باید بدست آوریم که به عنوان مجهولات مسائله هستند.

$$C = (c_1, c_2, c_3, \dots, c_k) \in R^d$$

در حالت کلی هر x_i به خوشه ی مربوط هست

$$A = (A_1, A_2, A_3, \dots, A_k)$$

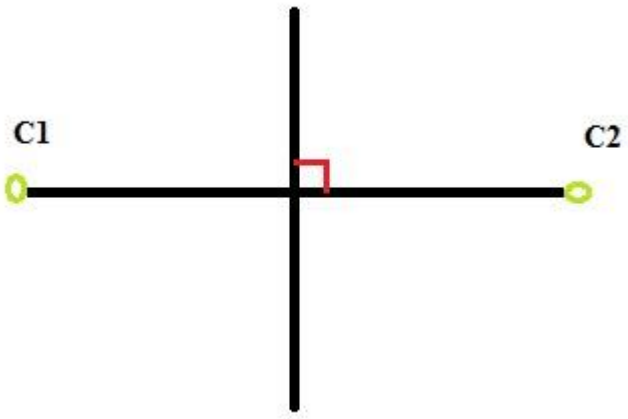
می توانیم بگوییم x_i عضو A_j اگر و فقط اگر داشته باشیم :

$$x_i \in A_j \Leftrightarrow j = argmin_l D(x_i, c_l)$$

یعنی x_i متعلق به خوشه A_j هست اگر که c_l نزدیک ترین مرکز خوشه به این داده باشد. این خوشه ها مراکزشان برای تصاحب این داده ها در رقابت هستند و به صورت Competitive عمل می کنند. پس می توانیم به این صورت بیان کنیم که :

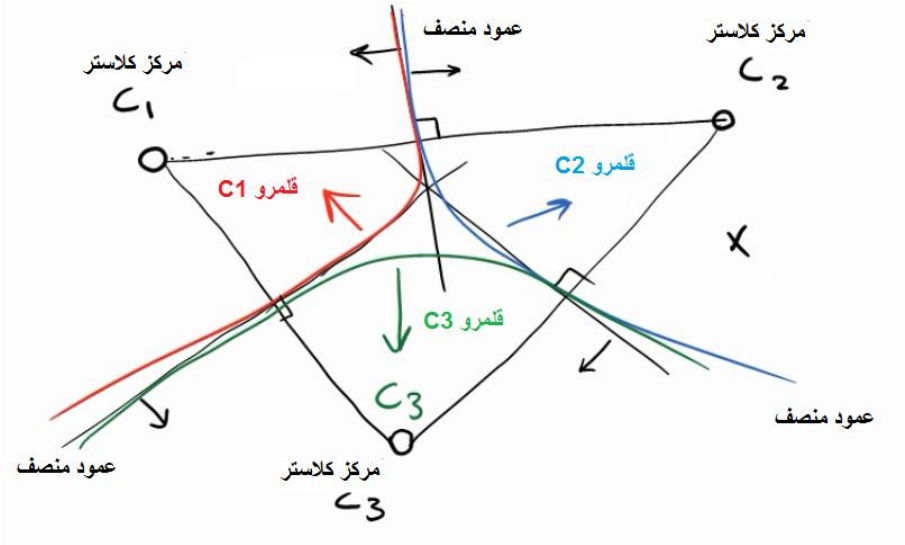
$$J(i) = argmin_l D(\mathrm{x_i}, c_l)$$

یعنی $J(i)$ شماره خوشه ی که مرکز آن خوشه نسبت به سایر مراکز خوشه نزدیک تر به x_i می باشد. به عنوان مثال وقتی که c_1, c_2 به این صورت باشند :



عمود منصف این خط بین دو خوشه را رسم می کنیم. عمود منصف مکان هندسی نقاطی است که از هر دو نقطه به یک فاصله هست و زمانی که عمود منصف را رسم می کنید نقاطی که سمت چپ آن هستند به c_1 نزدیک ترند و نقاطی که در سمت راست قرار دارند به c_2 نزدیک تر هستند. پس سمت چپ می شود قلمرو خوشه ۱ که مرکز آن c_1 هست و سمت راست می شود قلمرو خوشه ۲ که مرکز آن c_2 هست.

اگر فرض کنید که خوشه سومی نیز داشته باشیم شکل بالا به این صورت تغییر می کند:



آرایشی بین داده ها بهتر از بقیه هست که فاصله اعضای یک خوشه تا مرکز آن از همه کمتر باشد. به عبارت دیگر می دانیم که برای i خوشه $J(i)$ را در نظر بگیریم ویا در واقع x_i اختصاص پیدا می کند به $C_j(i)$ یا همان $A_j(i)$:

$$i \longrightarrow J(i)$$

$$x_i \dashrightarrow C_j(i) \ , \ A_j(i)$$

فاصله x_i و $C_j(i)$ می دانیم که به این اندازه است :

$$D(x_i, C_j(i))$$

درست است که این فاصله به نسبت دیگر مراکز خوشه ها کمترین هست ولی می خواهیم همین کمتر را نیز کمتر کنیم. پس کافی است برای همه i ها این را کمینه کنیم. این کار امکان پذیر نیست برای اینکه اگر داده ها زیاد باشد شما مجبورید همه داده ها را کمینه کنید. یک راه این است که همه داده ها را با هم جمع کنید و مجموع آن ها را کمینه کنید به این معنی که تابع هدفی را تعریف کنیم.

به ازای هر i ، فاصله x_i و مرکز خوشه مربوط به این ها را کمینه می کند:

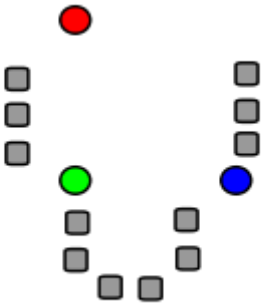
$$Cost\ Function : \frac{1}{N}\sum_{i} D(x_i, C_j(i))$$

بعضی ها علاقه دارند که میانگین بگیرند و بعضی افراد نیز علاقه دارند میانگین مربعات آن را محاسبه کنند. حالات مختلفی را برای آن می توان در نظر گرفت. شاید یک فرمول بندی دیگری که برای آن مرسوم هست و بخواهیم آن را در نظر بگیریم به این صورت است :

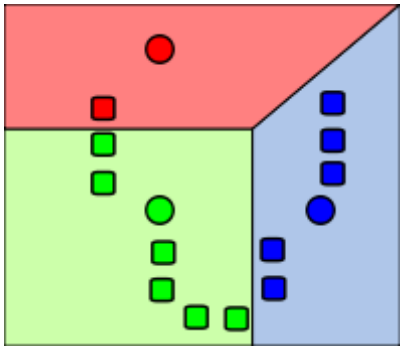
$$Cost\ Function : \frac{1}{N}\sum_{i}\sum_{x_i \in A_j} D(x_i, C_j)$$

می توان از نظر ریاضیات ثابت کرد که یک مرکز خوشه خوب مرکزی است که در مرکز جرم آن خوشه باشد یا میانگین اعضای آن خوشه باشد (میانگین وزن دار یا بدون وزن). لذا می شود فرض کرد که Cها همه می توانند میانگین خوشه هایشان باشند که نماینده خوبی برای اعضای خوشه باشند. پس این مسائله ی که اینجا مطرح شده است پیدا کردن k تا مرکز خوشه برای این m تا داده با این تعریفات به شرطی که تابع هدف مقدارش کمینه شود.

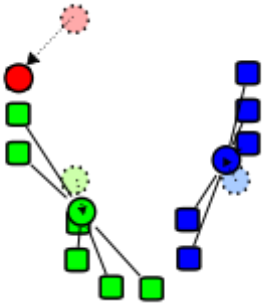
برای درک این موضوعات می بایست به یک حل یک مثال می پردازیم به صورتی که اگر فرض کنید این ها داده های ما هستند :



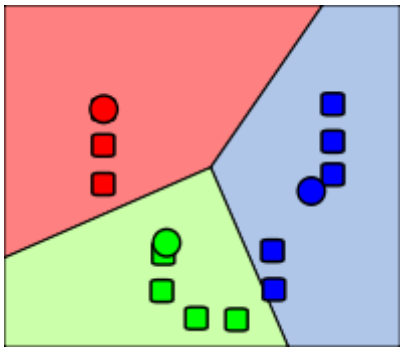
سه نقطه را به صورت تصادفی برای سه مرکز خوشه تعیین می کنیم. به این مرحله مقدار دهی اولیه گویند. بعد از این وارد یک حلقه می شویم تا جای که یک همگرایی حاصل شود. این سه مرکز خوشه سبز، قرمز و آبی با یکدیگر در رقابت هستند تا داده ها را تصاحب کنند.



همانطور که ملاحظه می فرمایید تنها یک داده در خوشه قرمز قرار گرفت. پس ابتدا چند مرکز تصادفی تعیین کردیم و ناحیه نفوظ آن ها را نیز تعیین کردیم و بعد دیدیم که داده ها جزو کدام خوشه هستند. یک فرضی داریم که می گوید که یک مرکز خوشه خوب باید نماینده خوبی برای اعضای آن خوشه باشد. از نظر آماری بهترین نماینده برای یک مجموعه اعداد میانگین یا مرکز جرمشان می باشد. در نتیجه از داده های که درون هر خوشه قرار دارند میانگین گیری می کنیم و مراکز را آپدیت می کنیم.



مراکز جدید که انتخاب شدند باز فاز تعیین قلمرو انجام می شود. در نهایت این روند باز تکرار شده و به این حالت می رسیم:



در نهایت می توان گفت این الگوریتم دارای دو فاز می باشد:

- فاز *Assignment* : تعلق دادن داده ها به خوشه ها
- فاز *Update* : مراکز خوشه ها آپدیت می شود به نحوی که دقیقا برابر میانگین همه اعضای آن خوشه باشد.

این دو فاز آن قدر تکرار می شوند که دیگر هیچ تغییری در خوشه ها نداشته باشیم.

توقف الگوریتم :

می توانیم بگوییم A ،B و C سه مرکز دسته هستند که هر کدام یک بردار می باشند :

$$[A\ ,B\ ,C] = \begin{matrix} A & B & C \end{matrix} \begin{bmatrix} x & x & x \\ y & y & y \end{bmatrix}$$

اگر این ماتریس دو در سه فعلی را منهای همین ماتریس در لحظه قبل کنیم و بررسی کنیم که اگر مقدار خطا از یک آستانه کمتر بود به این معناست که مرکز دسته ها *fix* شده اند و دیگر تغییر نمی کنند. اگر بخواهیم به صورت دقیق تر عمل کنیم می بایست A را از A و B را از B و C را از C به صورت نظیر به نظیر المنت هایشان را از هم کم کنیم برای اینکه ممکن است دو تا از خوشه ها خوب دسته بندی شده باشند و یکی از آن ها بد دسته بندی شده باشد.

$$\begin{cases} C_1 = (2\,,3) \\ C_2 = (3.5\,,3.75) \\ C_3 = (4\,,1) \end{cases} \qquad time: t-1$$

$$\begin{cases} C_1 = (2\,,3.000001) \\ C_2 = (3.5\,,3.78) \\ C_3 = (4.1\,,1.9) \end{cases} \qquad time: t$$

پس نظیر به نظیر المنت ها را از یک دیگر کم می کنیم و ماکزیمم تمام خطاها را می گیریم و اگر کم تر از اپسیلون بود آنگاه می گوییم که مرکز دسته ها ثابت می مانند.

$$\max(All\ 6\ Errors) \leq 0.00001$$

در این پروژه از مجموعه دادهای Movielensdataset (http://www.movielens.org) استفاده خواهد شد. این مجموعه داده شامل *1,00,000* رای می باشد که از ۱ تا ۵ توسط کاربران برای هر ایتم نسبت داده شده است. همچنین این مجموعه داده شامل ۹۴۳ کاربر بر روی ۱۶۸۲ فیلم می باشد که همه فیلم ها دارای یک سری ویژگی های اطلاعاتی تحت عنوان پروفایل می باشند که مشخص کننده ژانر آن فیلم هاست. این ژانر ها یا سبک فیلم ها می توانند کمدی، اکشن، انیمیشن و ... باشند که جمعا ۱۹ ژانر مختلف در نظر گرفته شده است. به این معنا که ما نیز در زمان خوشه بندی می بایست از این ۱۹ ژانر به عنوان ویژگی های هر یک از فیلم ها استفاده کنیم. پس ایتم ها یا همان فیلم هایمان به *k* خوشه تقسیم بندی شده و بر روی این *k* خوشه روش پالایش گروهی مبتنی بر ایتم ها یا همان *Collaborative Filtering* اعمال می کنیم.

برای خوشه بندی می بایست *k* تعداد از ایتم ها از مجموعه داده های ایتم به صورت تصادفی انتخاب شوند و از آن ها به عنوان *k* عدد مرکز خوشه استفاده کنیم. در مرحله بعد می بایست فاصله همه ایتم ها با تمامی مراکز خوشه ها را بدست آورد.

برای درک بهتر این روش می توان به این مثال توجه کرد. به عنوان مثال فرض کنید ۱۰ فیلم را خواهیم داشت و پروفایل هر فیلم شامل ۳ عدد ویژگی یا ۳ نوع ژانر مختلف می باشد که مشخص کننده سبک فیلم هست.

User	Comedy	Action	Musical
M1	0	1	0
M2	1	0	1
M3	1	1	0
M4	0	1	1
M5	0	0	1

M6	1	0	0
M7	0	0	1
M8	1	1	1
M9	1	1	0
M10	1	0	1

جدول ۱ : مثالی از مجموعه داده های فیلم

در این جا ما باید از معیار اندازه گیری فاصله منهتن (*Manhattan*) برای بدست آوردن فاصله بین مراکز خوشه ها و ایتم های دیگر استفاده کنیم.

$$d_1(p,q) = ||p-q||_1 = \sum_{i=1}^{n} |p_i - q_i|$$

پس اعمال خوشه بندی *K-Means* ، فیلم ها در این مثال به سه خوشه به صورت زیر تقسیم می شوند:

Cluster No.	Cluster Set
Cluster 1	{M1,M4}
Cluster 2	{M2,M5,M7,M10}
Cluster 3	{M3,M6,M8,M9}

جدول ۲ : خوشه بندی *K-Means*

حال می بایست بر روی هر کدام از خوشه ها، پالایش گروهی مبتنی بر ایتم که در فاز ۲ می باشد، اعمال گردد.

۲-۱-۲- خوشه بندی فازی C-Means :

در این روش داده ها با یک درصد تعلق به خوشه ها متعلق هستند و با یک نسبتی به همه خوشه ها تعلق دارند اما درجه تعلق برای هر خوشه متفاوت هست. همین نگرش فازی را به الگوریتم خوشه بندی k-means اضافه می کنیم و در نهایت به الگوریتم Fuzzy C-Means می رسیم. نکته ی که در مورد k-means وجود دارد این می باشد که انتخاب مراکز خوشه های اولیه خیلی در نتیجه تاثیر گذار هست با وجود اینکه الگوریتم ساده ای است هیچ تضمینی در پاسخ درست وجود ندارد. همین خاصیت در Fuzzy C-Means نیز وجود دارد ولی FCM به خاطر ماهیتی که دارد موفقعیت آمیز هست و خوب جواب می دهد. و کمتر در Local Optimum می افتد. به خاطر اینکه همیشه راهی برای فرار هست.

الگوریتم فازی C-Means به همان صورت K هایمان و مراکز دسته را خواهیم داشت و همچنین u_{ij} را خواهیم داشت که عددی بین ۰ و ۱ می باشد. u_{ij} را به عنوان درجه تعلق (Membership Function) داده x_i به خوشه j می شناسیم.

$$0 \leq u_{ij} \leq 1$$

با این اوصاف می توان به این صورت فرض کرد که u_{ij} عضو بازه ی ۰ و ۱ هست:

$$u_{ij} \in [0,1]$$

اما در K-Means، u_{ij} یا صفر هست یا یک هست:

$$u_{ij} \in \{0,1\}$$

اما در هر دو حالت این شرط را خواهیم داشت :

$$\sum_j u_{ij} = 1$$

یعنی درجه عضویت i در همه خوشه ها را اگر جمع ببندیم می شود ۱.

درجه عضویت u_{ij} به چه صورت تعیین می شود :

این مقدار درجه عضویت به صورت زیر تعیین می شود :

$$u_j(x) = \frac{1}{\sum_k \left[\frac{D(C_j,x)}{D(C_K,x)}\right]^{\frac{2}{m-1}}}$$

(نسبت فاصله مرکز j از x (مرکز مبنا) تقسیم بر فاصله مرکز k از x)

فرض کنید مرکز j صفر باشد که می شود صفر ضرب در یک عدد مثبت که می شود صفر و مجموع یک سری صفر ها می شود خود صفر و از طرفی صفر به توان یک عدد باز می شود صفر و یک تقسیم بر صفر نیز می شود بی نهایت. در حالت کلی به این صورت نیست ولی می دانیم که این هرچه قدر مقدارش کم تر باشد باعث می شود که مقدار این عبارت بیش تر شود.

M اگر برابر با یک باشد $\frac{2}{m-1}$ می شود بی نهایت: اگر آن عددی که به توان بی نهایت می رسانیم عدد بزرگ تر از یک باشد مخرج کلا بی نهایت می شود و اگر کم تر از یک باشد صفر می شود. در واقع این باعث می شود که اگر m به سمت بی نهایت میل کند این الگوریتم *FCM* نیز به سمت *K-Means* برود.

پس می توانیم بگوییم :

$$u_j(x) = \frac{u_j(x_i)}{\sum_k u_k(x_i)}$$

$$\sum_j u_{ij=1}$$

آپدیت مراکز :

می دانیم که u_{ij} یک عدد بین ۰ و ۱ هست و آن نیز $0 \leq u_{ij} \leq 1$ که درجه عضویت x_i در خوشه j می باشد. اگر $u_{ij} = 0$ باشد یعنی این خوشه j ربطی به x_i ندارد و در آن عضو نیست. در غیر این صورت اگر که $u_{ij} = 1$ باشد یعنی x_i به طور کامل عضو خوشه j هست. می توان گفت که u_{ij} درست هست که درجه عضویت x_i در خوشه j هست ولی می توان به عنوان سهم x_i در آپدیت مرکز خوشه j نیز در نظر گرفته شود.

فرض کنیم که u_{ij} وزن تاثیر هست پس میانگین وزن دار یا مرکز خوشه j را با این وزن ها حساب می کنیم:

$$C_j = \frac{\sum_i u_{ij} * x_i}{\sum_i u_{ij}}$$

در k-means نیز به همین صورت بود ولی در آن جا u_{ij} یا صفر یا یک بود.

مراحل الگوریتم :

۱) داده ها را آماده می کنیم.

۲) u_{ij} را به صورت تصادفی تولید می کنیم اما می توان با روش های بهتری مقدار دهی اولیه را انجام داد.

۳) مراکز دسته C_1 تا C_c را بدست می آوریم.

$$C_j = \frac{\sum_i {u_{ij}}^m * x_i}{\sum_i {u_{ij}}^m}$$

۴) u_{ij} را به روز رسانی می کنیم.

$$u_{ij} = \frac{1}{\sum_k \left[\frac{||x_i - C_j||}{||x_i - C_k||}\right]^{\frac{2}{m-1}}}$$

توقف الگوریتم :

با استفاده از تابع هدف J می توان مشخص کرد که چه زمانی این الگوریتم باید متوقف شود.

$$J = \sum_{i=1}^{N}\sum_{j=1}^{C} u_{ij}||x_i - C_j||^2$$

در هر بار اجرای برنامه مقدار *Cost Function* را حساب کرده و با مقدار مرحله قبل مقایسه می کنیم اگر اختلاف آن ها از یک مقدار اپسیلون کمتر بود الگوریتم را متوقف می کنیم.

حال که به توضیح کامل نحوه کار FCM پرداختیم می توانیم این روش خوشه بندی فازی را بر روی داده های جدول شماره ۱ اعمال کرد. نتایج این کار را در جدول شماره ۳ می توان مشاهده نمود.

User	Cluster 1	Cluster 2	Cluster 3
M1	0.81	0.03	0.16
M2	0.01	0.97	0.02
M3	0	0	1
M4	0.97	0.02	0.01
M5	0.08	0.91	0.02
M6	0.07	0.21	0.72
M7	0.08	0.91	0.2
M8	0.4	0.22	0.38
M9	0	0	1
M10	0.01	0.97	0.02

جدول 3 : خوشه بندی *Fuzzy C-Means*

جدول شماره ۳ نشان دهنده مقادیر عضویت هر ایتم به هر خوشه می باشد. جمع تمامی عضویت های هر ایتم می بایست برابر با یک باشد. فرض کنید ما برای اینکه مشخص کنیم که ایتم به کدام خوشه تعلق دارد یک مقدار آستانه در نظر می گیریم. به این صورت که اگر مقدار عضویت ها بزرگ تر مساوی آن آستانه باشد، پس آن ایتم متعلق به آن خوشه ها می باشد. با این رویکرد، خوشه ها ممکن است دارای ایتم های باشد که در دیگر خوشه ها نیز باشد. در این جا ما فرض می کنیم که مقدار آستانه 0.15 باشد. پس نتیجه اعمال این آستانه بر روی جدول ۳ در جدول شماره ۴ قرار دارد.

Cluster No.	Cluster Set
Cluster 1	{M1,M4,M8}
Cluster 2	{M2,M5,M6,M7,M8,M10}
Cluster 3	{M1,M3,M6,M7,M8,M9}

جدول ۴ : اعمال مقدار آستانه بر روی *FCM*

۲-۱-۳- کاهش مشکل Cold Start :

زمانی که یک ایتم یا کاربر جدید به سیستم اضافه می شود، این مشکل به وجود خواهد آمد که نمی توانیم از آن ها در پیشنهادات سیستم استفاده کنیم. به این دلیل کاربر جدید هیچ رای را به ایتم ها نداده است یا هنگامی که ایتم جدیدی اضافه شده است، هیچ رای را از طرف کاربران یا مشتریان دریافت نکرده است. این مشکل را با نام Cold Start می شناسیم که در ادامه نحوه مقابله با آن را بیان می کنیم. همانطور که ابتدای این مبحث اشاره داشتیم این مشکل در دو حالت مجزا ممکن است رخ دهد :

۱) کاربر جدید : زمانی که کاربر جدیدی به سیستم اضافه می شود ما شاهد مشکل Cold Start بودیم که برای حل آن می بایست کاربران جدید را جهت دریافت پیشنهادات ملزم کنیم تا یک آستانه مشخص به آیتم های که دوست دارند رای دهند.

۲) ایتم جدید : همچنین در ادامه نیز مشاهده کردیم که با اضافه کردن ایتم جدید به سیستم، این مشکل باز رخ می دهد که در جهت بر طرف ساختن آن می توان از این رویکرد تبعیت کرد – رای ایتم جدید M توسط کاربر U به صورت زیر تعیین می شود :

$$Rating\ of\ M(U) = \sum_{c=1}^{n} \overline{r_c(U)} * \mu_c(M)$$

به صورتی که $r_c(U)$ میانگین کاربر U در خوشه c می باشد. $\mu_c(M)$ مقدار عضویت ایتم جدید M در خوشه c می باشد.

۲–۲– فاز دوم – پالایش گروهی مبتنی بر ایتم :

خروجی فاز اول که خوشه بندی ایتم ها بود را به عنوان ورودی این فاز استفاده خواهیم کرد. و با استفاده از آن ماتریس رای های هر خوشه را بدست می آوریم. در نظر داشته باشد که ما در قسمت قبل ایتم ها را به k خوشه مختلف تقسیم کردیم. مقدار k که مشخص کننده تعداد خوشه ها می باشد را با تعداد ژانر های فیلم های درون مجموعه داده هایم قرار می دهیم. تعداد ژانر فیلم ها ۱۹ عدد می باشد پس ما نیز ایتم ها را ۱۹ خوشه تقسیم بندی می کنیم. خوشه های ایتم ها را که بدست آوردیم می بایست مشخص کنیم که در هر خوشه چه کاربرانی به چه ایتم های رای داده اند. در نتیجه بعد از عملیات خوشه بندی می بایست روش پالایش گروهی را بر روی هر خوشه به صورت جداگانه اعمال کنیم. به این صورت که شباهت ایتم های درون هر خوشه با یکدیگر را حساب می کنیم.

در این راه حل، ابتدا شباهت بین هر جفت ایتم i و j را به کمک فرمول ارتباطی Pearson بدست می آوریم. در نتیجه یک ماتریس n*n را بدست می آوریم که شامل شباهت بین ایتم ها می باشد. فرمول Pearson به صورت زیر می باشد:

$$S_{i,j} = \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_i)(r_{u,j} - \bar{r}_j)}{\sqrt{\sum_{u \in U}(r_{u,i} - \bar{r}_i)^2}\sqrt{\sum_{u \in U}(r_{u,j} - \bar{r}_j)^2}}$$

به صورتی که U مجموعه ی از تمامی کاربرانی می باشد که هم به ایتم i و هم به ایتم j رای داده اند. $r_{u,j}$ رای کاربر u به ایتم i می باشد و $\bar{r}_i$ نیز میانگین رای های ایتم i توسط تمامی کاربران رای داده به ایتم i می باشد.

حالا ما با این فرمول ماتریس شباهت تمامی ایتم ها با یک دیگر را نیز داریم که می توانیم با کمک این ماتریس k عدد از ایتم های که دارای بیش ترین شباهت می باشند را پیدا کنیم و از آن ها در پیش بینی رای کاربران به ایتم های که مشخص نیستند استفاده کنیم. پس رای ایتم i توسط کاربر فعال a با این فرمول به سادگی بدست می آید.

$$P_{a,i} = \frac{\sum_{j \in K}(r_{a,j} * S_{i,j})}{\sum_{j \in K}|S_{i,j}|}$$

به طوری که K مجموعه همسایه های k عدد ایتم که توسط کاربر فعال a رای داده شده اند و دارای بیش ترین شباهت به ایتم i می باشند.

پس تا اینجا ما k=19 خوشه از ایتم های درون مجموعه دادههایمان داشتیم که بر اساس این خوشه ها ما k ماتریس ایتم – کاربر را تشکیل دادیم که مشخص کننده ی این بود که در هر خوشه چه کاربرانی به چه ایتم

های رای داده اند. سپس بر روی این ماتریس ها به صورت جداگانه روش پالایش گروهی را پیاده سازی کردیم. در نهایت رای تمامی کاربران به تمامی ایتم ها را پیش بینی کردیم.

اما مساله دیگری که در خوشه بندی فازی با آن مواجه خواهیم شد این می باشد که این نوع خوشه بندی اجازه می دهد تا ایتم ها در بیش از یک خوشه با درجه عضویت های مختلف ظاهر شوند. برای حل این مشکل می بایست از این قانون پیروی کنیم که رای ایتم i میانگین رای های ایتم i در تمامی خوشه ها می باشد.

$$P_{a,i} = \frac{\sum_{c \in C} P_{a,i}^{c}}{N_c}$$

به صورتی که C مجموعه خوشه های است که کاربر a به ایتم i رای داده است و $P_{a,i}^{c}$ رای پیش بینی شده کاربر a به ایتم i در خوشه c می باشد. همچنین N_c تعداد خوشه های است که کاربر a به ایتم i رای داده است.

۳– نتایج پیاده سازی :

در این بخش از گزارش، به بررسی مجموعه داده های استفاده شده و مقایسه روش های قدیمی و حال پالایش گروهی یا همان Collaborative Filtering می پردازیم. در نهایت نتایج حاصل روش خود را نیز شرح می دهیم.

۳–۱– مجموعه داده ها :

در این پروژه ما از مجموعه داده های MovieLens برای سنجش کارایی الگوریتم پالایش گروهی خود استفاده می کنیم. این داده ها توسط گروه GroupLens در دانشگاه مینسوتا ایالات متحده جمع آوری شده است. در این مجموعه داده شامل ۴۵۰۰۰ کاربر که ۶۶۰۰ فیلم را رای داده اند. اما در این پروژه برای سنجش الگوریتم ما از 100,000 رای که به صورت تصادفی از مجموعه داده ها انتخاب شده اند استفاده کردیم. در نتیجه ما دارای ۱۰۰۰ کاربر بر روی ۱۶۸۰ فیلم هستیم که حداقل هر کاربر ۲۰ فیلم را رای داده است. رای های داده شده توسط کاربران در بازه ۱ تا ۵ می باشد. جهت دریافت مجموعه داده ها می توانید از این آدرس استفاده نمایید:

http://grouplens.org/datasets/movielens

اطلاعات بیش تر در مورد مجموعه داده را می توانید در فایل Data Set_README مورد جست جو قرار دهید. همچنین شایان ذکر است که اطلاعات پروفایل هر کاربر شامل سن، جنسیت، شغل و کد پستی می باشد. حال به بررسی فایل های که در پوشه مجموعه داده های خود داریم می پردازیم:

- u.data : این فایل شامل تمامی رای های ۹۴۳ نفر از کاربران به تمامی ۱۶۸۲ فیلم ها می باشد که جمعا شامل 100,000 رای است. هر کاربر به طور متوسط ۲۰ فیلم را رای داده اند. این نکته باید ذکر شود که کاربران و ایتم ها از ۱ شماره گذاری شده اند. داده های قرار گرفته در این فایل از این فرمت تبعیت می کنند :

user id | item id | rating | timestamp

timestamp زمانی که کاربر به سیستم پیوسته است را نشان می دهد. این پارامتر به فرمت unix می باشد.

- u.info : این فایل شامل اطلاعات عمومی مجموعه داده ها می باشد، تعداد کاربران، تعداد ایتم ها، تعداد رای های موجود در مجموعه داده ها است.
- u.item : این فایل شامل اطلاعات پروفایل همه ایتم ها یا همان فیلم هاست که از فرمت زیر تبعیت می کند:

movie id | movie title | release date | video release date |IMDb URL | unknown | Action | Adventure | Animation |Children's | Comedy | Crime | Documentary | Drama | Fantasy |Film-Noir | Horror | Musical | Mystery | Romance | Sci-Fi |Thriller | War | Western |The last 19 fields are the genres

در ۱۹ فیلد آخر ژانر فیلم را مشخص می کنیم که شامل مقادیر دودویی ۰ و ۱ می باشند. ۰ بیانگر این است که فیلم مورد نظر در آن ژانر قرار ندارد و ۱ به طور برعکس بیانگر این است که فیلم مورد نظر در آن ژانر قرار دارد. شماره ایتم ها همان شماره ایتم های که در فایل u.data در کنار رای ها داشتیم.

- u.genre : لیست تمامی ژانر های فیلم ها در این فایل قرار دارد.
- u.user : این فایل شامل اطلاعات پروفایلی کاربران می باشد که به این صورت فرمت بندی شده اند :

user id | age | gender | occupation | zip code

شماره کاربران همان شماره های است که در فایل u.data در کنار رای ها داشتیم.

- u.occupation: این فایل شامل تمامی شغل های کاربران می باشد.
- فایل u1.base و u1.test تا u5.base و u5.test شامل داده های از مجموعه داده های رای ها می باشد که به صورت 5-Fold به ۵ قسمت از داده های آموزش و تست تقسیم بندی شده اند.

۳-۲- معیار های سنجش :

در جهت سنجش کارایی الگوریتم اعمال شده بر روی مجموعه داده ها ما از چند معیار سنجش همچون Mean Absolute Error (MAE)، Root Mean Square Error (RMSE) استفاده خواهیم. اگر n تعداد رای های واقعی در مجموعه ایتم ها می باشد، MAE به عنوان تفاوت قدر مطلق میانگین بین دو ایتم می باشد. فرض کنید که p1,p2,p3,...,pn به عنوان رای های پیش بینی شده توسط الگوریتم باشد و q1,q2,q3,....,qn به عنوان

رای های واقعی که کاربران داده اند و متناظر با رای های پیش بینی شده اند باشد، پس MAE به صورت زیر می باشد:

$$MAE = \frac{\sum_{i=1}^{n} |p_i - q_i|}{n}$$

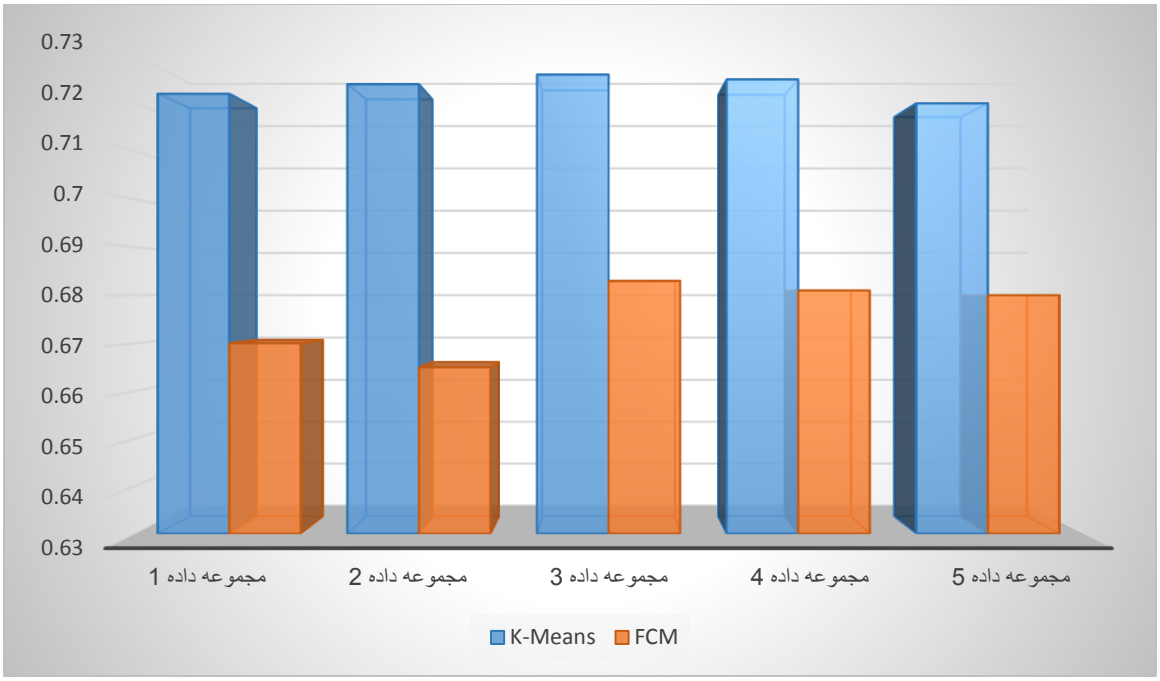
هرچه MAE کم تر باشد بیان گر دقت بیش تر پیش بینی انجام شده توسط الگوریتم می باشد.

Algorithm	MAE	RMSE
Linear FCM based CF	0.7160	
Our First Approach (K-Means)	0.7286	1.0101
Our Second Approach (FCM)	0.6806	0.9459

جدول ۵ : نتایج حاصل از الگوریتم های مختلف روش پالایش گروهی بر روی مجموعه داده ها

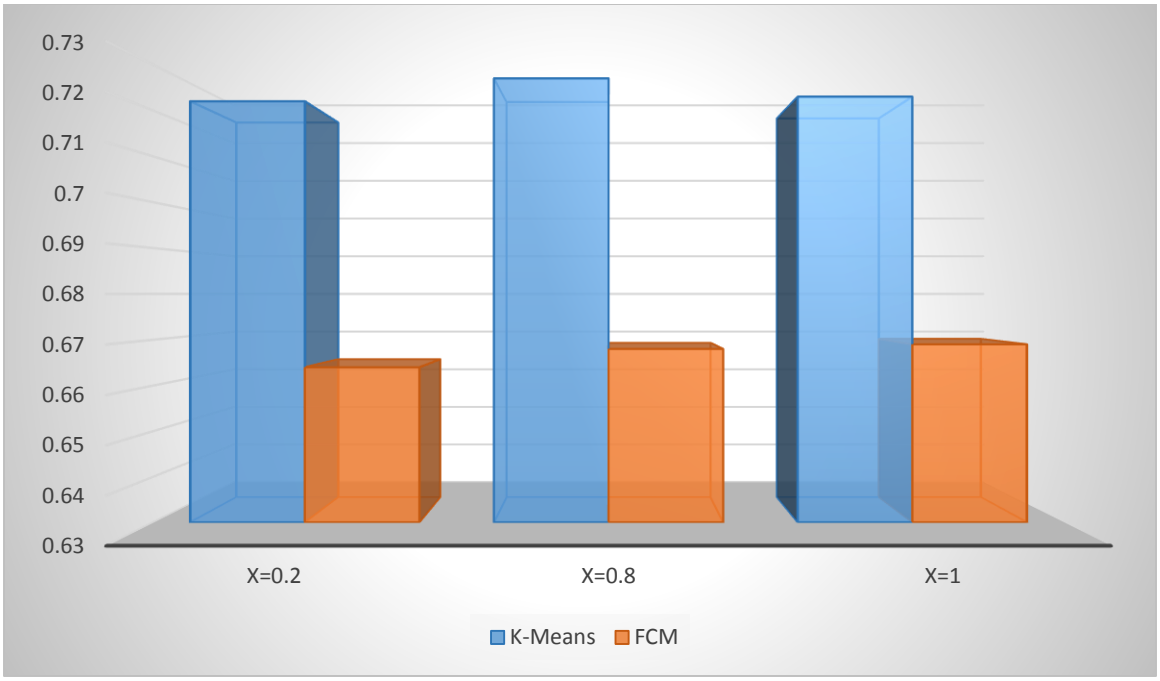
در جدول شماره ۵، نتایج حاصل از هر دو روش پالایش گروهی که در الگوریتم خود بیان کردیم نشان داده می شود. همچنین در کنار آن ها نتایج روش پیشین که در این حوزه بر روی مجموعه داده ها مورد استفاده قرار گرفته بود. همان طور که مشاهده می فرمایید نتایج حاصل از روش پالایش گروهی با خوشه بندی فازی از همه بهتر می باشد.

مجموعه داده ها را که از وب سایت مورد نظر دانلود می فرمایید به صورتی در نظر گرفته شده اند که به جای اینکه ما از رویکرد 5-Fold استفاده کنیم، خود جمع آوری کننده گان مجموعه داده ها ۵ مجموعه داده مجزا در اختیار ما قرار داده اند تا هر کدام را به صورت مجزا مورد بررسی قرار دهیم. اما می توان به جای این کار، فایل اصلی را نیز فراخوانی کرد و 5-Fold را بر روی آن مجموعه داده پیاده کرد.



نمودار ۱ : نتایج حاصل از *MAE* هر دو روش بر روی مجموعه داده ها

در نمودار شماره ۱ مشاهده می کنید که ما هر دو الگوریتم را بر روی تمامی مجموعه داده ها اعمال کردیم و *MAE* هر کدام را نیز به خوبی می توان مورد تحلیل قرار داد همان طور که قبلا نیز ذکر شد *FCM* دارای نتایج نسبتا بهتری در مقایسه با روش های دیگر دارد. دلیل برتری *FCM* بر *K-Means* در این پروژه به این دلیل است که خوشه بندی *K-Means* به صورت سخت گیرانه که هر ایتم تنها باید متعلق به یک خوشه باشد، عملیات خوشه بندی را انجام می دهد. اما این ایده در زندگی روزمره ما زیاد مشاهده نمی شود و این قضیه بیش تر ملموس می باشد که یک ایتم ممکن است به چند خوشه تعلق داشته باشد. به همین دلیل است که در روش دوم یا همان خوشه بندی فازی ما به نتیجه بهتری نسبت به روش *K-Means* رسیدیم.



نمودار 2 : نتایج حاصل از MAE هر دو روش بر روی مجموعه داده ها با چگالی متفاوت

همان طور که مشاهده می فرمایید در نمودار شماره ۲، نتایج هر دو روش خوشه بندی K-Means و خوشه بندی FCM بر روی مجموعه داده ها با چگالی متفاوت می توان مورد بررسی قرار داد. این سه مجموعه داده بر اساس چگالی رای های که دارند با یک دیگر تفاوت دارند به این صورت که X=0.2 به این معناست که در مجموعه داده 20,000 رای وجود دارد و همچنین X=0.5 به این معناست که 50,000 رای وجود دارد. در این آزمایش نیز نشان داده می شود که FCM بهتر از K-Means عمل می کند.

۴– منابع:

1) Verma, S. K., Mittal, N., & Agarwal, B. (2013). Hybrid recommender system based on fuzzy clustering and collaborative filtering. In *2013 4th International Conference on Computer and Communication Technology (ICCCT)*.

2) Duda, R. O., Hart, P. E., & Stork, D. G. (2012). *Pattern classification*. John Wiley & Sons