گام های تئوری و چالش ها:

T1 : موضوع پروژه ما ایجاد یک سیستم پیشنهاد دهنده کتاب براساس دیتاست goodbooks10-k می باشد. علت انتخاب این موضوع به سبب جالب و ملموس بودن آن از نظر کاربرد و همچنین پیدا کردن دیتاست مناسب برای آن بوده است. انتظار نهایی ما از انجام این پروژه اولا این بوده که حداکثر استفاده را از مطالبی که در این درس آموخته ایم در انجام پروژه به کار بگیریم و نهایتا یک سیستم کارا تحویل دهیم که بتواند عملکرد خود را به درستی انجام دهد.

T2 : دیتاست انتخاب شده دیتاست goodbooks_10k می باشد (سال ایجاد دیتاست : ۲۰۱۷ و منبع اصلی دیتاست سایت goodreads می باشد : https://www.goodreads.com) این دیتاست حاوی ۵ مورد فایل csv است و ما از ۳ مورد آن استفاده کرده ایم به علاوه ۲ فایل csv حاوی ژانر و اطلاعات نویسندگان را از دیتاست بزرگتر و کامل تر goodreads گرفتیم. ۳ مورد فایل csv انتخاب شده از goodbooks-10k شامل این موارد میباشد :

۱- فایل books.csv حاوی اطلاعات مربوط به کتاب های موجود در دیتاست. ستون های این فایل در قسمت زیر و بخش data understanding پروژه مفصلا شرح داده شده است :

00 ◆book_id : Book identifier in books dataset

01 ◆goodreads_book_id : Book identifier in original goodreads dataset

02 ◆best_book_id : Identifier which generally points to the most popular editions of a given book

03 ◆work_id : Refers to the book in the abstract sense

04 ◆books_count : The number of editions for a given work

05 ◆isbn : International Standard Book Number (ISBNs were 10 digits in length up to the end of December 2006)

06 ◆isbn13 : International Standard Book Number (since 1 January 2007 ISBNs now always consist of 13 digits)

07 ◆authors : Author or authors of the book

08 ◆original_publication_year : The year which book has been published

09 ◆original_title : The name under which the book was published

10 ◆title : Original_title in short format

11 ◆language_code : The language of book

12 ⬍average_rating : The average rating of the book received in total

13 ⬍ratings_count : Total number of ratings the book received

14 ⬍work_ratings_count : Total number of ratings the book received regarding all kinds of format or edition of each book

15 ⬍work_text_reviews_count : Total number of written text reviews the book received regarding all kinds of format or edition of each book

16 ⬍ratings_1 : Total number of rate (1) the book received

17 ⬍ratings_2 : Total number of rate (2) the book received

18 ⬍ratings_3 : Total number of rate (3) the book received

19 ⬍ratings_4 : Total number of rate (4) the book received

20 ⬍ratings_5 : Total number of rate (5) the book received

21 ⬍image_url : Url of the original image of book

22 ⬍small_image_url : Url of the short image of book

۲- فایل to_read حاوی اطلاعات کاربران و کتاب هایی که علامت زده اند تا در آینده آن ها را بخوانند اما هنوز این کتاب ها توسط آن ها خوانده نشده است :

00 ⬍user_id : The identifier of book that user wants to read it

01 ⬍book_id : The identifier of user who wants to read the book

۳- فایل ratings حاوی اطلاعات کاربران و کتاب هایی که آن ها را خوانده اند و به آن ها امتیاز داده اند :

00 ⬍book_id : The identifier of book that has received rate

01 ⬍user_id : The identifier of user who has given rate

02 ⬍rating : The rate that the user has given to the book.

دو فایل دیگری که استفاده نشده است حاوی book-tags و tags می باشد که بیشتر نظرات پراکنده کابران را شامل می شود و طبق بررسی های صورت گرفته چندان قابل مدل سازی و استفاده نبود.

لینک مربوط به دانلود فایل های دیتاست اصلی goodbook-10k استفاده شده :

https://github.com/zygmuntz/goodbooks-10k

فایل های goodreads_book_authors و goodreads_book_genres_initial نیز از لینک زیر برداشته و به دیتاست اصلی اضافه شده اند.

https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/books

چند مورد کارهای مرتبط بر روی این دیتاست موارد زیر می باشد :

https://github.com/ebehlmann/book-recommender/blob/master/book_recommendation_engine.ipynb

https://www.kaggle.com/renehlavova/recommender-system-for-books

T3 : فاز های CRISP DM شامل موارد زیر است :

Business/research understanding phase : بررسی پیرامون سیستم های پیشنهاد دهنده، نحوه کار آن ها و نیز بررسی کارهای مشابه بر روی دیتاست مورد نظر.

Data understanding phase: بررسی تک تک ستون های دیتاست و فهمیدن اینکه هر فیلد دقیقا چه چیزی را نشان می دهد. بدست آوردن شاخص های آماری داده ها مانند مینیمم، ماکسیمم، میانگین، میانه و چارک ها و ... ، بررسی رنج داده های هر ستون.

Data preparation phase : فاز آماده سازی دیتا شامل شناسایی و حذف یا جایگذاری داده های پرت در دیتاست های to_read و ratings و books، جایگذاری مقادیر null درون دیتاست books. (دیتاست های ratings و to_read فاقد داده پرت یا missing value بودند)، بررسی متغیر های categorical و dummy کردن آن ها درون دیتاست books. (متغیر هایی مانند زانر، زبان و نویسنده)، خوشه بندی نویسنده ها بر اساس اطلاعاتی که از آن ها داریم و انتخاب تعداد خوشه مناسب و ارزیابی خوشه بندی براساس شاخص هایی اینرسی و سیلوئت، کشف متغیر های مرتبط و دارای همبستگی درون دیتاست کتاب و بررسی ارتباط میان تک تک متغیر ها با متغیر هدف یعنی rating، بررسی توزیع متغیر های درون دیتاست و نرمالایز کردن آن ها در صورت ضرورت، یکسان سازی رنج متغیر های دیتاست، کاهش ابعاد با استفاده از pca برای حذف متغیر های مرتبط و بررسی تعداد کامپوننت انتخابی برای خروجی pca و میزان پوشش متغیر هدف توسط هر یک از کامپوننت ها، جوین زدن برخی از دیتاست ها مانند books و ratings برای ایجاد ورودی مدل.

Modeling phase : ابتدا توزیع متغیر خروجی درون داده ها بررسی شده و سپس به سبب نامتعادل بودن این توزیع عمل balancing صورت گرفته است سپس داده ها به دو دسته تست و یادگیری تقسیم شده و مجددا توزیع متغیر خروجی درون هر یک از داده های تست و یادگیری نیز ارزیابی شده است. در این قسمت از دو task مدل سازی استفاده کرده ایم اولی کاوش قواعد برای یافتن کتاب هایی که عموما باهم خوانده شده اند و نیز نویسندگانی که کتاب های آن ها معمولا باهم خوانده شده است و سپس classification و پیش بینی rate ای که هر کاربر به هر کتاب می دهد. برای classification از مدل های شبکه عصبی کتابخانه sklearn و keras استفاده کرده ایم. هردو دقت یکسانی را نتیجه داده اند اما کتابخانه keras بسیار سریعتر است لذا از آن برای مدل سازی نهایی استفاده کرده ایم. در کاوش قواعد نیز از کاوش قواعد ساده استفاده کرده ایم و دیتافریم های one_hot را براساس اینکه هر کاربر چه کتاب هایی را مطالعه کرده و چه کتاب هایی را مطالعه نکرده است و نیز براساس اینکه از چه نویسندگانی کتاب خوانده و از چه نویسندگانی کتاب نخوانده است اماده کرده و در مورد اول با minsup = 0.1 و در مورد دوم با minsup = 0.25 آیتم ست های متناوب را بدست آورده و قواعد مربوطه را استخراج کرده ایم.

Evaluation phase : در این فاز دقت مدل ها سنجیده شده است با توجه به اینکه با استفاده از هیچ مدلی نتوانستیم به طور خالص به دقت بیش از ۳۰ درصد دست یابیم از تکنیک هایی برای بهبود دقت مدل استفاده کرده ایم. در نتایج مشاهده کردیم که چندین مورد از احتمالات امتیازات در خیلی از موارد نزدیک به هم هستند و در مدل صرفا ماکسیمم انها انتخاب می شود و به عنوان کلاس نهایی گزارش می شود. لذا تصمیم گرفتیم در انتخاب امتیاز نهایی خودمان راهی پیش گیریم که نتیجه قابل قبول تر باشد . روش ما به این صورت است که میانگین ۵ احتمال امتیاز را محاسبه میکنیم . سپس آن احتمالاتی که از این میانگین بیشتر هستند را در نظر گرفته و بین اندیس آنها میانه میگیریم . مثلا اگر سه عدد ۱ و۲و ۳ انتخاب شدند میانه ۲ می شود و امتیاز نهایی را ۲ درنظر میگیریم . اینکار باعث می شود که به صورت تک بعدی به احتمالات نگاه نشود و یک بررسی بین تمام انها انجام شده باشد تا بتوان نتیجه بهتری گرفت.

Deployment phase : با استفاده از مدل های نهایی یک تابع recommender ایجاد کرده ایم که ابتدا لیستی از کتاب های خوانده شده توسط هر کاربر را دریافت کرده و سپس با کاوش قواعد کتاب های مرتبط و نیز کتاب های نویسندگان مرتبط را نیز استخراج کرده و سپس امتیازی که احتمال می رود کاربر مربوطه بر هر یک از کتاب های پیشنهادی بدهد را پیش بینی کرده و ۱۰ مورد از بهترین نتایج پیش بینی شده را به عنوان خروجی صادر می نماییم.

توضیحات مفصل تری نیز درون فایل های ipython برای تک تک بخش ها داده شده است. پاور پوینت ضمیمه شده نیز حاوی گام هاب برداشته شده می باشد.

چالش ها :

۱- بزرگترین چالش ما در انجام این پروژه داده های با حجم زیاد بود که در برخی موارد باعث سنگین و زمانبر شدن عملیات ها می گشت و بعضا ناچار شدیم به دنبال راه هایی باشیم که از نظر عملیاتی سبک هستند.

۲- برای پر کردن داده های null و missing value ها از کتابخانه pyisbn استفاده کردیم اما محدودیت این کتابخانه در تعداد پاسخ هایی که در یک روز به ما میداد اندکی چالش برانگیز بود و هر بار پس از حدود ۱۵۰۰ درخواست ip مربوطه را ban میکرد. لذا عملیات resovle برای ۱۰ هزار کتاب اندکی زمانبر شد.

۳- چالش دیگر عدم وجود ژانر کتاب درون دیتاست goodbook-10k بود که برای حل این چالش به دنبال دیتاست دیگری گشته و ژانر ها را بدست آورده و به دیتاست اصلی اضافه کردیم. همین راه حل را برای بدست آوردن اطلاعاتی درباره نویسندگان نیز به کار گرفتیم.

۴- چالش بزرگ دیگری که با ان مواجه شدیم دقت مدل ها بود. از مدل های درخت تصمیم، random forest، mpl و شبکه عصبی کتابخانه keras استفاده کردیم اما دقت بیش از ۳۰ الی ۳۵ درصد حاصل نشد برای حل این مشکل یک ترفند به کار گرفتیم که با استفاده از عملیات هایی بر روی خروجی مدل شبکه عصبی بتوانیم به دقت مطلوب دست یابیم.

۵- در فاز EDA متوجه شدیم متغیر user_id یعنی کاربر و سلیقه وی بیشترین تاثیر را در متغیر هدف classification task دارد و به دنبال روش هایی گشتیم که بتوانیم تاثیر این متغیر را در مدل افزایش دهیم.

۶- در ابتدای کار تصور کردیم برای کاهش ابعاد تمام متغیر ها میتوانیم از pca استفاده کنیم اما در ادامه متوجه شدیم این روش کاهش ابعاد برای ستون های flag مانند کابرد ندارد و لذا به دنبال راه هایی برای اینکار بودیم از یک مورد مشابه به نام mca نیز بهره بردیم اما باعث کاهش ابعاد چندانی نشد و میزان پوشش هر کامپوننت خروجی آن بسیار کم بود. البته این مورد مربوط به زمانی است که ژانر را در اختیار نداشتیم و سعی داشتیم از title کتاب ها آن ها را دسته بندی کنیم.

۷- حتی با وجود خوشه بندی نویسندگان و ژانر ها دقت خالص شبکه عصبی افزایش چندانی نداشت و صرفا اندکی بهبود یافت که برای حل مشکل دقت از تکنیک های دیگری استفاده کردیم.