# گزارش پروژه CA0

نام و نام خانوادگی: پریسا یحیی‌پور فتیده شماره دانشجویی: ۸۱۰۱۰۱۵۵۱

**مرحله پیش‌پردازش:**

ابتدا هر دو فایل csv توسط کتابخانه pandas باز می‌شوند. سپس ابتدا جداسازی اطلاعات فایل books_train انجام می‌شود.

کلمات و علائم توضیح کتاب و عنوان کتاب توسط توابع کتابخانه هضم جداسازی می‌شوند. در زمان تشکیل BOW علائم نگارشی و اعداد از لیست حذف می‌شوند چون اطلاعات خاصی برای تعیین دسته بندی کتاب در اختیار ما قرار نمی‌دهند. در همینجا با توجه به ژانر کتاب مورد بررسی کلمات در یک dictionary قرار می‌گیرند به طوری که کلمه را به تعداد دفعات تکرار شده متصل می‌کند.

پس از تشکیل BOW مرحله جداسازی کلمات فایل books_test انجام می‌شود. به گونه‌ای که کلمه های ستون توضیح و عنوان کتاب در یک لیست ریخته می‌شوند و سپس علائم نگارشی و اعداد از آن لیست حذف می‌شوند. در نهایت لیست کلماتی که برای محاسبه احتمال لازم داریم در قالب یک لیست به ستون دوم همان dataframe منتقل می‌شوند. چرا که پس از محاسبه احتمال صحت بررسی انجام شده را با توجه به ژانر بسنجیم.

---

**پیاده سازی ساده:**

فایل مربوطه: without_additive.py

در این فایل احتمالی برای کلمه هایی که در BOW نیستند در نظر گرفته نشده و فقط احتمال کلمه هایی که در فایل train وجود داشت محاسبه شده اند.

به دلیل اینکه تعداد کلمات ناموجود در BOW در داده های تست زیاد است این روش با دقت کمی خروجی می‌دهد.

دقت در این روش ۴۲.۲۲ درصد شده است.

لازم به ذکر است برای پیدا کردن ژانر، کافی است مقدار $P(X|c)$ را محاسبه و در نهایت مقایسه کنیم. چرا که ترم $P(X)$ در مخرج برای همه وجود دارد. همچنین $P(c)$ برای همه ژانر ها برابر یک ششم خواهد بود.

---

**پرسش اول:**

۱- صفر در نظر گرفتن: در صورتی که به کمک فرمول ضرب (لگاریتم نگیریم) احتمال هر ژانر را محاسبه کنیم اگر حتی یکی از کلمه های داده تست در BOW وجود نداشته باشد، احتمال آن ژانر صفر گزارش می‌شود چراکه احتمال در صفر ضرب شده بنابراین دقت به شدت کاهش می‌یابد.

۲- در نظر نگرفتن: همان طور که در بالا اشاره شد درصورتی که هیچ احتمالی برای کلماتی که در BOW نیستند در نظر نگیریم، می‌توانیم هم از فرمول جمع (لگاریتم) و هم ضرب استفاده کنیم. دقت در این روش ۴۲.۲۲ درصد خواهد بود.

---

**پیاده سازی additive smoothing:**

فایل مربوطه: without_stopword.py

در این روش فرمول بندی محاسبه احتمال تغییر می‌کند به گونه ای که به کلمه هایی که در BOW نیستند احتمال کوچکی اختصاص می دهیم. در این کد به جای آلفا (طبق فرمولی که در گروه درسی برای این روش بود) عدد ۱۰۰ گذاشته شده. این عدد باید با توجه به میزان اهمیتی که به داده هایی که در BOW نیستند می دهیم، تعیین شود. با توجه به اینکه در اکثر ژانر ها کل کلمات از ۱۰ هزار بیشتر بود ۱۰۰ جوابگو است و بالاترین دقت را می دهد. درصورتی که اعداد کوچکتری به جای الفا قرار دهیم دقت کاهش می یابد. همچنین برای الفا های بزرگتر از ۱۰۰ دقت تغییری نمی کند.

دقت محاسبه شده در این روش ۶۳.۱۱ درصد است.

---

**پرسش دوم:**

با توجه به اینکه تعداد کلمات بسیار بالاست و در عین حال احتمال رخ دادن اکثر کلمه ها در یک ژانر بسیار کوچک است، همچنین احتمال اختصاص داده شده به کلماتی که در BOW نیستند هم کم است، در صورتی که از ضرب استفاده کنیم حاصل در برخی جاها به صفر میل خواهد کرد. همچنین اگر روش additive smoothing اعمال نشود با ضرب شدن صفر به جای آن، کل احتمال صفر خواهد شد.

بنابراین به جای این کار از همه احتمال ها لگاریتم می گیریم و جمع می کنیم همچنین نیازی به محاسبه P(Y) نمی باشد چون احتمال رخ داد همه ژانر ها باهم برابر است. بنابراین فقط حاصل سیگما را حساب و مقایسه می کنیم.

---

**بخش امتیازی: قسمت دوم**

**حذف کردن stop words:**

فایل های مربوطه: final.py & final_no_additive.py

در این کد ها فایل sw.csv توسط کتابخانه pandas خوانده شده و داده های آن در یک لیستی از رشته های حرفی ذخیره می‌شوند. در هنگام پردازش داده های فایل های books_train & books_test علاوه بر اعداد و علائم نگارشی، این کلمات که پر تکرار هستند و اطلاعات خاصی درباره ژانر نمی دهند را حذف می‌کنیم.

در صورتی که به کلماتی که در BOW نیستند هیچ احتمالی اختصاص ندهیم، دقت برنامه ۴۴ درصد خواهد بود که یک درصد از حالتی که stop word را حذف نکرده ایم بیشتر است. چون ممکن است در ژانری کلمه های معمولی که ربطی به ژانر ندارد زیاد تکرار شده باشد در نتیجه احتمال بزرگی را به خود اختصاص خواهند داد. از طرفی این کلمات ممکن است در هر متنی باشند. بنابراین با حذف آنها دقت را افزایش می دهیم.

در صورتی که additive smoothing پس از حذف stop word اعمال شود، دقت برنامه به ۶۴.۲۲ خواهد رسید که باز هم یک درصد از حالتی که حدف نکرده بودیم بیشتر است(به دلیل قبلی).

---

**نکات کلی:**

همان طور که اشاره شد برای پردازش متن نیازی به محاسبه تمام پارامتر های فرمول بیز نیست، چراکه برای همه حالات این احتمالات برابر هستند و صرفا با محاسبه پارامتر متفاوت و مقایسه آن به نتیجه خواهیم رسید.

برای جلوگیری از رخ دادن overflow در محاسبه احتمالات، در تمامی کد ها از لگاریتم استفاده شده است. به دلیل اینکه به اعداد خیلی کوچک و بعضا منفی نرسیم، مبنای لگاریتم ۰.۵ در نظر گرفته شده و در فرایند ماکزیمم گیری هم مشکلی ایجاد نخواهد شد.

در روش additive smoothing برای جلوگیری از تکرار کد، تابع مربوط به محاسبه احتمال هر کلمه در هر ژانر تغییر کرده تا میزان احتمالی که باید برای کلماتی که در BOW نیستند را ریترن کند.

داده های احتمالات برای سرعت بخشیدن به برنامه در آرایه های numpy نگه داری شده اند، چراکه که طبق مطالعه انجام شده، سرعت انجام محاسبات در داده های کتابخانه numpy بیشتر است.

علاوه بر درصد دقت برنامه، تعداد کتاب های درست تشخیص داده شده از ۴۵۰ تست و زمان اجرای برنامه نیز خروجی داده می شود.