

مقدمه

در این پروژه قصد داریم با استفاده از Naive Bayes Classifier به تجزیه و تحلیل مقالات سایت دیجی کالا مگ^۱ و دسته‌بندی آن‌ها بپردازیم و سعی کنیم با استفاده از داده‌هایی که در مورد توضیحات هر مقاله داریم، دسته‌بندی آن را پیش‌بینی کنیم.

معرفی مجموعه داده

مجموعه داده تعدادی مقاله در فرمت CSV در اختیار شما قرار گرفته است. در هر داده، توضیحات مقاله و همینطور دسته‌بندی آن مشخص شده است. در این مجموعه داده ۴ دسته وجود دارند که به صورت زیر می‌باشند:

علم و تکنولوژی، هنر و سینما، بازی ویدئویی و سلامت و زیبایی

label_id	label	content
3	علم و تکنولوژی	... نمایش تبلیغ در لاکاسکرین تعدادی از گوشی‌های ه
5	هنر و سینما	... در باکس آفیس پس از بازخورد Justice League شکست
5	هنر و سینما	... کلاسیک بینی؛ همه چیز در یک شب اتفاق افتاد فیلم
3	علم و تکنولوژی	... ایل دوباره سراغ رنده رفته چراکه آنگرید کردن سط
0	بازی ویدئویی	... Ori and the Blind بررسی جزء به جزء بهترین بخش

دو فایل در اختیار شما قرار گرفته است که یکی برای آموزش و دیگری برای ارزیابی مدل شما است. فایل مربوط به آموزش مدل به عنوان train.csv و همینطور فایلی که مربوط به ارزیابی مدل شما است با نام test.csv در اختیار شما قرار گرفته است. دقت داشته باشید که تعداد سطرها به ازای هر موضوع دسته‌بندی در هر فایل به صورت متوازن قرار داده شده است و نیازی به یکسان کردن تعداد آگهی‌ها از دسته‌بندی‌های متفاوت که به نام resampling شناخته می‌شود نیست. البته

¹ <https://www.digikala.com/mag/>

برای مطالعه بیشتر می‌توانید این موضوع را نیز در نظر بگیرید. این کار برای از بین بردن bias موجود در داده‌هایی که تعداد کلاس‌های خروجی آن‌ها با هم برابر نیست استفاده می‌شود.

فاز اول: پیش‌پردازش داده

در فاز اول باید اطلاعات متنی داخل مجموعه داده را برای تحلیل‌های بعدی پیش‌پردازش کنیم. برای این کار می‌توانید از کتابخانه‌ی [Parsivar](https://github.com/ICTRC/Parsivar)² یا [هضم](https://github.com/sobhe/hazm)³ استفاده کنید یا خودتان موارد مورد نیازتان را پیاده‌سازی کنید. شما باید عنوان و توضیحات‌هایی که موجود است را تا حد ممکن Normalize کنید. (روش‌های ممکن، شامل حذف کلمات پرتکرار یا همان stop words، تبدیل کلمات به ریشه آنها و ... است.)

دقت کنید که این کار هم روی داده‌های train و هم روی داده‌های test باید انجام شود و لزوماً اجرای هر نوع پیش‌پردازشی باعث بالا رفتن دقت مدل شما نخواهد شد. روش‌های متفاوت را با استفاده از کتابخانه یا بدون آن امتحان کنید و ترکیب هر کدام از آنها که به مدل شما بیشتر کمک می‌کند را اجرا کنید.

البته به جز موارد توضیح داده شده می‌توانید تنها به حذف ایست واژه‌ها و کاراکترهای بی‌اهمیت مانند \n و \r بسنده کنید. اما لازم است تا تاثیر انواع دیگر پیش‌پردازش‌ها را نیز مشاهده کنید و در گزارش خود توضیحی در مورد آن‌ها ارائه دهید. ۱. در گزارش کار خود، جایگزین کردن کلمات با روش stemming یا lemmatization را توضیح دهید.

فاز دوم: فرآیند مسئله

در این مسئله می‌خواهیم با استفاده از Naive Bayes بر اساس توضیحات موجود برای هر مقاله تشخیص دهیم که این مقاله در کدام یک از دسته‌بندی‌های مربوطه جای می‌گیرد. در این مسئله از مدل bag of words استفاده می‌کنیم. به این صورت که هر کلمه را مستقل از جایگاه و ترتیب آن در جمله در نظر می‌گیریم. feature های این مسئله را تعداد هر کلمه در کلاس مربوطه در نظر بگیرید. یعنی هر چه تعداد یک کلمه در یک کلاس بیشتر باشد، احتمال اینکه آن کلمه متعلق به آن کلاس باشد بیشتر است. برای حل این مسئله به صورت کلی از naive bayes استفاده می‌کنیم که مفهوم پشت آن با توجه به مفاهیم احتمالی زیر قابل بحث است.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

Likelihood Class Prior Probability
↓ ↓
Posterior Probability Predictor Prior Probability

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

² <https://github.com/ICTRC/Parsivar>

³ <https://github.com/sobhe/hazm>

۲. در گزارش کار خود، توضیح دهید که هر کدام از (evidence, likelihood, prior, posterior) بیانگر چه مفهومی در این مسئله هستند و چگونه محاسبه می‌شوند.

دقت کنید که نیازی نیست عبارت Evidence در مخرج کسر به صورت مستقیم محاسبه شود. فرآیند کلی‌ای که باید انجام دهید به این شکل است که در ابتدا برای متن‌هایی که در اختیار دارید تعداد هر کلمه را به تفکیک کلاس آن پیدا کنید. با این کار به نوعی مدل خود را train کرده‌اید. حال برای بررسی یک متن جدید از naive bayes استفاده کنید و با استفاده از احتمال قبلی که در مورد هر کلاس داشته‌اید و همینطور استفاده از کلمات موجود در متن و احتمال دیده شدن آن‌ها در آن کلاس احتمال اینکه متن برای کلاس بخصوصی باشد را بیابید.

برای محاسبات، می‌توانید دو ستون عنوان و توضیحات را یکی کنید که در آن صورت یک ستون برای داده خواهید داشت و یک ستون برای دسته‌بندی. همینطور راه دیگری که وجود دارد این است که می‌توانید یکی از دو ستون عنوان یا توضیحات را کاملاً نادیده بگیرید. دو راه گفته شده تنها مثال‌هایی برای ایده دادن به شما هستند و شما می‌توانید کارهای دیگری نیز برای یکی کردن داده‌های در دست و همینطور استفاده از آن‌ها به شکل‌های متفاوت انجام دهید.

در این مورد می‌توانید با عنوان مهندسی ویژگی‌ها^۴ اطلاعات بیشتری نیز کسب کنید و در این پروژه استفاده کنید. فقط توجه کنید که در بخش ارزیابی، باید دقت شما روی داده‌ی تست از حداقل گفته شده بیشتر باشد.

Bigrams

نکته‌ای که در مورد فرآیند ابتدایی naive bayes در قسمت قبل وجود دارد این است که در این مدل، وجود هر کلمه را به تنهایی و بدون توجه به ترتیب کلمات و همینطور دیگر نکات مربوط به بافت^۵ متن در نظر می‌گیریم، در حالی که همانطور که مشهود است نکات گفته شده می‌توانند تاثیر گذار باشند.

در مورد مشکلی که در این قسمت بیان شد، می‌توان گفت اشکال در فرآیندی است که با استفاده از آن token ها را از متن داده شده بیرون می‌کشیم و هر کلمه را به خودی خود بررسی می‌کنیم. در مقابل این کار می‌توان هر دو کلمه که پشت هم آمده‌اند را یک token در نظر گرفت. به بیان دیگر به جای استفاده از unigram هایی که در قسمت قبل در نظر گرفتیم، در این قسمت از bigram ها استفاده کنیم.

۳. دو جمله مثال بنویسید که یک کلمه یکسان در آن‌ها دو معنی متفاوت داشته باشد. استفاده از bigram ها چگونه

به مشخص شدن معنی آن کلمه کمک می‌کند؟ آیا bigram برای مشخص کردن معنی کلمه در مثال شما کافیه یا نیاز به n-gram طولانی‌تری هست؟

اختیاری: استفاده از ترکیب bigram با unigram را در مدل خود اعمال کنید و آن را روی داده‌های خود train کنید و نتیجه بدست آمده را گزارش کنید. دقت کنید طراحی خود را به شکلی انجام دهید که استفاده از هر دو مدل تغییرات زیادی را در کد شما ایجاد نکند به گونه‌ای که اگر خواستید token های ۳ کلمه‌ای و یا حتی ۴ کلمه‌ای را نیز در نظر بگیرید، تغییرات زیادی نیاز نباشند.

^۴ Feature Engineering

^۵ Context

Additive Smoothing

مشکلی که ممکن است در بدست آوردن دسته‌ها به آن برخورد کنید این است که در مقاله‌هایی که مربوط به دسته‌بندی مشخصی هستند، کلمه‌ای وجود داشته باشد که در مقاله‌هایی از دسته‌ای دیگر نباشد و بالعکس، یا حتی به کلمه‌ای در مقاله جدیدی که می‌خواهیم بررسی کنیم برخورد کنیم که در هیچ کدام از مقاله‌های دیده شده در داده train وجود نداشته باشد.

مشکلی که در حالت گفته شده ایجاد خواهد شد به این مسئله خواهد برگشت که اگر به عنوان مثال کلمه‌ی "نمایشگر" تنها در مقاله‌های مربوط به بازی ویدویی باشد ولی در مقاله‌های مربوط به دسته‌بندی دیگر مثل علم و تکنولوژی نباشد، مدل ایجاد شده با قطعیت تشخیص می‌دهد که هر مقاله‌ای که در متن آن کلمه "نمایشگر" وجود دارد مربوط به مقاله‌های بازی ویدویی است در حالی که نتیجه‌گیری انجام شده لزوماً درست نمی‌باشد.

۴. در گزارش خود با در نظر داشتن naive bayes توضیح دهید چرا این اتفاق رخ می‌دهد.

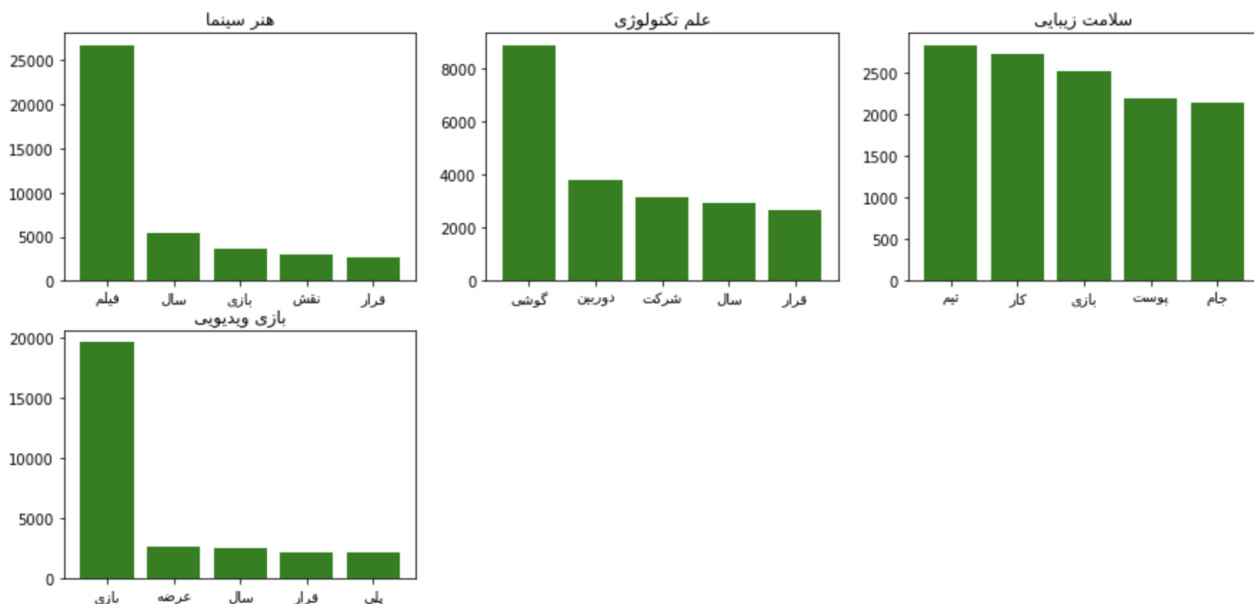
۵. درباره روش Additive Smoothing تحقیق کنید و با پیاده‌سازی آن در پروژه، این مشکل را برطرف کنید.

در گزارش خود این روش را توضیح دهید و بگویید دقیقاً چطور به حل این مشکل کمک می‌کند.

(در بخش ارزیابی، تفاوتی که استفاده از این روش بر دقت می‌گذارد را باید گزارش کنید)

بررسی صحت

۶. با توجه به تعداد کلمات دیده شده مربوط به هر دسته، چهار عدد bar plot رسم کنید که نشان دهد در مقاله‌های هر دسته‌بندی چه کلماتی بیشترین تکرار را دارند. (۴ نمودار که در هر کدام حداقل ۵ کلمه با بیشترین تکرار در آن دسته آمده است. نمودار زیر به عنوان نمونه است و با تصمیماتی که در بخش‌های قبل می‌گیرید می‌توان نتایج متفاوتی گرفت.)



شرط گفته شده مربوط به بالا بودن تعداد تکرار، تنها یکی از راه‌هایی است که می‌توان کلماتی با بیشترین تاثیر در هر دسته را شناسایی کرد. تنها نکته‌ای که خوب است در نظر گرفته شود این است که بعضی کلمات در تمامی دسته‌ها تعداد تکرار بالایی دارند که در نتیجه آن باعث می‌شود تاثیری در شناسایی دسته برای مقاله نداشته باشند. در این مورد، حذف این کلمات از دایره کلمات می‌تواند گزینه خوبی باشد.

فاز سوم: ارزیابی

برای ارزیابی مدل خود باید از 4 معیار زیر استفاده کنید.

$$Accuracy = \frac{Correct\ Detected}{Total}$$

$$Precision = \frac{Correct\ Detected\ Class}{All\ Detected\ Class\ (Including\ Wrong\ Ones)}$$

$$Recall = \frac{Correct\ Detected\ Class}{Total\ Class}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Correct Detected Class: تعداد مقاله‌هایی که به درستی در دسته‌بندی مورد نظر تشخیص داده شده‌اند.
All Detected Class: تعداد تمام مقاله‌هایی که در دسته‌بندی مورد نظر تشخیص داده شده. (حتی اگر به اشتباه)
Total Class: تعداد تمام مقاله‌هایی که در مجموعه داده تست در آن دسته‌بندی خاص بودند.
به جای Class می‌توانید هرکدام از دسته‌بندی‌های موجود مانند vehicles را بگذارید.

۷. در گزارش کار خود توضیح دهید که چرا مقدار Precision و Recall هر کدام به تنهایی برای ارزیابی مدل کافی نیست؟ برای هر کدام مدلی را مثال بزنید که در آن، این معیار مقدار بالایی دارد ولی مدل خوب کار نمی‌کند.

۸. در گزارش کار خود توضیح دهید معیار F1 از چه نوع میانگین‌گیری بین Precision و Recall استفاده می‌کند؟ تفاوت آن نسبت به میانگین‌گیری عادی چیست و در اینجا چرا اهمیت دارد؟

۹. با توجه به اینکه مسئله ما بیشتر از ۲ کلاس دارد در مورد multi-class metrics تحقیق کنید. در گزارش کار خود، سه حالت میانگین‌گیری macro و micro و weighted را شرح دهید. برای تحقیق می‌توانید از این [سایت](https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-cbe8b2c2ca1)⁶ استفاده کنید.

مدل خود را که با استفاده از naive bayes و براساس داده‌ی train ساخته‌اید، روی داده‌ی test که در اختیاران قرار دارد اجرا کنید و برای هر کدام از سطرهای آن، تشخیص مدل‌تان را بدست آورید. سپس براساس آن، معیارهای بالا را برای هر کلاس به صورت تنها و سپس با استفاده از سه نوع میانگین‌گیری گفته شده برای تمام کلاس‌ها محاسبه کنید. (برای

⁶ <https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-cbe8b2c2ca1>

محاسبه معیارها نباید از کتابخانه‌ها استفاده شود اما برای مطمئن شدن از محاسباتتان می‌توانید از توابعی مثل [classification_report](#)⁷ استفاده کنید.

مقدار accuracy و Macro F1 در حالت الف باید به ترتیب بیش از 85 و 85 باشند.

۱۰. در گزارش خود، معیارها را به ازای هر دو حالت زیر به دست آورید (نمونه‌ای از معیارهایی که باید گزارش کنید در ادامه آمده است. توجه کنید که این فقط یک مثال از نحوه ارائه نتایج است).

الف. نتایج با استفاده از Additive Smoothing

ب. نتایج بدون استفاده از Additive Smoothing

	Science Technology	Art Cinema	Video Games	Health Beauty	All Classes
Precision					-
Recall					-
F1-score					-
Accuracy	-	-	-	-	
Macro Avg	-	-	-	-	
Micro Avg	-	-	-	-	
Weighted Avg	-	-	-	-	

۱۱. در گزارش خود، مقادیر بدست آمده در بخش قبل را تحلیل کنید.

۱۲. در گزارش خود ۵ مورد از مقاله‌هایی که در داده‌ی تست هستند و مدل شما دسته اشتباهی برای آن‌ها تشخیص داده است بیاورید. به نظر شما چه بخش یا بخش‌هایی از راه حلی که پیش گرفتیم باعث شده این موارد اشتباه تشخیص داده شوند؟

⁷ https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

نکات پایانی

- دقت کنید که هدف پروژه تحلیل نتایج است بنابراین از ابزارهای تحلیل داده بطور مثال نمودارها استفاده کنید و توضیحات مربوط به هر بخش از پروژه را بطور خلاصه و در عین حال مفید در گزارش خود ذکر کنید.
- نتایج و گزارش خود را در یک فایل فشرده با عنوان `AI_CA3_<#SID>.zip` تحویل دهید. محتویات پوشه باید شامل فایل `jupyter-notebook`، خروجی `html` و فایل های مورد نیاز برای اجرای آن باشد. توضیح و نمایش خروجی های خواسته شده بخشی از نمره این تمرین را تشکیل می دهد. از نمایش درست خروجی های مورد نیاز در فایل `html` مطمئن شوید.
- در صورتی که سوالی در مورد پروژه داشتید بهتر است در فروم درس مطرح کنید تا بقیه از آن استفاده کنند؛ در غیر این صورت توسط ایمیل با طراحان در ارتباط باشید.
- هدف از تمرین، یادگیری شماست. لطفا تمرین را خودتان انجام دهید.