

به نام خدا

پروژه سوم

هوش مصنوعی

علی دارابی - شماره دانشجویی : ۸۱۰۱۰۰۲۶۴

عنوان پروژه : Hidden Markov Model

پیش پردازش و استخراج ویژگی :

۱ - به چه علت نیاز می باشد که نمونه ها دارای طول یکسان باشند؟

برای مقایسه نمونه ها، لازم است در ابتدا آنها را نرمالایز کنیم، بدین معنا که طول آنها باید یکسان باشد. زیرا که تمامی نمونه ها باید به یک فرمت باشند، در مراحل یادگیری ماشین اگر طول نمونه ها متفاوت باشد، ممکن است یک نمونه با اینکه ویژگی های لازم رو برای قرار گرفتن در یک دسته را نداشته باشد، به علت داشتن داده های اضافی در خود، در فرایند تصمیم گیری اختلال ایجاد کند (زیرا امکان می کردن همه ی داده ها بین دو نمونه در هنگام مقایسه آنها وجود ندارد.) و وارد دسته ای شود که در واقعیت متعلق به آن نیست.

۲ - چرا در محاسبه MFCC، فریم های استفاده شده با یکدیگر هم پوشانی دارند؟

هم پوشانی فریم ها به جای تغییرات ناگهانی بین فریم ها، انتقال همواری بین ضرایب MFCC را در طول زمان ایجاد می کند. در نتیجه این عمل ویژگی ها را پیوسته تر و طبیعی تر می کند. دلیل دوم این است که هم پوشانی کمک میکند که اطلاعات از دست رفته در لبه های هر فریم به حداقل برسد و نمایش MFCC را بهتر میکند. همچنین با پرس های کوتاه تر بین فریم ها وضوح بهتری برای ثبت تغییرات سریع زمانی در صدا ایجاد میشود. در آخر میتوان گفت که افزایش تعداد فریم های استخراج شده، داده های MFCC بیشتری را برای آموزش مدل فراهم می کند.

۳ - چرا در اکثر پروژه های مرتبط با صوت، تنها از ۱۲ یا ۱۳ ضریب ابتدایی MFCC استفاده میشود؟

دلیل اول : ضرایب پایین، دارای مفیدترین اطلاعات طیفی هستند در حالی که ضرایب بالاتر نشان دهنده جزئیات دقیق و نویز هستند. دلیل دوم : استفاده از ضرایب کمتر، پیچیدگی مدل های یادگیری ماشین را کاهش می دهد و امکان آموزش سریع تر و بهتر را برای ما ایجاد می کند. و همچنین میتوان این چنین گفت که ضرایب کمتر به معنی زمان محاسباتی و توان کمتر برای مجموعه داده های صوتی بزرگ است. در نتیجه استفاده از ۱۲ یا ۱۳ ضریب ابتدایی به عنوان یک قرارداد ایجاد شد، زیرا که این ضرایب در عین فشرده بودن (تعداد کم ضرایب) حاوی اطلاعات مفید از اجزای کلیدی داده های صوتی هستند.

آشنایی با HMM :

۱ - توضیح دهید منظور از State ها و Observation ها چیست؟ در این تمرین State ها کدامند و Observation چگونه بدست می آید؟

State ها: به مجموعه ای از حالت هایی که گفته میشود که پنهان هستند یا احتمالاً هنوز مشاهده نشدند. سیستم می تواند در یک زمان در آنها قرار بگیرد. آنها را میتوان از داده های قابل مشاهده استنباط کرد. (مثلاً در مثال پیشبینی آب و هوا ما نمیتوانیم تغییرات آب هوایی را بصورت دقیق ببینیم ولی میتوانیم از داده های قابل مشاهده مثل بارانی یا آفتابی بودن هوا استنباط کنیم).

Observation ها: به خروجی ها یا ویژگی هایی که برای ما قابل مشاهده هستند گفته میشود. این مشاهدات همان چیزی است که ما در واقعیت می بینیم یا ثبت می کنیم و شواهد هایی در مورد حالات (state) ارائه می دهیم. (مثلاً در مثال آب و هوا داده های قابل مشاهده برای ما آفتابی بودن یا بارانی بودن هوا است).

در این تمرین، State ها برای ما معنی خاصی ندارند و باید خودمان با انجام تست، بهترین تعداد state را بیابیم. و به عنوان پارامتر مسئله در نظر میگیریم.

در این تمرین، Observation های ما، همان ویژگی هایی هستند که به کمک ضرایب MFCC از نمونه هایی که در اختیار ما قرار گرفته، استخراج کردیم. (برای هر ژانر، ۱۰۰ نمونه در اختیار ما قرار گرفته که برای همه آنها ضرایب MFCC را محاسبه کردیم).

۲ - مدل های HMM را میتوان بر اساس میزان وابستگی میان State های پنهان دسته بندی کرد. مدلی که در این تمرین به پیاده سازی آن می پردازید، یک مدل First-Order HMM است. دلیل نامگذاری آن و همچنین ویژگی های آن را بررسی کنید و تفاوت آن با مدل های دیگر در این دسته بندی را بیان کنید.

دلیل نامگذاری: مدل های First-Order HMM، به این دلیل مرتبه اول نامیده میشوند که احتمال یک حالت (State) در زمان t ، فقط به حالت (State) در زمان $t-1$ بستگی دارد.

ویژگی های آن: همانطور که گفته شد مهم ترین ویژگی First-Order HMM، این است که احتمال یک حالت (State) در زمان t ، فقط به حالت (State) در زمان $t-1$ بستگی دارد. ویژگی دیگر آن این است که تعداد حالات آن محدود است و احتمال مشاهده یک خروجی خاص در یک زمان معین فقط به وضعیت آن زمان بستگی دارد.

تفاوت با مدل های دیگر: در HMM مرتبه صفر، حالت ها در طول زمان مستقل هستند. هیچ انتقالی بین حالت های پنهان در طول زمان مدل سازی نشده است. در HMM های مرتبه بالاتر (مرتبه بالاتر از یک) هر حالت به چندین حالت قبلی بستگی دارد برای مثال در یک HMM مرتبه دوم، هر حالت به دو حالت گذشته بستگی دارد.

۳ - درباره HMM تحقیق کنید و توضیح دهید که این مدل برای بررسی و تحلیل چه پدیده هایی مناسب است؟ چرایی این موضوع را توضیح دهید.

HMM ها برای رسیدگی به پدیده هایی مناسب اند که وضعیت سیستم در حال مدل سازی به صورت مستقیم قابل مشاهده نیست. زیرا که HMM ها ماهیت احتمالی انتقال بین حالت های نهفته را به تصویر می کشند. آنها می توانند احتمال انتقال از یک حالت به حالت دیگر را در طول زمان مدلسازی کنند. با توجه به این قابلیت ها، HMM ها در حوزه هایی مثل تشخیص گفتار، تشخیص ناهنجاری، مدل سازی مالی، تجزیه و تحلیل رفتار مشتری، و... کاربردی هستند. دلایلی که HMM ها را برای تجزیه و تحلیل چنین پدیده هایی مناسب می کند، نمایش هر دو انتقال حالت پنهان و خروجی های قابل مشاهده است که در طی مراحل زمانی تولید می کنند.

۴ - مدل HMM نیز مانند هر مدل دیگری دارای مزایا و معایبی است که آن را ویژه می کند. مزایا و معایب این مدل را بررسی کرده و هرکدام را مختصراً توضیح دهید.

مزایا: HMM ها می توانند سیستم هایی با حالت های نهفته را مدل سازی کنند که حالت هایی هستند که نمی توان به صورت مستقیم مشاهده کرد اما می توان از داده های قابل مشاهده استنباط کرد. HMM ها الگوریتم های efficient برای آموزش و استنتاج دارند که آنها را برای مجموعه داده های بزرگ مناسب می کند. HMM ها را می توان با تنظیم پارامترها و ساختار مدل با طیف وسیعی از کاربردها سازگار کرد.

معایب: HMM ها برای استنباط حالات نهفته ، به شواهد قابل مشاهده متکی هستند. اگر این اطلاعات دست کاری شده باشند یا دارای خطا باشند، ممکن است عملکرد مدل به مشکل بخورد. تخمین پارامترهای یک HMM می تواند چالش برانگیز باشد، به ویژه برای مدل های پیچیده که مدل های نهفته زیادی دارند. HMM ها اثرات ورودی ها و متغیرهای خارجی را نادیده می گیرند.

۵ - انواع مختلفی از مدل های HMM وجود دارد. درباره آنها تحقیق کنید و چند مورد را بطور مختصر بررسی کنید.

انواع مختلف : HMM گسسته، پیوسته، گاوسی، چندبعدی، Hierarchical و Input-Output

مدل گسسته: اساسی ترین و رایج ترین مدل است. هر دو حالت نهفته و قابل مشاهده مقادیر گسسته هستند. در تشخیص گفتار و غیره استفاده می شود.

مدل پیوسته: حالت های نهفته مقادیر پیوسته ای هستند که انعطاف پذیری بیشتری را برای نمایش فراهم می کنند. اگر حالت های سیستم به طور طبیعی پیوسته باشند کاربردی است. برای ردیابی حرکت و غیره استفاده می شود.

مدل چندبعدی: به جای یک متغیر حالت واحد در طول زمان، چندین متغیر حالت را در یک بردار حالت حفظ می کند. برای مدل سازی سیستم های چند بعدی مناسب است.

ارزیابی و تحلیل :

نحوه محاسبه هریک از معیار ها :

باتوجه به جدول confusion که در notebook رسم کردیم، در ابتدا باید مقادیر TN و TP و FP و FN را برای هر ژانر از روی جدول مشخص کنیم، سپس مقادیر معیار های خواسته شده برای هر مدل را بدست می آوریم.

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN}$$

برای بدست آوردن accuracy ما باید تعداد تمام پیشبینی های درست (true negative + true positive) را تقسیم بر تعداد کل پیشبینی ها کنیم .

$$Precision = \frac{TP}{TP + FP}$$

برای بدست آوردن precision ما باید تعداد پیشبینی های درست و موافق (true positive) را تقسیم بر تعداد کل پیشبینی های موافق (positive = true positive + false positive) کنیم.

$$Recall = \frac{TP}{TP + FN}$$

برای محاسبه ی recall ، ما باید تعداد پیشبینی های درست و موافق (true positive) را تقسیم بر تعداد پیشبینی های موافق و درست (true positive) بعلاوه ی تعداد پیشبینی های نادرست و غیر موافق کنیم (false negative).

$$F1 \text{ Score} = 2 * \frac{Precision * Recall}{Precision + Recall}$$

برای محاسبه ی F1 Score، طبق فرمول داده شده باید با استفاده از مقادیر Precision و Recall، آن را محاسبه کنیم.

سوالات :

۱ - توضیح دهید که هر کدام از معیارها چگونه مدل را ارزیابی میکنند.

Accuracy : بیانگر تعداد کل سمپل هایی است که به درستی classified شده اند نسبت به تعداد کل سمپل ها است. Accuracy معیار ساده و قابل فهمی است، اما در برخی موارد می تواند گمراه کننده باشد. به عنوان مثال، اگر یک مدل همیشه رایج ترین کلاس را پیش بینی کند، دقت بالایی خواهد داشت، حتی اگر در واقع از داده ها چیزی یاد نگرفته باشد، اگر مجموعه داده های ما متعادل نباشد، ممکن است معیار خوبی نباشد.

Precision : بیانگر تعداد پیش بینی های درست و موافق (مثبت) به تعداد کل پیش بینی های موافق (مثبت) است. نشان میدهد چند درصد از مواردی که مدل ما مثبت پیش بینی کرده، واقعا مثبت هستند.

Recall : بیانگر تعداد پیش بینی های درست و مثبت به تعداد پیش بینی های درست و مثبت بعلاوه پیش بینی های نادرست و منفی است.

F1 Score : باتوجه به فرمول آن، تنها زمانی F1 Score ، ۱ میشود که هم Precision و هم Recall مقدار ۱ داشته باشند. یک معیار جامع برای ارزیابی مدل است و چون هم Precision و هم Recall رت در نظر میگیرد، میتواند تصویر بهتری از عملکرد مدل به ما ارائه دهد. و در واقع harmonic mean بین Precision و Recall است و از Accuracy بهتر است.

۲ - تفاوت میان Precision و Recall را بیان کنید و توضیح دهید چرا هر کدام به تنهایی برای ارزیابی مدل کافی نیست ؟ برای هریک مثالی بیاورید که در آن، این معیار مقدار بالایی دارد اما مدل عملکرد خوبی ندارد.

Precision تعداد پیشبینی های درست و مثبت به تعداد کل پیشبینی های موافق مثبت است در صورتی که Recall بیانگر تعداد پیشبینی های درست و مثبت به تعداد پیشبینی های درست و مثبت بعلاوه پیشبینی های نادرست و منفی است و در واقع منجر کسر ها متفاوت است. اگر تنها به یکی از آنها تکیه کنیم ، ممکن است ارزیابی ما به درستی صورت نگیرد و ممکن است تصویر ناقصی از عملکرد مدل ارائه شود. مثال : فرض کنید یک فیلتر اسپم داریم و به عنوان ورودی ۱۰۰ ایمیل داریم که واقعا ۶۰ تا از آنها اسپم هستند، مدل ما ۲۰ تا از ایمیل هارا اسپم تشخیص میدهد و از بین این ۲۰ ایمیل ۱۸ تای آنها واقعا اسپم هستند در نتیجه $Precision = 90\%$ ولی $Recall = 18 / 60 = 30\%$ در نتیجه این مدل عملکرد خوبی ندارد با اینکه precision مقدار بالایی دارد.

مثال دیگر : فرض کنید یک فیلتر اسپم داریم و به عنوان ورودی ۱۰۰ ایمیل داریم که واقعا ۲۰ تا از آنها اسپم هستند، مدل ما ۶۰ تا از ایمیل هارا اسپم تشخیص میدهد و از بین این ۶۰ ایمیل ۲۰ تای آنها واقعا اسپم هستند در نتیجه $Recall = 20 / 60 = 33\%$ ولی $Precision = 20 / 60 = 33\%$ در نتیجه این مدل عملکرد خوبی ندارد با اینکه Recall مقدار بالایی دارد.

۳ - معیار F1 از چه نوع میانگین گیری استفاده میکند؟ تفاوت این نوع میانگین گیری با میانگین گیری عادی چیست و در اینجا چرا اهمیت دارد؟

این معیار از میانگین گیری Harmonic استفاده میکند.

Normal mean: $(x_1 + x_2 + \dots + x_n) / n$

Harmonic mean: $n / (1/x_1 + 1/x_2 + \dots + 1/x_n)$

تفاوت میانگین Harmonic و عادی در این است که میانگین Harmonic ، وزن بیشتری به داده های کوچکتر میدهد و این در معیار F1 مهم است، زیرا که باعث میشود بهتر بتواند Trade-off بین Precision و Recall را نشان دهد.

۴ - همانطور که میدانید، در این مسئله بیش از دو کلاس داریم، در مورد Multi-Class Metrics تحقیق کنید و سه حالت میانگین گیری Macro و Micro و Weighted را شرح دهید.

حالت میانگین گیری Macro: بدون توجه به فراوانی کلاس ها در داده ها، با همه کلاس ها به طور یکسان رفتار می کند. معیار ها (Accuracy, Precision, Recall, F1 Score) را برای هر کلاس محاسبه می کند و سپس این مقادیر را در همه کلاس ها میانگین می گیرد.

حالت میانگین گیری Micro: به همه ی داده ها به طور همزمان نگاه میکنیم، برای مثال TP در اینجا، تمام نمونه های پیش بینی شده درست را به عنوان True Positives در نظر می گیریم. از آنجایی که ما همه کلاس ها را با هم بررسی می کنیم، هر پیش بینی نادرست برای کلاسی که پیش بینی شده بود یک FP است. (مجموع درایه های غیر قطری در ماتریس confusion). در نتیجه در این حالت میانگین گیری مقدار تمام معیار های F1 Score, Precision, Recall, Accuracy با هم برابر هستند.

حالت میانگین گیری Weighted: ما برای محاسبه ی هر معیار، به هر کلاس با توجه به فراوانی داده های آن، به آن وزن می دهیم. برای مثال اگر ۲۰ داده داشته باشیم که ۸ تا از کلاس A و ۷ تا از کلاس B و ۵ تا از کلاس C باشند، برای محاسبه ی F1 Score بدین شکل عمل میکنیم:

$$\text{Weighted F1 Score} = 1/20 * (8 * \text{F1 Score A} + 7 * \text{F1 Score B} + 5 * \text{F1 Score C})$$

۵ - برای نتایج بدست آمده، Confusion Matrix رسم کنید و معیار های بالا را برای هر کلاس به صورت جداگانه و سپس با استفاده از سه نوع میانگین گیری گفته شده برای تمام کلاس ها محاسبه کنید.

در نوت بوک تمام مقادیر خواسته شده بدست آورده شد و پرینت شد.

۶ - مقادیر بدست آمده برای معیار های ارزیابی را تحلیل کنید.

برای بخش اول (استفاده از کتابخانه آماده):

همانطور که انتظار داشتیم مقادیر بدست آمده برای معیار ها، مقدار های نسبتا بالایی هستند. Accuracy بدست آمده برای ماتریس Confusion برابر با ۸۵٪ بود که مقدار خوب و نسبتا بالایی است. مقدار هایی بدست آمده برای معیار های Precision و Recall هم تقریبا برای هر ۴ مدل خوب بود و نسبتا بالا بودند، میدانیم هر کدام از این دو معیار به تنهایی نمیتوانند معیار خوبی برای ارزیابی باشند و اگر مقدار یکی بسیار پایین باشد و دیگری بسیار بالا، مدل ما عملکرد خوبی ندارد. ولی همانطور که انتظار میرفت در اینجا و با استفاده از کتابخانه آماده HMM، مقادیر بدست آمده این دو معیار برای هر یک

از ۴ مدل به طور تقریبی بالا بودند، سپس معیار F1 Score را بدست آوردیم که برای هر ۴ مدل بالای ۷۰٪ بود و مقدار تقریباً بالایی بود، برای مدل pop از دیگر مدل ها بهتر بود و بالای ۹۰٪ بود و در کل میتوان گفت که بین این ۴ معیار برای ارزیابی مدل، F1 Score از همه بهتر میتواند عملکرد مدل ما را نمایش دهد.

برای بخش دوم :

همانطور که انتظار میرفت، مدل ما نسبت به بخش قبلی اصلاً دقت مناسبی ندارد، در واقع مدل ما اصلاً نمیتواند سمپل های blues را تشخیص دهد، با این حال اگر Accuracy را فقط برای مدل blues حساب کنیم، مقدار بالایی دارد، که نشان دهنده ی این است که این معیار ، معیار زیاد خوبی برای ارزیابی نیست. ۳ مدل دیگر (hiphop, metal, pop) عملکرد نسبتاً خوبی دارند، با اینکه عملکرد آنها نسبتاً به مدل های کتابخانه آماده ضعیف تر هستند. مدل pop بهتر از همه عمل میکند و معیار F1 Score آن نزدیک به ۸۰٪ است که نشان میدهد عملکرد نسبتاً خوبی دارد، مدل های hiphop و metal معیار F1 Score نزدیک به ۴۰٪ دارند که نشان میدهد عملکرد مدل آنها زیاد خوب نیست و با اینکه مقدار Recall آنها بالاست، مقدار Precision آنها پایین است. مدل blues هم که به کل اشتباه کار میکند و هیچ کدام از سمپل هارا تشخیص نمیدهد، به همین علت معیار های ارزیابی برای آن بی معنا هستند.

۷ - ممکن است نتایج شما در بخش اول و دوم فرق کند و مدل آماده (که در بخش اول از آن استفاده کردید) نتایج متفاوت و دقت بالاتری نسبت به مدل طراحی شده توسط شما داشته باشد. این اختلاف ممکن است چه دلایلی داشته باشد؟ درباره عوامل تاثیرگذار بر روی این اختلاف دقت تحقیق کنید.

این اختلاف ممکن است دلایل گوناگونی داشته باشد، از مهم ترین آنها میتوان به این عوامل اشاره کرد :

بهینه سازی : قطعاً کتابخانه hmmlearn با استفاده از پارامتر های بهتر برای بهینه سازی، مدل ساخته شده عملکرد بهتری نسبت به مدلی که خودمان ساختیم ارائه میدهد.

پیاده سازی : الگوریتم های استفاده شده در کتابخانه hmmlearn ممکن بهینه تر پیاده سازی شده و در نتیجه دقت بالاتری برای مدل ساخته شده به ما ارائه دهند.

در نتیجه میتوان گفت که کلاس HMM که خودمان نوشتیم با توجه به اینکه قطعاً ما از الگوریتم های متفاوتی استفاده کردیم و بهینه سازی در کدمان ممکن است بسیار پایین تر از مقدار بهینه سازی کتابخانه از پیش تعریف شده ی hmm باشد، دقت مدل طراحی شده توسط ما بسیار پایین تر دقت مدل ساخته شده با کتابخانه hmm باشد.

حتی زمانی که تعداد iteration هارا افزایش میدهیم، باز هم نمیتوانیم به دقت کتابخانه hmm دست یابیم.