باسمه تعالی

گزارش پروژه 2 NLP

مهدی فیض الهی 810199356

1. Spam detection

در این قسمت از روش multinomial naïve bayes استفاده کردم. اسم فایل spam.pyمی­باشد. مراحل کار در کل به این شرح است:

1. اول از همه با استفاده از chardet، encoding فایل csv را به دست آوردم. Encoding، Windows-1252 بود.
2. دو ستون v1و v2 را از dataframe استخراج کردم. و کل داده را به lowercase تغییر دادم.
3. وضعیت ham را کلاس 1 و وضعیت spam را کلاس 2 در نظر گرفتم.
4. در مرحله پیش پردازش، کلمه های stopword و punctuation ها رو پاک کردم.
5. با استفاده از کتابخانه CountVectorizer و TfIdfTransformer و MultinomialNB یک طبقه بند درست کردم و predict کردم.

نتایج:

|  |  |  |
| --- | --- | --- |
| Ham detect | Spam detect |  |
| 33 | 112 | Spam |
| 970 | 0 | Ham |

1 - ماتریس confusion در حالت دارای پیش پردازش

بنابراین:

True Positive = 112 , True Negative = 970 , False Negative = 33, False Positive = 0

Accuracy = 112 + 970/112 + 970 + 33 = 0.97

Precision = 112/112 + 0 = 1

Recall = 112/112 + 33 = 0.77

F1-score = 112/112 + ½(0 + 33) = 0.872

حال، موقعی که پیش پردازش نداریم هم برنامه را اجرا می­کنیم:

|  |  |  |
| --- | --- | --- |
| Ham detect | Spam detect |  |
| 43 | 102 | Spam |
| 970 | 0 | Ham |

2- ماتریس confusion بدون پیش پردازش

True Positive = 102 , True Negative = 970 , False Negative = 43, False Positive = 0

Accuracy = 102 + 970/102 + 970 + 43 = 0.96

Precision = 102/102 + 0 = 1

Recall = 102/102 + 43 = 0.70

F1-score = 102/102 + ½(0 + 43) = 0.826

مشاهده می­کنیم که دقت و precision و F1 کاهش یافت.

مشاهده می­کنیم که recall بالاست بنابراین می­توانیم راحت باشیم که هیچ spam ی را به عنوان ham تشخیص نداده­ایم.

1. Lie Detection

در این بخش، یک عبارت داریم و چند ویژگی دیگر، شامل: sentiment, anger, fear, disgust, sad, joy

برای این موارد به این شکل عمل کردم:

1. عبارت را یک بار پیش پردازش کردم و بار دیگر بدون پیش پردازش دنبال کردم.
2. در انتهای هر عبارت، با استفاده از ویژگی ها، چند کلمه با فاصله اضافه کردم. برای مثال: $Neg برای منفی بودن معنا. و برای 5 حس دیگر، هر عدد را به بازه های 0.2 تقسیم کردم و یک عبارت با فاصله به عبارت اضافه کردم. برای مثال: $ANG\_2 برای anger های بین 0.2 تا 0.4.
3. طبق بخش قبل، یعنی spam detection، ادامه دادم و به نتایجی رسیدم.

نتایج:

|  |  |  |
| --- | --- | --- |
| Truth detect | Lie detect |  |
| 406 | 147 | Lie |
| 620 | 94 | Truth |

3 - ماتریس confusion با پیش پردازش برای دروغ سنجی

بنابراین:

True Positive = 147 , True Negative = 620 , False Negative = 406, False Positive =94

Accuracy = 147 + 620/147 + 620 + 94 + 406 = 0.60

Precision = 147/147 + 94 = 0.61

Recall = 147/147 + 406 = 0.26

F1-score = 147/147 + ½(94 + 406) = 0.37

متاسفانه، نتایج آن طور که باید، خوب نیستند. با این که از ویژگی های مختلف استفاده کردیم. در این قسمت از پیش پردازش استفاده کردیم.

نتایج:

|  |  |  |
| --- | --- | --- |
| Truth detect | Lie detect |  |
| 424 | 129 | Lie |
| 629 | 85 | Truth |

4 - ماتریس confusion بدون پیش پردازش برای دروغ سنجی

بنابراین:

True Positive = 129 , True Negative = 629 , False Negative = 424, False Positive =85

Accuracy = 129 + 629/129 + 629 + 85 + 424 = 0.60

Precision = 129/129 + 85 = 0.60

Recall = 129/129 + 424 = 0.23

F1-score = 129/129 + ½(85 + 424) = 0.34

متاسفانه، نتایج آن طور که باید، خوب نیستند. با این که از ویژگی های مختلف استفاده کردیم. در این قسمت از پیش پردازش استفاده نکردیم.

مشاهده می­کنیم که وضعیت در صورت داشتن پیش پردازش، کمی بهتر است ولی متاسفانه طبقه بند خوبی به دست نیامد.