گزارش پروژه ی دوم درس NLP فاز دوم

قسمت اول

الف. پیش پردازش ها

HTML cleanup:

در این مرحله داده های لازم را از آرشیو سایت خبرنگاران جوان جمع آوری کرده ام و چون در آرشیو، داده ها مرتب شده و منظم بود نیازی به روش های استخراج متن از وبسایت نبود و به طوری دستی از سایت داده ها را کپی کردم.

Sentence Breaking:

برای جدا کردن جمله ها، از تابع GetSentences در کلاس wordmap استفاده می کنم که در این تابع به کمک کلاسSentenceTokenizer از کتابخانه ی Nhazm جمله های متن را تشخیص داده و در data structure دیکشنری ذخیره می کنم و به عنوان خروجی تابع برمی گردانم.

Number/Date/... Handling and Tokenization:

در کلاس های wordmap و naiveBayes متدی را تعریف کردم که متن ورودی را بر اساس جدا کننده های زیر به صورت آرایه ای از tokenها در خروجی بر می گرداند.

```
char[] delimiters = new char[] {
                ',', '"', ')', '(', ';', '.', '\n', '\r', '\t',
                '>', '<', ':', '=', '\'', '[', ']',' ','!','#',
                '؛','،','-','$','%','&','\'', '+','/','?','؟',
                '{','}','»','«','1','2','3','4','5','6','7','8',
                '9','0','۱','۲', '۳','۴','۵','۶','۷','۸','۹','۰'};
```

Normalization:

در کلاس naiveBayes متد normalize را تعریف کردیم که در آن از کلاس normalizer کتابخانه ی Nhazm برای نرمال کردن متن ورودی استفاده می کنیم.

Test/Train Split up:

داده های استخراج شده از سایت برای خبر های ورزشی و سیاسی حدود 35 هزار کلمه بوده است که 10 در صد را برای تست از هر دو کلاس جداکردیم یعنی حدود 3500 کلمه که 10 فایل تست برای کلاس خبر های ورزشی و 10 فایل تست هم برای کلاس خبر های سیاسی ایجاد کرده ایم. برای هر یک از کلاس ها یک فایل train که حدود 32000 کلمه است را درنظر گرفته ایم. فایل ها در فولدر data قرار داده شده است.

ب. Naive Bayes Text Classifier:

کلاس بندی فایل های تست در تابع naivebayes و در متد classifier انجام می شود. به این صورت که Token های منحصر به فرد هر دو کلاس را از داده های train با متد uniqueToken بدست می آوریم و سپس لگاریتم احتمال هر کلمه را در هر یک از کلاس ها با smoothing با متدCalulateWordLogProbs بدست می آوریم. احتمال هر یک از کلاس ها یک دوم است زیرا تعداد فایل های train برای هر یک از کلاس ها یکی است. سپس هر یک از فایل های تست را می خوانیم و tokenize می کنیم و احتمال آن فایل تست را برای هر یک از کلاس ها محاسبه می کنیم. احتمال هر کدام از کلاس ها که بیشتر شد فایل تست متعلق به آن کلاس خواهد بود.

ج. محاسبه و گزارش Precision/Recall د. محاسبه و گزارش موثرترین feature ها

در این مرحله دو فایل تست برای هر یک از کلاس هایمان داریم که جمله به جمله چک می کنیم که مربوط به کدام یک از کلاس ها هست که نمونه ای از نتایج را در زیر مشاهده می کنید.

جهانی جام میزبان ایران، /شد لاتزیو یک شماره هدف هم باز سردار / ۲۰۲۰ المپیک در ایران برداری‌وزنه سهمیه رفتن دست از خطر فوتبال ۲۰۲۲ is class 1
Class1: -168.910128977227 Class2: -199.397339197672
TopFeature: جهانی لاتزیو المپیک فوتبال جام
داد دست از را تاتنهام با دیدار منچسترسیتی ستاره is class 1
Class1: -56.3708814125707 Class2: -68.165091926898
TopFeature: دیدار دست تاتنهام منچسترسیتی ستاره
تصاویر + کویت در ها استقلالی تمرین دومین is class 1
Class1: -44.1531157730665 Class2: -52.1235273888054
TopFeature: کویت تصاویر دومین هااستقلالی تمرین
پالاتونو رم از ایران های‌واترپلوئیست شکست is class 1
Class1: -45.7137311316083 Class2: -49.9196281878909
TopFeature: های‌واترپلوئیست ایران از شکست رم
کنم نمی حلال را داور :بارانی is class 1
Class1: -43.0529873674173 Class2: -47.0593897571351
TopFeature: بارانی را حلال کنم‌نمی داور
زدمی تهران نفت مقابل دیدار در را اول حرف استرس :مهاجری is class 1
Class1: -78.6764781831639 Class2: -87.8241340274989
TopFeature: حرف مهاجری نفت اول مقابل
داشتیم پدیده با ای پیچیده و سخت بازی :افاضلی is class 1
Class1: -58.3578280254858 Class2: -67.045327634232
TopFeature: سخت داشتیم پدیده افاضلی بازی

ریلی همکارهای توسعه و سیاسی همکاری تفاهم یادداشت is class 2
Class1: -75.9770311674128 Class2: -63.9771060989333
TopFeature: همکاری یادداشت تفاهم سیاسی توسعه
است یافته بهبود گذشته ماه دو به نسبت داخلی رسانهای‌پیام وضعیت is class 2
Class1: -90.620330303193 Class2: -82.8176657077984
TopFeature: بهبود است نسبت دو داخلی
یاسوج تهران مسافربری هواپیمای سقوط سانحه پی در باقری سرلشکر تسلیت پیام is class 2

Class1: -110.047880514935 Class 2: -94.730905801837
TopFeature: یاسوج سانحه سرلشکر هواپیمای تسلیت
is class 2 است ضروری جناحی و قشر هر از امنیت مخل عوامل با برخورد /نداریم دراویش با مشکلی
Class1: -131.251730054269 Class2: -119.659521492831
TopFeature: است ضروری امنیت برخورد دراویش
is class 2 عفرین در ترکیه تجاوزکارانه عملیات آغاز از سوری شهروند 175 شدن کشته
Class1: -101.398094302061 Class2: -87.9420128274217
TopFeature: شهروند عملیات ترکیه کشته عفرین
is class 2 دارد حضور شجاعانه و غیرتمندانه اسلامی، میهن از دفاع های‌جبهه تمامی در ارتش
Class1: -107.180747857998 Class2: -97.0231083999568
TopFeature: میهن دفاع تمامی اسلامی ارتش
is class 2 نکنید صحبت مجلس با غرور با/است داشته افزایش درصد 500 کارگران درمانی هزینه
Class1: -123.120842915763 Class2: -109.492341522507
TopFeature: کارگران درمانی ۰۰ درصد مجلس
is class 2 کشید تصویر به را صهیونیسم و سلطه نظام کارانه تجاوز خوی آمریکا اقدام
Class1: -113.613204716686 Class2: -101.092469682492
TopFeature: سلطه اقدام تجاوز آمریکا نظام
is class 2 تاجیکستان در ترکمنستانی همتای با ظریف رایزنی
Class1: -61.3223812697264 Class2: -53.2494477194729
TopFeature: ترکمنستانی همتای تاجیکستان رایزنی ظریف

....

precision = 0.996124031007752
recall = 0.984674329501916
number of statements = 520

قسمت دوم

الف. آماده سازی داده برای ورودی VW

برای اینکه داده ها را برای VW آماده کنیم در کلاس vowpalWabbit در متد convertToVW داده های train هر دو کلاس ورزشی و سیاسی را به صورت یک درمیان و همراه با label در یک فایل تست به نام inputVW.txt ذخیره می کنیم.

ب. تولید و اجرای دستور مناسب VW برای آموزش و امتحان.

از دستور زیر برای ساختن مدل در VowpalWabbit استفاده کردیم:

```
vw -d trainVW.txt --loss_function logistic --ngram 1  -f predictor.vw
```

و از دستور زیر برای امتحان داده های تست روی مدل ساخته شده استفاده می کنیم:

```
vw - testVW.txt -t -i predictor.vw -p predictions_1gram.txt
```

نتایج به دست آمده در فایل ذخیره شده است که از این فایل استفاده می کنیم و precision و recall را محاسبه می کنیم.
نتایج به صورت زیر است:

1gram:

```
Precision = 0.965779467680608
Recall = 0.980694980694981
```

2gram:

```
Precision = 0.962121212121212
Recall = 0.980694980694981
```

3gram:

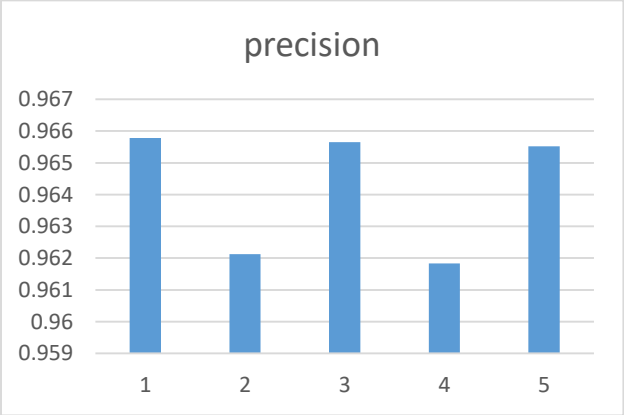
```
Precision = 0.965648854961832
Recall = 0.976833976833977
```

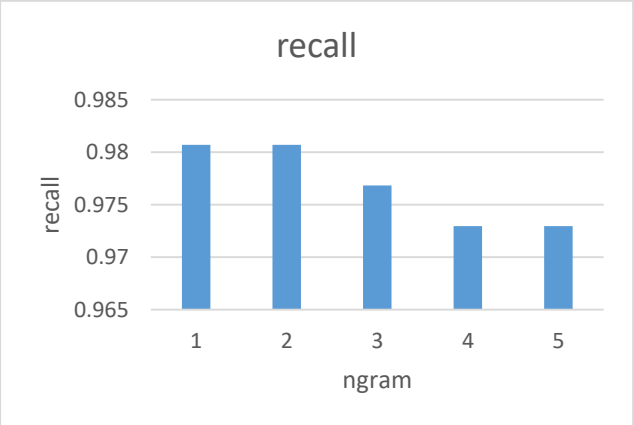
4gram:

```
Precision = 0.961832061068702
Recall = 0.972972972972973
```

5gram:

```
Precision = 0.96551724137931
Recall = 0.972972972972973
```





این کار را برای 10 گرام و 20 گرام هم انجام دادیم که precision تغییر چندانی با نتایج بالا نداشت ولی مقدار recall در حدود 0.96 بدست آمد. توجه شد که مرز انتخاب کلاس در این محاسبات صفر بوده است. یعنی اگر اعدادی که در فایل predictions_1gram.txt است بالا تر صفر بوده است کلاس 1+ و اگر کمتر از صفر بوده است کلاس 1- تلقی شده است.

حال مرز انتخاب کلاس را برای فایل predictions_1gram.txt را به صورت زیر تغییر نتایج بدست آمده به صورت زیر می باشد:

Boundary = 0

```
Precision = 0.965779467680608
Recall = 0.980694980694981
```

Boundary = 1

```
Precision = 1
Recall = 0.995614035087719
```

Boundary = 2

```
Precision = 1
Recall = 1
```

Boundary = 3

```
Precision = 1
Recall = 1
```

Boundary = 4

```
Precision = 1
Recall = 1
```