پروژه ی پایانی درس NLP : Sentiment analysis

موضوع من برای این پروژه بررسی **محبوبیت دکتر آذری جهرمی وزیر ارتباطات و فناوری اطلاعات قبل و بعد از فیلتر تلگرام** است. تلگرام در تاریخ ۱۰اردیبهشت سال ۱۳۹۷ در ایران فیلتر شد. برای جمع آوری داده از زبان پایتون و برای sentiment analysis از زبان c# استفاده کردم.

۱) جمع آوری داده:

برای جمع آوری تصمیم گرفتم که توییت ها و کامنت هایی که در مورد وزیر در توییتر است را با استفاده از زبان پایتون بدست آورم. برای این کار از tweepy،API استفاده کردم. از دو نمونه کد استفاده کردم که با هیچ کدام به نتیجه مطلوب نرسیدم. بعد از سرچ زیاد متوجه شدم که در اولین نمونه کد که در فایلgettweet1.py موجود است مشکل این است که تنها توییت هایی که از همین لحظه تا یک هفته ی گذشته پست شده است را می توان بدست آوردکه این برای این کاری که ما می خواهیم مناسب نیست. در نمونه ی کد دوم که در فایل gettweet2.py می باشدبا اینکه از encode utf-8 استفاده شده است ولی باز هم خروجی مناسب را در فایل نگرفتم. علاوه بر این، مشکل کد قبلی همچنان در این کد پابرجاست. به همین دلیل تصمیم گرفتم که کامنت هایی که در زیر پست های دکتر آذری جهرمی است را با استفاده ازInstagramAPI در زبان پایتون بگیرم. بعضی از پست های وزیر را به صورت رندوم انتخاب کردم و کامنت های زیر این پست ها را گرفتم. فایل های داده را به صورت تفکیک زمانی در فولدر data می توانید مشاهده کنید.

کل کامنت های قبل از فیلتر تلگرام در فایل before.txt و کل کامنت های بعد از فیلتر تلگرام در فایل after.txt می باشد.

۲) پردازش و تمیز کردن داده ها:

برای این مرحله تمامی ایموجی ها و منشن هایی که در کامنت ها بود؛ حذف شد. که برای این کار در ابتدا کلمات را با استفاده از علائم زیر توسط تابع split جدا کردیم. سپس با یک regex تنها کلمات فارسی را بدست می آوریم که در واقع با این کار ایموجی و منشن ها هم از داده حذف خواهد شد.

```
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'};
```

```
var myregex = new Regex(@"^[\u0600-\u06FF\uFB8A\u067E\u0686\u06AF\u200C\u200F ]+$");
```

۳) الگوریتم polarity :

در این روش ابتدا باید دو کلمه که یک کلمه بار معنایی مثبت و دیگری بار معنایی منفی که مرتبط با موضوع پروژه است را انتخاب کنیم که لزوما می تواند تنها دو کلمه نباشد می توانیم مجموعه ای از این دو کلمه ای ها را انتخاب کنیم تا تحلیل ما به واقعیت نردیک تر باشد و از دقت کافی برخوردار باشد. انتخاب این دو کلمه برای من سخت بود و تنها از دو کلمه « خوب و بد» برای تست الگوریتم استفاده کردم. در این الگوریتم باید تعداد تکرار کلمه مثبت و منفی که در نظر گرفته ایم را در کل متن ورودی بدست آوریم.(در تابع (CalcNumberWord(string word سپس باید یک سری عبارت را از متن ورودی انتخاب کنیم که برای انتخاب این عبارت ها از part of speech tag استفاده کرده ایم. برای این پروژه عبارت های «موصوف و صفت» و «مضاف و مضاف الیه» را انتخاب کردیم یعنی POS tagآن ها باید به شکل زیر باشد: (در تابع (FindPhrase

N ADJ
N N

بعد از اینکه این عبارت ها را در کل متن بدست آوردیم باید تعداد دفعاتی که هر یک از این عبارت ها که در نزدیکی دو کلمه ی مثبت و منفی که در نظرگفته ایم را بدست آوریم.(در تابع (hitPhraseWithWord(List<string> phrases , string word سپس فرمول زیر را برای هر یک از عبارت ها محاسبه می کنیم که polarity آن عبارت را نشان خواهد داد.(در تابع (calcPhrasesPolarity(string posWord, string negWord

$$\text{Polarity}(\text{phrase}) = \log_2\left(\frac{\text{hits}(\textit{phrase}\ \text{NEAR "excellent"})\text{hits}(\text{"poor"})}{\text{hits}(\textit{phrase}\ \text{NEAR "poor"})\text{hits}(\text{"excellent"})}\right)$$

polarity کل متن برابر میانگین polarity عبارت ها است. که این مقدار در خروجی چاپ می شود.(در تابع (calcPolarityText(string posWord , string negWord

۴) ارزیابی:

میزان محبوبیت دکتر آذری جهرمی باتوجه به کامنت های پست های اینستاگرامی ایشان و با استفاده از دو کلمه ی **خوب و بد** قبل از فیلتر تلگرام برابر با ۱/۲۳- و برای بعد از فیلتر تلگرام برابر ۱/۴۴- می باشد یعنی می توان این طوری تفسیر کرد که قبل از فیلتر تلگرام مردم علاقه ی چندانی نسبت به ایشان نداشته اند و بعد از فیلتر تلگرام این علاقه کمتر شده است و نارضایتی بیشتر شده است. البته با این نتیجه

نمیتوان نتیجه ی دقیقی گرفت زیرا فقط از معیار **خوب و بد** استفاده کردیم(به دلیل نبود وقت واجرای طولانی برنامه فرصت نکردم برای کلمات دیگری برنامه را اجرا کنم)

نمودار زیر میزان قطبیت کامنت ها در تاریخ های مختلف با **کلمات خوب و بد** را مشاهده می کنید.