توضیحات پروژه متن کاوی در متون ادبیات فارسی
نویسنده : علی زاهدی سهروفروزانی
شماره دانشجویی : 9813016013
/.
ضمن عرض خسته نباشید خدمت شما استاد بزرگوار بنده برنامه ام رو تا جایی که مد نظرم بود پیش بردم و توضیحات کد های ارسالی در پیوست این فایل را خدمتتان ارائه می کنم.
دیتاستی که در اختیار بنده قرار دادید شامل چندین هزار توییت بود که هر توییت با توجه به محتوای آن نمره ای را از هریک از شش احساس اصلی anger, fear, happy, hate, sad, wonder دریافت می کرد.
هدف این هست که tf-idf تمامی کلمات را محاسبه کنم

1. بدین منظور در فایل main.py ابتدا دیتاست خام را وارد می کنم و به منظور انجام پیش پردازش و حذف کلمات توقف و دیگر کلمات فاقد ارزش سه فایل دیگر با نام های وارد می کنم
2. در قدم بعدی دیتاست را به تابع پیش پردازش ارسال می کنم و در این تابع اولین سطر دیتاست که حاوی نام احساسات است را جدا میکنم و پاکسازی جزئی ای روی دیتاست انجام میدم نظیر حذف کامل سطر های ناقص پس از آن تمامی کلماتی که در الفبای فارسی نیستند را بایپس میکنم و در ادامه متن هر سطر را به تابع دیگری به نام استاپ ورد ارسال می کنم اون هم عملیات بایپس کردن کلمات مشترک با فایل هایی هست که از قبل وارد برنامه کردم.
3. در ادامه فرمت متن ها امتیاز های هر توییت رو به نحوی تغییر میدم که بتونم روی اون کار کنم و اینکه لیست کلمات یونیک دیتاست رو پیدا کنم
4. در ادامه یکسری ثابت های برنامه را تعریف می کنم
tf-idf-limit : مشخص می کند که کلماتی که مقدار tf-idf انها از این مقدار کمتر است آن را نادیده بگیرد
emotion_limit : مشخص می کند که کلماتی که مقدار emotion انها از این مقدار کمتر است آن را نادیده بگیرد
5. در اخرین حلقه این برنامه اصلی ترین عملیات صورت می گیرد و به ازای تمام کلمات موجود در دیتاست مقدار tf-idf محاسبه می گردد و در فایلی به نام tf_idf_word.txt ذخیره می گردد.

**** با توجه به صحبت های شما باید مقادیر رو جوری تغییر میدادم که بهینه ترین حالت ممکن باشه ولی محدودیتی که باهاش مواجه شدم حجم محاسباتی بود که هربار برای محاسبه tf-idf کل کلمات باید انجام میشد و نمیتونستم لپ تاپمو زیر اون فشار بزارم به شدت داغ می کرد بخاطر مشکل سخت افزاری که این ترم واسش پیش اومده و به همون مقدار های اولیه بسنده کردم.**

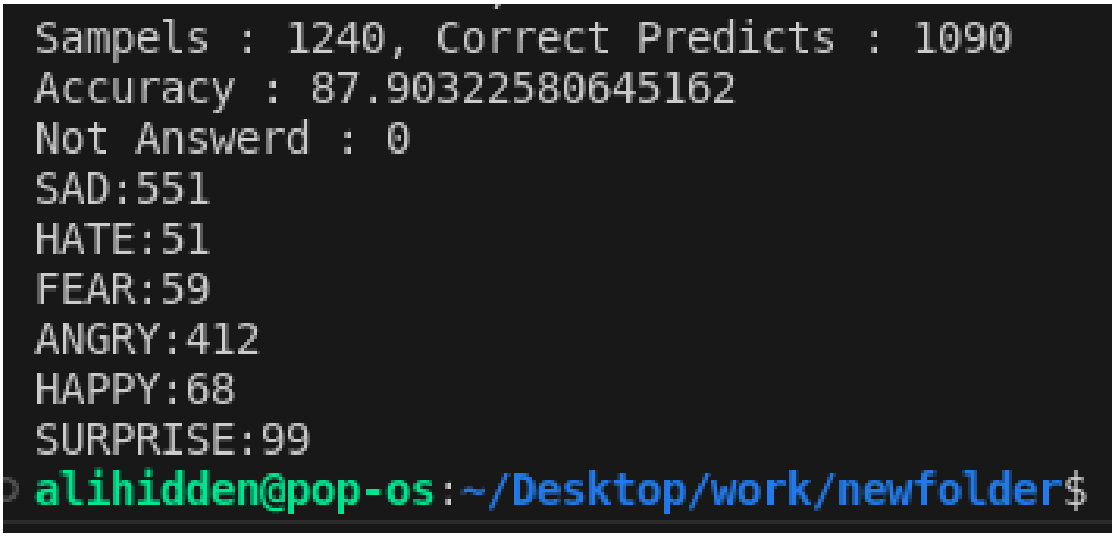
برنامه دومی که نوشتم با نام sorting.py هست که کارش اینه که دیتا هایی که با برنامه قبل تولید کردم رو به تفکیک هریک از احساسات به فایل اکسل می برد و در پژوهش های آینده می توان برای بهینه سازی برنامه ازش استفاده کرد و فعلا کاربردی ندارد.

برنامه سوم با نام readerlast.py عملیات گرفتن ورودی های تست را انجام می دهد و آن را با مدلی که ساختم تطبیق می دهد و مشخص می کند احساس هر توییت به چه شکل هست. اما اینکه چطور این کار را انجام می دهد:

1. این برنامه در دو حالت اجرا می شود:
    1. زمانی که خود این فایل را به تنهایی اجرا می کنید برنامه در یک حلقه بی نهایت از کاربر ورودی می گیرد و مشخص می کند متن وارد شده به کدام احساس تعلق دارد.
    2. اما زمانی که در کد دیگری آن را وارد می کنیم و تابع اصلی آن را صدا میزنیم تنها احساس پیش بینی شده را باز می گرداند.
2. زمانی که متن ورودی به هر شکلی به برنامه می رسد برنامه تک تک کلمات رشته ورودی را در فایلی که در برنامه اول تولید کردیم جستجو می کند و برای هر یک از احساس های شش گانه یک نمره می دهد ، به این شکل که نمره تمام کلمات را با توجه نمره آن کلمه در احساس مورد نظر جمع می کند و در انتها آن رشته را به احساسی تعلق می دهد که نمره آن بیشتر از همه باشد و خروجی را چاپ می کند.

برنامه چهارم با نام test.py در ابتدا برنامه readerlast را وارد می کند تا از تابع اصلی ان استفاده کند، در ادامه از دیتاست تستی که مربوط به پروژه ArmanEmo است برای تست برنامه استفاده می کنم. این دیتاست حاوی 1240 توییت ، نظر اینستاگرام و دیجیکالا است که با احساس آن متن کلاس بندی شده اند. هریک از متن های این دیتاست را به تابع main_app در readerlast میفرستم و این تابع از اطلاعاتی که قبل تر توضیح دادم تشخیص می دهد که احساس آن چیست، احساس تشخیص داده شده را با مقدار واقعی احساس آن تطبیق می دهم و در تنها درصد درستی مدل را اعلام کنم.

تصاویر اجرای برنامه در ادامه اورده شده است



چند مثال از برنامه readerlast

```
alihidden@pop-os:~/Desktop/work/newfolder$ python readerlast.py
Enter Text : دیشب پدرم مرد و من کلی گریه کردم
The emotion scores for the string 'دیشب پدرم مرد و من کلی گریه کردم' are:
ANGRY: 1.33735145066847
FEAR: 0
HAPPY: 0
HATE: 23.762812566710096
SAD: 69.78009025044562
SURPRISE: 0
The highest score is 69.78009025044562 and it belongs to the emotion(s): SAD
Therefore, the input string is related to the emotion(s): SAD
Enter Text : برق رفت
The emotion scores for the string 'برق رفت' are:
ANGRY: 0
FEAR: 0
HAPPY: 0
HATE: 0
SAD: 0
SURPRISE: 13.171010911548915
The highest score is 13.171010911548915 and it belongs to the emotion(s): SURPRISE
Therefore, the input string is related to the emotion(s): SURPRISE
Enter Text : من اتو متنفرم
The emotion scores for the string 'من اتو متنفرم' are:
ANGRY: 8.939625389952157
FEAR: 0
HAPPY: 0
HATE: 7.786397869221853
SAD: 0
SURPRISE: 0
The highest score is 8.939625389952157 and it belongs to the emotion(s): ANGRY
Therefore, the input string is related to the emotion(s): ANGRY
Enter Text : هوا عالیه
The emotion scores for the string 'هوا عالیه' are:
ANGRY: 0
FEAR: 0
HAPPY: 13.839951287571658
HATE: 0
SAD: 0
SURPRISE: 0
The highest score is 13.839951287571658 and it belongs to the emotion(s): HAPPY
Therefore, the input string is related to the emotion(s): HAPPY
Enter Text : کرونا برادرم رو ازم گرفت
The emotion scores for the string 'کرونا برادرم رو ازم گرفت' are:
ANGRY: 308.38047994962324
FEAR: 0
HAPPY: 0
HATE: 0
SAD: 10.061393346087453
SURPRISE: 4.66682218545583
The highest score is 308.38047994962324 and it belongs to the emotion(s): ANGRY
Therefore, the input string is related to the emotion(s): ANGRY
Enter Text :
```