به نام خدا

قسمت تحلیل عواطف و احساسات و هشتگ گذاری

پروژه نهایی درس تحلیل وسیستم‌های داده‌ای حجیم

استاد درس:

دکتر نعمت بخش

بهنام صوفی

۴۰۱۳۶۱۴۰۲۸

تیر ۱۴۰۲

لینک گیت‌هاب:

در مسیر زیر :

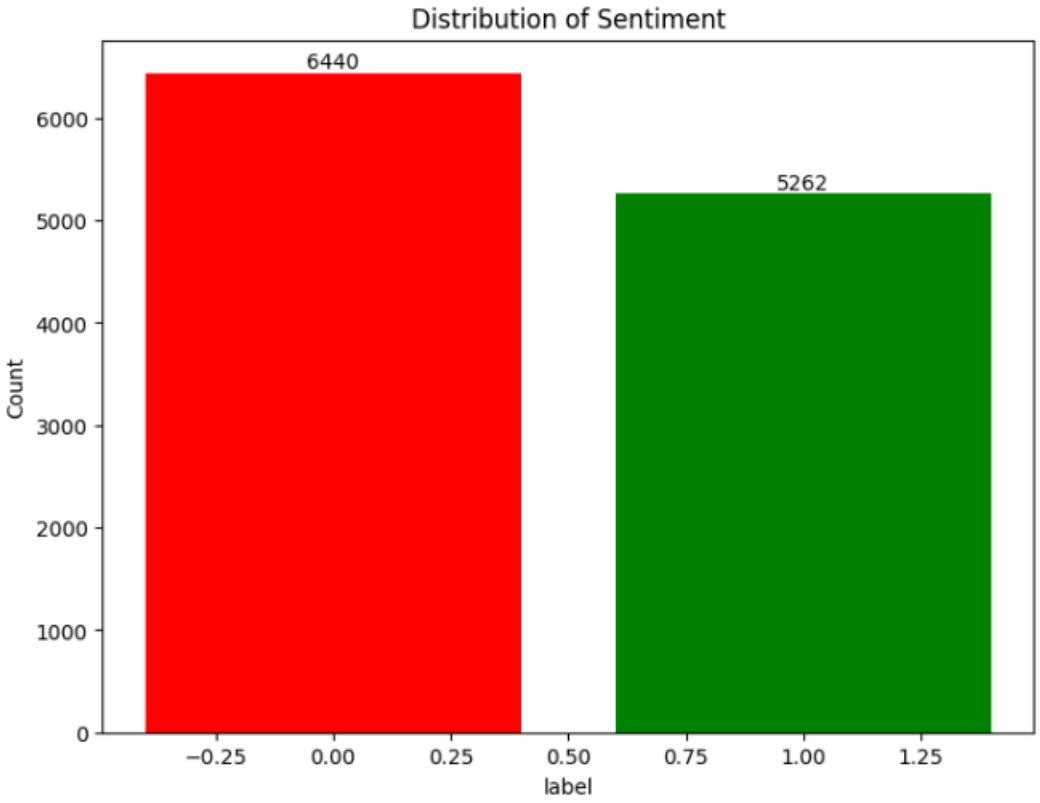
https://github.com/bscoder9595/BigDataProjects/

فایل bigdata_finalproject.rar

کار ایجاد مدل تحلیل احساسات و ایجاد برچسب در ۶ مرحله انجام شد، در ادامه مراحل را توضیح می‌دهیم:

۱. گرفتن داده‌ها

ما داده‌ها را از فایل csv می گیریم و در قالب دیتافریم اسپارک ذخیره می‌کنیم، در داده‌هایی که ما داریم ۱ به عنوان خبر مثبت و ۰ به معنی خبر منفی است. رنگ سبز نشان‌دهنده اخبار مثبت و رنگ قرمز نشان دهنده اخبار منفی است . در شکل زیر توزیع داده‌ها را مشاهده می‌کنید:



۲. انجام اعمال پیش‌پردازش

چهار عمل اصلی که در پیش‌پردازش انجام می‌شوند عبارتند از :

- توکن‌سازی

در این مرحله، متن ورودی (اطلاعات متنی مانند یک مقاله یا پاراگراف) به صورت جملات یا واحدهای کوچکتر تقسیم می‌شود، که به عنوان "توکن‌ها" شناخته می‌شوند. توکن‌ها ممکن است کلمات، نقاط، ویرگول‌ها، علائم نگارشی و ... باشند. این کار اولیه مرحله‌ای اساسی است که متن را به یک سری از توکن‌ها تبدیل می‌کند.

- حذف Stop Words

کلمات توقف (Stop Words) کلمات معمولاً رایج در زبان هستند و اغلب اطلاعات کمی را به ما می دهند. این کلمات شامل حروف اضافه، افعال متعلق به بودن، حروف ربط و ... می‌شوند. البته ما علائم نشانه گذاری را هم در این فایل قراردادیم. در این مرحله، توکن‌هایی که کلمات توقف هستند، از متن حذف می‌شوند. این کار به کاهش تعداد واژگان غیرضروری کمک می‌کند و می‌تواند در بهبود عملکرد مدل‌های یادگیری ماشینی کمک کند.

- تبدیل توکن‌ها به وکتورها

در این مرحله، توکن‌های پاک‌سازی شده به بردارهای عددی تبدیل می‌شوند. برای انجام این کار، از روش‌های مختلفی مانند CountVectorizer و TF-IDF استفاده می‌شود. در اینجا، از روش CountVectorizer استفاده شده است. این روش تعداد تکرارهای هر توکن را در متن مشخص می‌کند و این تعداد تکرار را به عنوان یک بردار عددی نمایش می‌دهد.

- محاسبه TF-IDF

TF-IDF یک روش محاسبه ویژگی‌های مهم در متن‌ها است که با استفاده از دو مفهوم "تعداد تکرار کلمه در سند (TF)" و "تعداد سندهایی که کلمه را شامل می‌شوند (IDF)" عمل می‌کند. TF-IDF برای اندازه‌گیری اهمیت یک کلمه در یک متن نسبت به مجموعه اسناد کلی استفاده می‌شود. با محاسبه این مقادیر، بردارهای ویژگی TF-IDF برای هر سند (متن) ایجاد می‌شود که برای استفاده در مدل‌های یادگیری ماشینی مناسب هستند. این ویژگی‌ها معمولاً به عنوان ورودی به مدل‌های یادگیری ماشینی تحویل می‌شوند تا بتوانند روی متن‌ها مسائلی مانند دسته‌بندی، خوشه‌بندی و ... را انجام دهند

در داده‌های موجود، ما فقط به ستون label، article و tiltle نیاز داریم . در قدم اول باید عمل توکن‌سازی را بر روی ستون articleرا انجام دهیم و کلمات موجود در آن متن را به صورت جدا بنویسیم، در قدم بعدی کلماتی که معروف به stop هستند و در یک فایل جدا آنها را آماده کردیم را ازمتن حذف کنیم. در مرحله بعدی کلمات جداشده را وکتور میکنیم و درآخر TF-IDF را برای آن‌ها محاسبه می‌کنیم.

۳. فراخوانی ۵ نوع الگوریتم و انتخاب یکی از آن‌ها

از ۴ الگوریتم Random Forest، Svm، Logistic Regression و NaiveBayes و شبکه cnn برای ایجاد مدل تحلیل احساسات استفاده شد که نتایج زیر را داشت و با توجه به اینکه cnn در بستر pyspark انجام نشده است، svm انتخاب شده است.

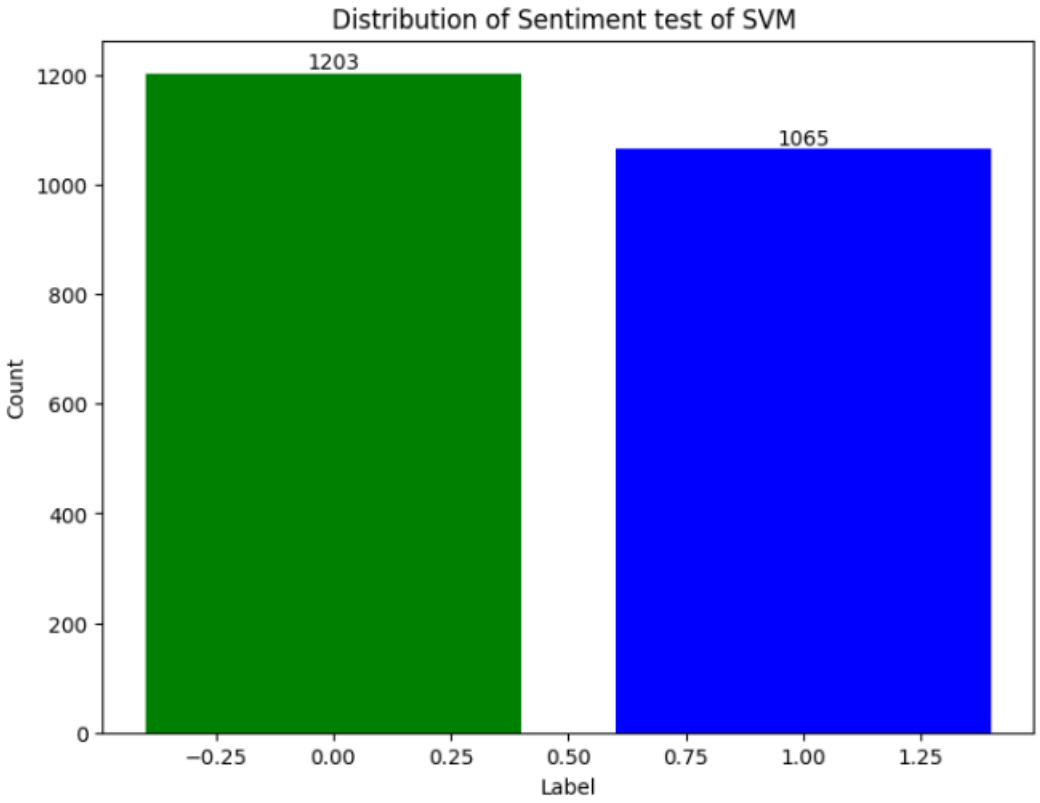
ایجاد مدل تحلیل احساسات

ما از همه الگوریتم‌ها استفاده کرده در جدول زیر دقت هر کدام را مشاهده می‌کنید:

الگوریتم‌ها	Accuracy
Svm	0.743
Logistic Regression	0.701
Random Forest	0.663
NavieBayes	0.717
Cnn	0.973

نمودار زیر توزیع پیش‌بینی svm روی داده‌های تست را نشان می‌دهد:

رنگ آبی نشان‌دهنده تعداد داده‌هایی که مثبت درنظرگرفته است و رنگ سبز تعداد نظراتی که منفی در نظر گرفته است.



۴. ذخیره مدل و لود دوباره آن برای تست

به دلیل اینکه وظایف تقسیم شده، مدل ایجاد شده را ذخیره کرده و دوباره لود کرده و تست می‌کنیم که مطمئن شویم مدل به درستی کار می‌کند. فراخوانی مدل ذخیره‌شده و تست دوباره مدل را در تصاویر زیر مشاهده می‌کنید:

```
newspd = test_news_df.toPandas()
newspd
```

	article
0	در اطلاعیه‌های منتشر شده امروز در سامانه کدال ...
1	در اطلاعیه‌های منتشر شده امروز در سامانه کدال ...
2	همچنین در اطلاعیه‌های امروز سامانه کدال شرکت س...

```
lr_loaded = LogisticRegressionModel.load('/content/drive/MyDrive/csvFiles/Logistic_model')

rf_loaded= RandomForestClassificationModel.load('/content/drive/MyDrive/csvFiles/random_model')

svm_loaded = LinearSVCModel.load('/content/drive/MyDrive/csvFiles/svm_model')
nv_loaded = NaiveBayesModel.load('/content/drive/MyDrive/csvFiles/NaiveBayes_model')
```

```
lr_loaded_predictions = lr_loaded.transform(test_news_df)
rf_loaded_predictions = rf_loaded.transform(test_news_df)
svm_loaded_predictions = svm_loaded.transform(test_news_df)
nv_loaded_predictions= nv_loaded.transform(test_news_df)
```

```
lr_loaded_sentiments = ["Positive" if label == 1 else "Negative" for label in lr_loaded_predictions.select("predictior
rf_loaded_sentiments = ["Positive" if label == 1 else "Negative" for label in rf_loaded_predictions.select("predictior
svm_loaded_sentiments = ["Positive" if label == 1 else "Negative" for label in svm_loaded_predictions.select("predicti
nv_loaded_sentiments = ["Positive" if label == 1 else "Negative" for label in nv_loaded_predictions.select("predictior

print("Logistic Regression Sentiments_loaded:", lr_loaded_sentiments)
print("Random Forest Sentiments_loaded:", rf_loaded_sentiments)
print("SVM Sentiments_loaded:", svm_loaded_sentiments)
print("NV Sentiments_loaded:", svm_loaded_sentiments)
```

```
Logistic Regression Sentiments_loaded: ['Negative', 'Negative', 'Negative']
Random Forest Sentiments_loaded: ['Negative', 'Negative', 'Negative']
SVM Sentiments_loaded: ['Negative', 'Negative', 'Negative']
NV Sentiments_loaded: ['Negative', 'Negative', 'Negative']
```

۵. ایجاد برچسب بر روی داده‌ها

ما به دو روش برچسب گذاری را انجام داده ایم روشی که اینجا در مورد آن توضیح خواهیم داد روش دوم است. تابعی داریم تحت عنوان extract_keywords که وظیفه دارد که از لیست کلمات دریافتی، کلمات کلیدی را استخراج کند. این تابع ابتدا لیست کلمات را به عنوان ورودی دریافت می‌کند، سپس با استفاده از لیست stopwords ، کلمات ناخواسته را حذف می‌کند. سپس تعداد تکرار هر کلمه را محاسبه کرده و کلمات را بر اساس تعداد تکرار به صورت نزولی مرتب

می‌کند. سپس لیست stopwords_list2 نیز به عنوان لیست دیگری از کلمات ناخواسته برای حذف اعمال می‌شود. در نهایت، تا سه کلمه کلیدی با تعداد تکرار بالاتر به عنوان نتیجه برگشت داده می‌شوند.

۶. ذخیره داده‌ها در قالب دیتافریم و فایل csv

داده‌ها در نهایت بصورت دیتافریمی از عنوان شرکت، خبر، پیش‌بینی این‌که مثبت یا منفی است و دو مجموعه هشتگ ذخیره می‌شود. دیتافریم زیر نشان‌دهنده دیتافریم خروجی است:

	title	article	prediction	hashtag1	hashtag2
0	«افزایش نرخ های فروش «خربخت	...به گزارش کدال نگر بورس24 ، شرکت صنایع ریخته گ	0.0	میباشد, نرخ, مذبور, مبلغ, میللنگ	نرخ, باشد., مذبور
1	روند فروش "لخزر" را اینجا ببینید	...به گزارش کدال نگر بورس24 ، شرکت صنعتی پارس خ	1.0	ماهم, فروش, مبلغ, بورس	ماهم, فروش ,
2	"بررسی روند فروش "غشاذر	...به گزارش کدال نگر بورس24 ، شرکت شیر پاستوریزه	1.0	ماهم, فروش, مبلغ, بورس	ماهم, فروش ,
3	بلوک 7 درصد «گنگین» عرضه می شود	...به گزارش کدال نگر بورس24 ، شرکت کارگزاری توسعه	0.0	عرضه, بورس, کارگزاری, توسعه, سهند	عرضه, کارگزاری, توسعه
4	تحقق سود 6 میلیارد تومانی «سنوین» در خرداد ماه	...به گزارش کدال نگر بورس24 ، شرکت سرمایه گذاری ا	1.0	خرداد, ارزش, بورس	.خرداد, کرد ,
...	...	...	...	...	...
2264	بررسی عملکرد «نکنار» تا پایان شهریور	...به گزارش کدال نگر بورس24 ، شرکت مجتمع معادن مس	1.0	سود, بورس, مجتمع, معادن, نکنار	سود, مجتمع ,
2265	«تحقق سود 18 میلیارد تومانی در «شرنگی	...به گزارش کدال نگر بورس24 ، شرکت صنعتی و شیمیای	1.0	سود, بورس, صنعتی, شیمیایی, رنگین	سود, صنعتی, شیمیایی
2266	آیا از تغییرات پرتفوی «ومعادن» خبر دارید؟	...به گزارش کدال نگر بورس24 ، شرکت سرمایه گذاری ت	0.0	شهریور, ارزش, بورس, سرمایه, توسعه	شهریور, ارزش, سرمایه
2267	علت رشد سود «سغرب» چه بود؟	...به گزارش کدال نگر بورس24 ، شرکت صنایع سیمان غ	1.0	بورس, صنایع, سیمان, غرب, سود	شرکت , صنایع, سیمان
2268	تحقق سود 19 میلیارد تومانی «ومهان» در مهر ماه	...به گزارش کدال نگر بورس24 ، شرکت گروه توسعه مال	1.0	مهر, بورس, گروه, توسعه, مالی	مهر, گروه, توسعه

2269 rows × 5 columns

در نهایت دیتافریم را در یک فایل csv برای استفاده در مرحله بعدی، ذخیره می‌کنیم.