2019 Fall ISGB/BYGB 7978

Homework 3 Building a review classifier

Name: Jiayin Hu

Before the report:

1. *Two Jupyter Notebooks are submitted with this report. One contains data collection. The other is mainly about exploratory data analysis and building classifier.*
2. *For better understanding dataset and consistency, EDA is conducted before building classifier in code part.*
3. *Some figures in the report are generated by Tableau. There is no code for them in Jupyter Notebooks.*

In this homework, I completed collecting reviews and ratings from film review Rotten Tomatoes, conducting basic exploratory data analysis and build classifier to predict rating based on review text.

1. Data Collection

In this part, I get all critics' reviews of the weekly top 15 box office movies (Nov. 8 - Nov. 10) from Rotten Tomatoes.

a. I collect the movies' name and links from the weekend box office ranking chart.

```
In [748]: movie_dict

Out[748]: {'Midway': '/m/midway_2019/',
           'Doctor Sleep': '/m/doctor_sleep/',
           'Playing with Fire': '/m/playing_with_fire_2019/',
           'Last Christmas': '/m/last_christmas_2019/',
           'Terminator: Dark Fate': '/m/terminator_dark_fate/',
           'Joker': '/m/joker_2019/',
           'Maleficent: Mistress of Evil': '/m/maleficent_mistress_of_evil/',
           'Harriet': '/m/harriet/',
           'Zombieland: Double Tap': '/m/zombieland_double_tap/',
           'The Addams Family': '/m/the_addams_family_2019/',
           'Jojo Rabbit': '/m/jojo_rabbit/',
           'Countdown': '/m/countdown_2019/',
           'Parasite (Gisaengchung)': '/m/parasite_2019/',
           'Motherless Brooklyn': '/m/motherless_brooklyn/',
           'Black and Blue': '/m/black_and_blue_2019/'}
```

b. After getting the links of these 15 movies, I parse the html of the review pages to retrieve reviews. The process is basically similar as the last homework, except for I get the total number of review pages to make sure retrieve all reviews. (Reference of last homework: github.com/hujiayin/WebAnalytics/blob/master/Web%20Content/crawling_movie_reviews.ipynb)

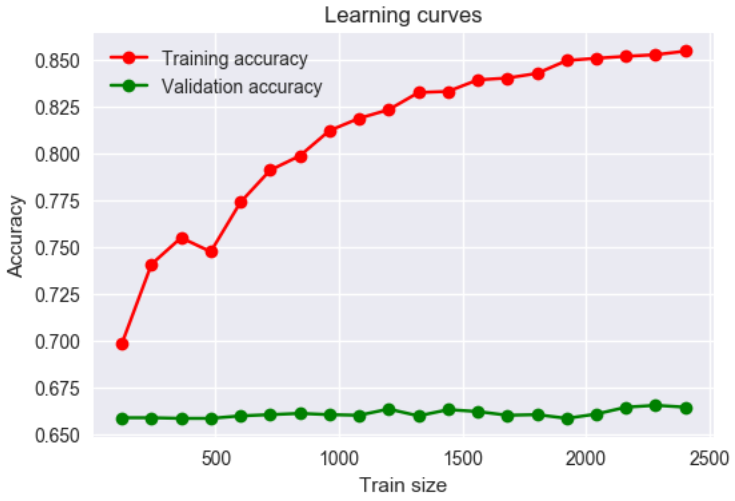
I obtain 3006 reviews of all these 15 movies. The head of the DataFrame shows as follow.

	Critic	Date	Movie	Rating	Review	Review_text	Review_text_lemma	Source
0	Sara Michelle Fetters	November 17, 2019	Midway	rotten	There wasn't any tension or suspense, and by t...	there wasnt any tension or suspense and by the...	wasnt tension suspense time thing came end alm...	MovieFreak.com
1	Matt Brunson	November 16, 2019	Midway	rotten	Director Roland Emmerich's 1996 smash Independ...	director roland emmerichs 1996 smash independe...	director roland emmerichs 1996 smash independe...	Film Frenzy
2	Jackie K. Cooper	November 15, 2019	Midway	fresh	Good historical recounting, and solid special ...	good historical recounting and solid special e...	good historical recounting solid special effec...	jackiekcooper.com
3	Abigail Camarillo	November 15, 2019	Midway	rotten	The director of The Day After Tomorrow does no...	the director of the day after tomorrow does no...	director day tomorrow much offer full review s...	Chilango.com
4	Mark Kermode	November 15, 2019	Midway	rotten	He's Mr. Bombastic, but he's not very fantasti...	hes mr bombastic but hes not very fantastic	he mr bombastic he fantastic	Kermode & Mayo's Film Review

2. Building Classifier and Evaluation

I first use different text feature extraction tools to get the words in reviews. To find a relatively better feature extraction tools, I use logistic regression to fit a model and conduct cross validation in our dataset.

Different Text Extraction Tools with Logistic Regression				
	CountVectorizer	CountVectorizer (Binary)	TfidfVectorizer	TfidfVectorizer (Bigram)
Mean Accuracy	63.1732%	62.9735%	**66.3672%**	65.8351%



Learning Curve for TfidfVectorizer (1-gram)

The problem here is we did not make the training accuracy and testing accuracy converge because we did not have sufficient samples here. Anyway, I will use these data to find better classifiers.

Then, I tried three machine learning methods to build a model and tuning some of the parameters to get higher accuracy.

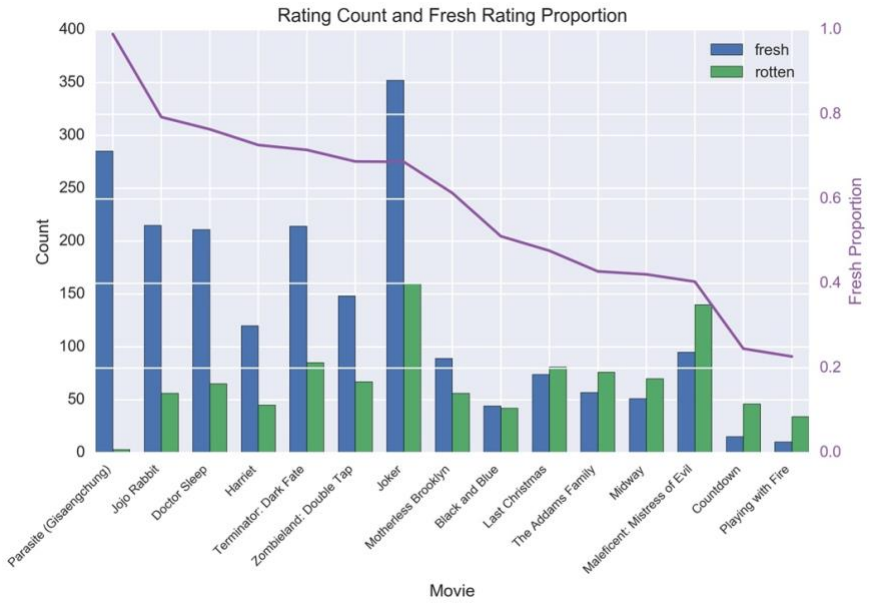
n-gram (TfidfVectorizer)	Classifier	Main Parameters	Accuracy
1-gram	Logistic Regression	C=1.0, solver=liblinear, penalty=l2	**66.3672%**
	Support Vector Machine	C=1.0, degree=3, gamma=scale, kernel=poly	65.8682%
	Naïve Bayes	alpha=0.5, fit_prior=True	65.7685%
2-gram	Support Vector Machine	C=1.0, degree=3 gamma=scale, kernel=poly	65.8682%
	Naïve Bayes	alpha=1, fit_prior=True	65.6354%

Therefore, I finally get the classifier with 66.3672% accuracy.

3. Exploratory Data Analysis and Text Analysis

Among the 3006 records, 65.87% ratings are “fresh” and 34.13% are “rotten”. The dataset is not balanced, this may cause some challenges to the classification problem. In our data, the movies with higher fresh rating proportion often have more reviews towards them.

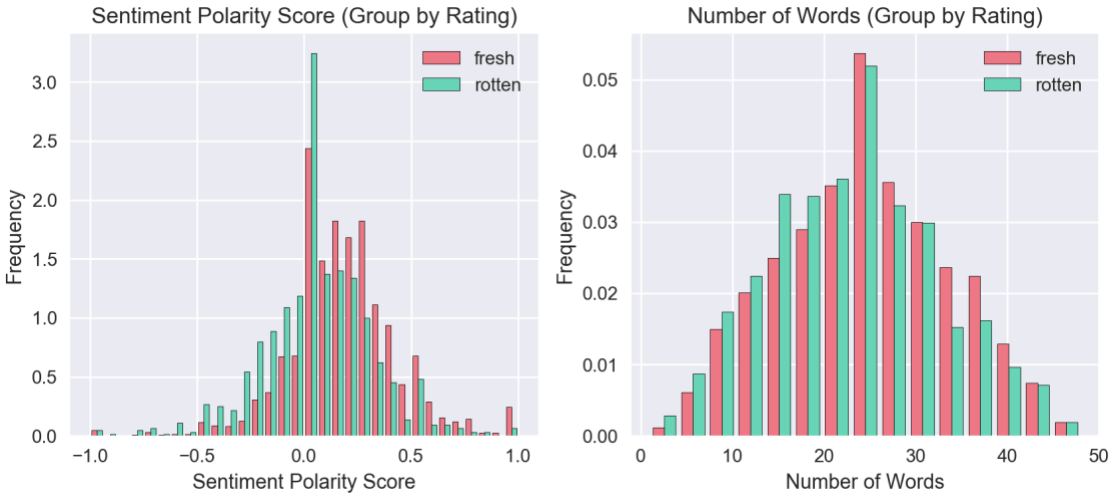
	Count	Proportion
Fresh	1980	0.658683
Rotten	1026	0.341317



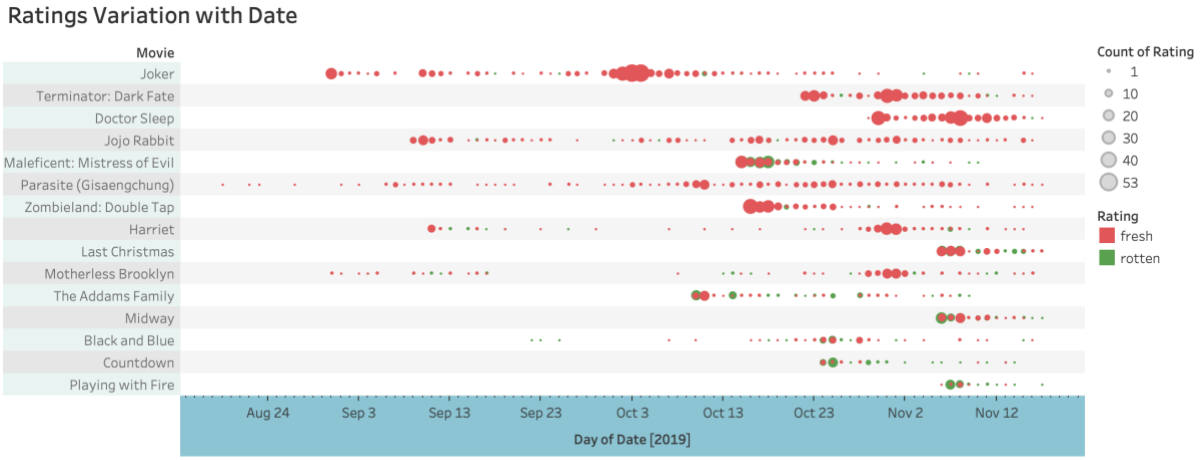
To conduct more basic exploratory data analysis, I remove the punctuations in the review text, count the words in each review and transform date to the standard date format. I import a sentiment analysis library TextBlob to calculate the polarity of the reviews. The polarity is a number between -1 to 1. A positive/negative number means the content of the text is positive/negative. Zero represents no significant sentiment in the text.

68% reviews get positive polarity and only 9% have negative score based on TextBlob. Based on the distribution, fresh reviews are more likely to get a positive score than rotten reviews.

The distribution of number of words is basically symmetrical between 0-50. Most of reviews have around 20-30 words. Therefore, critics always write short comments on Rotten Tomatoes.

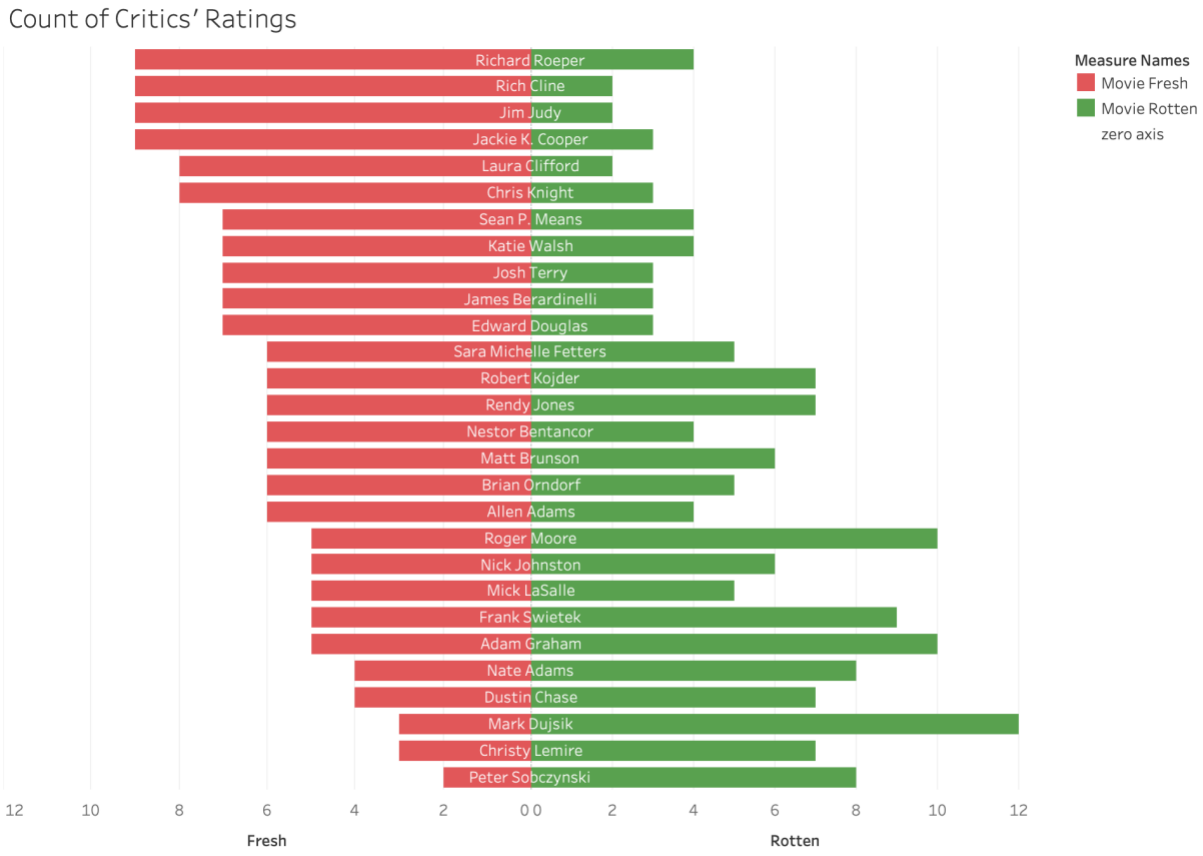


I truncate date range from August 18 to November 17 to see the number of reviews (only removing part of reviews of Parasite). Several movies got a bunch of reviews at the beginning of release. After a period, other critics wrote more reviews intensively in several days.



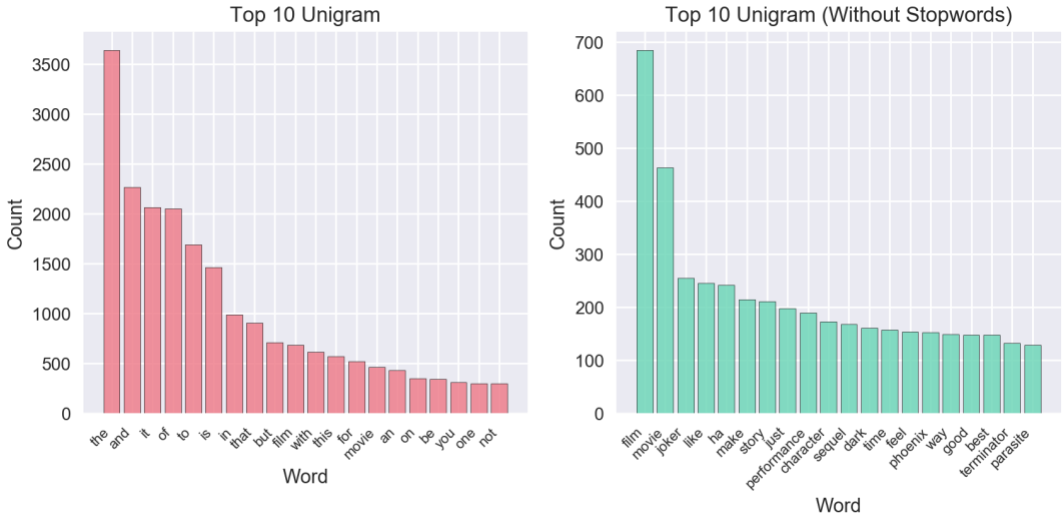
Date Day for each Movie. Color shows details about Rating. Size shows count of Rating. The view is filtered on Date Day, which ranges from August 18, 2019 to November 17, 2019.

Different critics have obviously diverse taste to the movies. Some critics tend to have higher ratings to the top box office movies, while some do not.

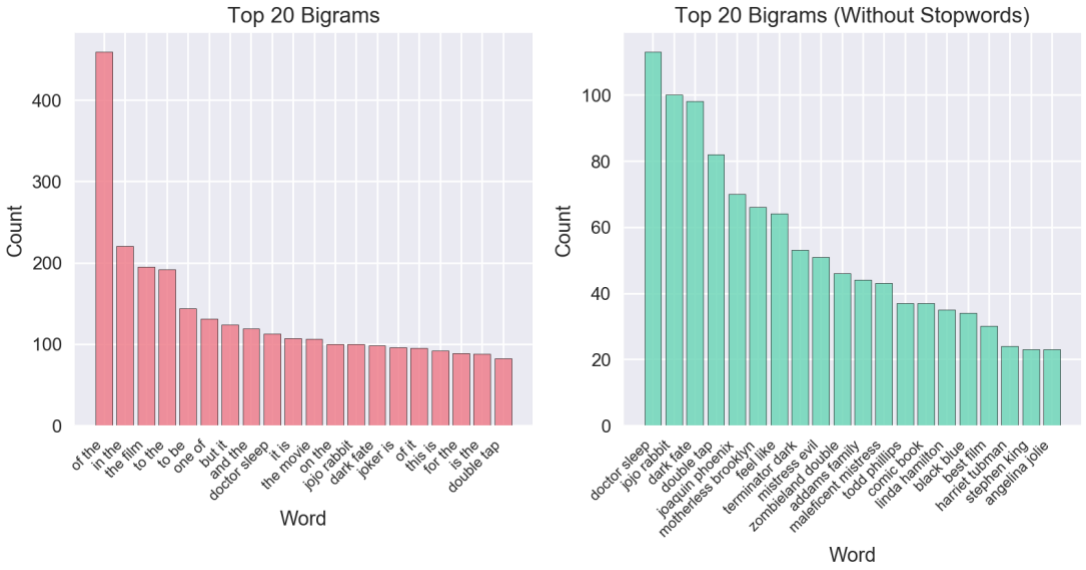


Movie Fresh, zero axis , Movie Rotten and zero axis for each Critic. For pane Sum of Movie Fresh: Color shows details about Movie Fresh, zero axis and Movie Rotten. For pane Sum of Movie Rotten: Color shows details about Movie Fresh, zero axis and Movie Rotten. For pane Sum of zero axis : Color shows details about Movie Fresh, zero axis and Movie Rotten. The data is filtered on sum of Movie, which ranges from 10 to 15.

When see the top unigram and bigram chart, if we do not remove stop words, most of the frequent words/bigrams are stop words.



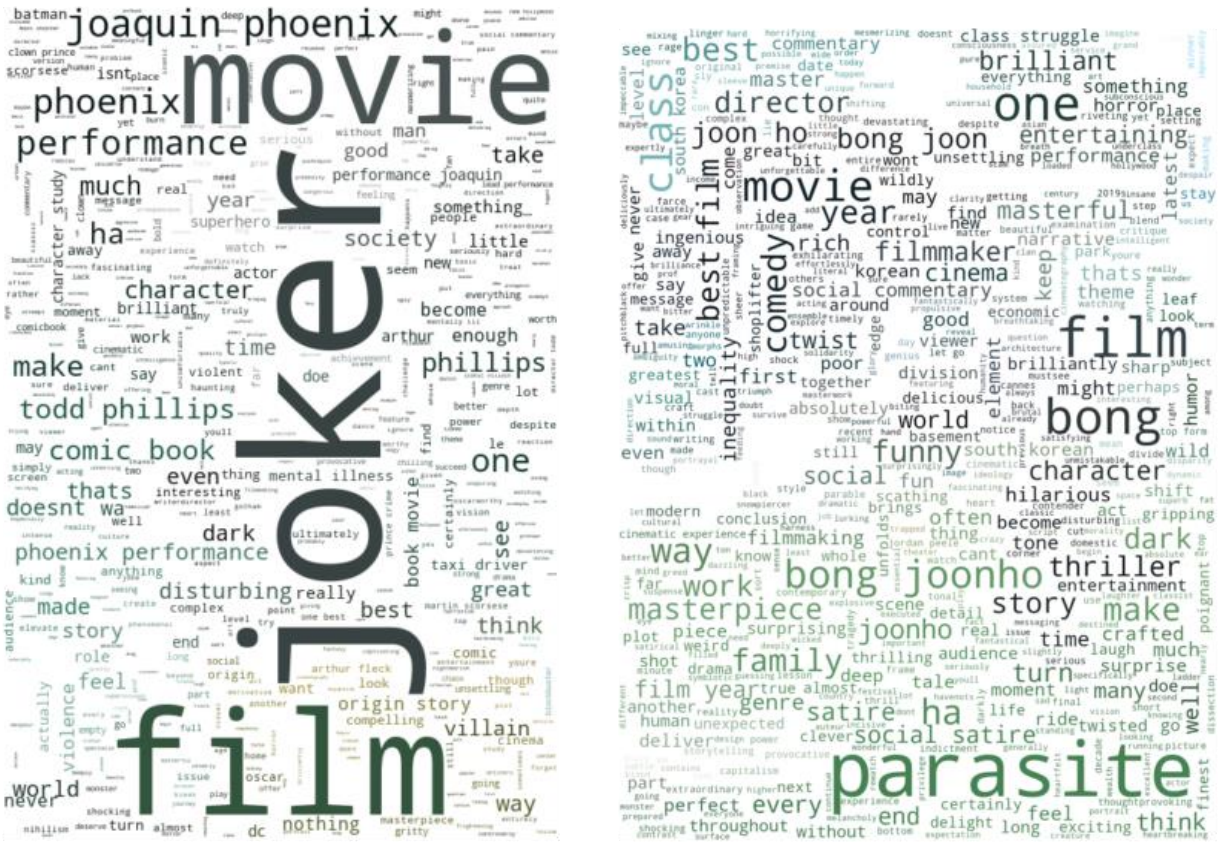
After removing stop words, we can observe the reviews are collected from a movie review website. “Film” and “movie” are most frequently used. Also, some words are related to film, such as “story”, “performance”, “sequel”. And names of movies, elements in movies and names of actors and director are mentioned.



I also generate several figures of word cloud to show the frequent mentioned words.

Word Cloud for All Reviews

Directors and actors are crucial parts of a movie. Joaquin Phoenix, the star actor of Joker, Todd Phillips, the director of Joker, and Bong Joon-ho, the director of Parasite are all shown in the reviews. Both of the movies gain a lot of fresh ratings, while Joker also has relative more rotten ratings. The word cloud of Joker shows several negative words, such as doesn’t and isn’t. The word cloud of Parasite is full of compliment, like masterpiece, surprising, brilliant, crafted.



Word Cloud for Joker (left) and Parasite (right)