BIA660 Midterm Report

Jichen Jiang, Zijun Fang

March 2023

1 Introduction

As a major movie review website, IMDb provides not only the rating of the movie but also many different opinions. However, some of these opinions may not be seen by moviegoers due to the large number of reviews. Therefore, a review recommendation system can be a solution to the inconvenience caused by the long time it takes to read through reviews.

To build such a system, the most important thing is to determine which type of reviews is most useful up to the current time, and upvote percentage should be the primary consideration for existing reviews. After identifying the useful reviews, sentiment analysis can be used to weigh different text in the review and provide a numerical way to evaluate different text with regard to emotion and grammar concerns. This weighted process will help determine how "useful" a new review posted in the future is.

Furthermore, analyzing the relationship between reviews and movie age is also important to determine how useful a review is. Movie taste largely change different time periods, which will change the reviews accordingly.
(Midterm report version, we will elaborate our goal more closely during our further programming and discovering)

2 Literature Review

Sentiment analysis is a powerful tool to categorize words with emotional polarity. Given a piece of written text, the problem is to categorize the text into one sentiment polarity: positive, negative, or neutral. Based on the scope of the text, there are three levels of sentiment polarity categorization, namely the document level, the sentence level, and the entity and aspect level [1]. In our project, the major research gap is adding age feature and genre feature into the sentiment analysis with a more accuracy score for whole review text. By adding these features, the environment of analysis will be changed, since emotion toward different movie genre and age could be entirely opposite. Prakash and Chakravarthy[2] provide a statistical way to weighted reviews, their method

using tf-idf and word-weight scheme. Their weighting process will be useful for our model building inside of each genre and age, but we still need to contain certain value for genre and age to give an precise weighted accordingly.

(Above is Midterm report version, we will keep adding more article review in this part as well as our reference)

3 Research Question

The main research question in this project is how to determine the weights of different text with emotion and genre considerations in movie reviews. Determining the weighted value of text will help build a review recommendation system that allows people to follow the most useful reviews before they go to see a movie. Moreover, determining the weights of different text with emotion and genre considerations can be applied in other fields, such as shopping and restaurants, where reviews are prevalent.

4 Methodology

In this section, we are going to talk about the methods we used in this project, including how we collect the data, preprocessing and some basic NLP analysis.

For all of our code, including the scraping script and preprocessing scripts, and the analysis scripts, please refer to the files we uploaded, or our GitHub project page: https://github.com/aturret/IMDB-Data-Mining

4.1 Data Collection

The dataset was obtained from IMDb.com. This study uses data from the famous IMDb Top 250 Movies. We get the metadata and all user reviews of these movies. Table 1 shows the information on the metadata that we scraped. Table 2 shows the information on the reviews that we scraped.

The number of all reviews is 340195. We believe that such a huge dataset is enough to get meaningful conclusions that align with our research goal. All of the data are scraped by Mar 16, 2023.

4.1.1 Scraping

We use Beautifulsoup4 and Selenium to scrape these data, and saved all the data in the format of CSV, so that we can store it locally and then import the

data into Pandas data frame easily.

Columm	Meaning	Data Type	Example
movie_id	The IMDB id of the movie	String	tt0468569
movie_title	The title of the movie	String	The Dark Knight
movie_rating	The average rating of the movie	Float	9.0
movie_rating_number	The number of ratings of the movie	Int	2687759
movie_genres	The genres of the movie	List	['Action', 'Crime', 'Drama']
movie_year	The year of the movie	Int	2008
movie_content_rating	The content rating of the movie	String	PG-13
movie_duration	The duration of the movie	String	2h 32m
review_number	The number of reviews of the movie	Int	8430

Table 1: Raw Data Information of Movie metadata

4.1.2 Format Standardization

To make data analysis easier, we convert the movie duration column to int format by minutes and convert the review date column to standard Pandas dataframe datetime64 format. Table 3 shows the columns that we standardized.

4.2 Preprocessing

In this section, we discuss the preprocessing of the collected data. We split documents into tokens and clean them up. Table 4 provides examples of the text before and after tokenization, stopword removal, and normalization.

4.2.1 Text Cleaning and Tokenization

This process involves removing symbols and characters from sentences, breaking sentences into individual words, and converting uppercase letters to lowercase. We perform the tokenization process on both the review_title and review_text.

4.2.2 Removing Stop Words

Stop words are commonly used words with little meaning, such as "and" and "the." We filter the tokenized text using the stopword list provided by nltk.corpus.

Columm	Meaning	Data Type	Example
movie_id	The IMDB id of the movie	String	tt0050083
movie_title	The title of the movie	String	12 Angry Men
review_id	The average rating of the movie	String	rw3666418
review_author	The username of the reviewer	String	mark.waltz
review_title	The title of the review	String	One of the great theatrical examples of what makes for superb drama.
review_date	The date of the review	String	20 March 2017
review_rating	The rating of the review for the movie. If the user didn't give a rating, leave it as None	Int	10
review_text	The duration of the movie	String	Theater at its best is practically impossible to get down on film correctly. When Hollywood gets it right, they create a work of art. In this case, ...
review_helpfulness_upvote	The number of upvotes for "if this review is helpful"	Int	42
review_helpfulness_total	The total number of votes for "if this review is helpful"	Int	46

Table 2: Raw Data Information of Movie review

Columm	Meaning	Data Type	Example
movie_duration	The duration of the movie	Int	152
review_date	The date of the review	Datetime64	2017-03-20

Table 3: The Columns that get preprocessed

review_id	original text	tokenized text
rw0083653	If you like ransom/police stories-mysteries, and have interest in Kurosawa &or Mifune, check it out at least once	['like', 'ransom', 'polic', 'stories-mysteri', 'interest', 'kurosawa', 'mifun', 'check', 'least']
rw0083651	Kurosawa at his best and most subtle	['kurosawa', 'best', 'subtl']

Table 4: Examples of Tokenization

4.2.3 Normalization

We transform words with different surface forms in the tokenized text into a more uniform representation. For movie reviews, we believe that the Snowball Stemmer is the most effective method for normalizing words.

4.3 Data Description

In this section, we did some exploratory data analysis for the data we collected and preprocessed, including correlation analysis and feature extraction.

4.3.1 Correlation Analysis

We did a correlation analysis for the numeric data of the IMDb top 250 movies metadata. Table 5 shows the result. The correlation analysis table shows the relationships between four variables: movie_rating, movie_rating_number, movie_year, and review_number.

A strong positive correlation is observed between movie_rating_number and review_number (0.727), suggesting that movies with more ratings also tend to have more reviews. Movie_rating also has a moderate positive correlation with movie_rating_number (0.602) and review_number (0.524), indicating that movies with higher ratings generally have more ratings and reviews.

For the relationship between movie_year and other columns, movie_year has a weaker positive correlation with movie_rating_number (0.427) and review_number (0.359), implying that more recent movies may have slightly more ratings and reviews. The correlation between movie_year and movie_rating is negligible (0.029), indicating no meaningful relationship between the movie's release year and its rating.

4.3.2 Feature Extraction

This process generates features from the movie review documents using the Term Frequency-Inverse Document Frequency (TF-IDF) technique. TF-IDF is

	movie_rating	movie_rating_number	movie_year	review_number
movie_rating	1.000000	0.602066	0.029436	0.524194
movie_rating_number	0.602066	1.000000	0.427648	0.727157
movie_year	0.029436	0.427648	1.000000	0.359601
review_number	0.524194	0.727157	0.359601	1.000000

Table 5: Correlation Analysis for Numeric Data of Metadata

a method for calculating weights after the feature extraction process, effectively capturing the importance of words within individual documents and across the entire document collection.

$$tfidf(w,d) = \frac{s(w,d)*}{\sqrt{\Sigma_{w \in d} s(w,d)^2}}$$

According to the TF-IDF technique, we can get document similarity from the documents.

5 Mid-term Conclusion

According to the data analysis, we find that the dataset satisfies our research questions. The correlation analysis indicates that we should research more about the contents of the reviews, and the document similarity can help us with it. We will dig more into the data and continue our research with the classification methods we will learn later.

References

[1] Fang, X., Zhan, J. *Sentiment analysis using product review* data.Journal of Big Data 2. https://doi.org/10.1186/s40537-015-0015-2

[2] S. Prakash, T. Chakravarthy, E. Kaveri, *Statistically weighted reviews to enhance sentiment classification*, Karbala International Journal of Modern Science, Volume 1, Issue 1, 2015,

[3] Ramadhan, N., Ramadhan,T. (2022). *Analysis Sentiment Based on IMDB Aspects from Movie Reviews using SVM.* https://doi.org/10.33395/sinkron.v7i1.11204