Predicting IMDb Rating From Movie Scripts

Siyu Liang, Ivy Wang, and Jingjing Lin

Georgetown University

Abstract

Movie scripts contain a wealth of information that can be used for building prediction models. In this work, we build a Linear Regression model and Random Forest Regression model, based on movie scripts to predict film IMDb ratings. Our models combine a variety of linguistic features based on screenplay texts, such as TF-IDF, POS tags, etc. The dataset we use comes from a publicly accessible source, consisting of 824 films in various genres. The performance of the model demonstrates that there exists a weak correlation between those features extracted from the movie script and its IMDb rating, and suggests that the possibility that information beyond movie scripts are also crucial for building a prediction model for the public ratings.

1 Introduction

As the entertainment industry continues to produce numerous shows and films every year, it is increasingly important for creators, directors, and business investors in the industry to be able to pinpoint the key factors that contribute to a films success, which is measured in various ways, such as monetary return from box office as well as the societal reception of the themes, characters, and concepts from the film. To the latter aspect, online ratings from public platforms (such as Rotten Tomatoes and IMDb.com) provide a good indication of the public reception of the production with respect to socially-salient topics, public aesthetics, trend, and even economic impacts (Peck 2015). With the ratings, production teams, investors, and third parties obtain a concrete metric and even ultimately financial insights that can measure how well the product did in practice (Prag et al. 1994; Peck 2015).

In this paper, we are interested in examining the screenplay in films, and whether this textual information has any weight in determining the public ratings of the film. Screenplays are often seen as the raw material from the playwrights before any directing, acting, and special effects are added. Furthermore, dialogues in films often convey idiosyncratic characteristics that stage directions or set descriptions simply cannot portray. Dialogues have also been shown to represent the character complexity, plot development, as well as stylistic features of a film (Eliashberg et al. 2007; Martinez et al. 2019). Thus, to NLP researchers who are concerned with film ratings, screenplays contain rich textual resources that can have statistical correlation through feature engineering and modeling. Then, if feature selection and modeling are both successful, the task can potentially help movie-makers and playwrights identify favorable language usages and linguistic choices that can potentially increase the public evaluation of the film in the end (Eliashberg et al. 2007).

2 Objective

In this paper, we aim to examine IMDb ratings based on linguistic data from the scripts of films. We perform NLP-related correlation metrics on two machine learning models trained, in order to investigate the relationship between the two. We hypothesize that, while an accurate prediction is nearly impossible, one can still find and identify higher correlations from certain features than others.

3 Dataset

We used two separate datasets, one of the screenplay texts and the other of user rating, among other movie metadata. Our screenplay dataset is NLDS Film Corpus 2.0[1], a publicly accessible corpus of

[1] Dataset available at: https://nlds.soe.ucsc.edu/fc2

Total # of movies	1245
Total # of sentences	1288190
Total # genres	22
Median # sent per script	33

Table 1: Discription of the Movie Script Dataset

Rating	<2	2-4	4-6	6-8	>8
No.	1	17	150	630	92
%	0.00112	0.0191	0.169	0.708	0.103

Table 2: Distribution of the Rating

1248 individual films screenplays across 22 different genres (Hu et al. 2015). The dataset comes in the form of text files, identified by the movie titles (see Table 1). On the other hand, we used rating data from IMDbs publicly accessible API files. The metadata include, for every film, a unique IMDb identification number, genre, release year, title, rating, and the number of votes received (see Table 2). Of importance to our study are the unique IMDb ID, movie title (used to link the ID and the rating), and movie rating. Later on we manipulate the datasets, and incorporate these information to correspond the screenplay files (as strings) to map onto their ratings from two originally distinct and separate data files from two different sources.

4 Background

The technique used for textual analysis and prediction has evolved a long way from pure statistical methods, such as linear regression, logistic regression, towards advanced machine learning models, such as neural networks, which are regarded as the principal method of many NLP research.

Previously regarding this topic of research, there exists a number of machine learning research that uses the script or other movie attributes as data, to predict (numerically or categorically) the box office or rating levels. Eliashberg et al. (2007) used movie scripts in addition to the production budget to simulate box office performance at the time of green-lighting. This prediction model is based on three levels of textual features extracted from scripts: genre and content, semantics, and bag-of-words. They experimented with various machine learning models, including linear regression, support vector machine, neural networks, while adopting some sentiment analysis methods. In Lee et al. (2018), it is the movie attribute features (genre, number of plays on the release day, language and region, etc.) rather than the texts of scripts that are used to predict box office performance, through a prediction model called Cinema Ensemble Model (CEM), which is a complex model built on adaptive tree boosting and logistic regression for the comparison of performance in prediction models. More recently, in Martinez et al. (2019), the authors used screenplay scripts to both predict and classify violence ratings through NLP and machine learning techniques. Specifically, they used a screenplay dataset that contains 945 Hollywood movies from 12 different genres, and they engineered linguistic features including N-grams, lexical features, sentiment features, distributed semantics, and abusive language features. They first trained a Linear Support Vector Classifier (LinearSVC) and then investigated context using an RNN model.

Summarizing previous work, we notice that from an NLP engineering perspective, many common features and techniques such as TF-IDF, bag-of-words, lexical semantics, etc. are frequently used. Since we are similarly dealing with a large body of characteristic text data, we place our priority on such linguistic features that can be extracted from text data (see Methodology section below, though we opted for less complex and ambiguous features, due to time and labor constraints). From a machine learning perspective, we notice that regression models would be preferred in our case since we are not concerned with classification, but a numerical prediction, where the correlation rather than accuracy is more informative in terms of model performance. These observations are reflected below in our selection of features, model types, and metrics.

5 Methodology and Data organization

Since the dataset is without original annotations, we plan to engineer the features from the pure text. We cleaned and preprocessed our data, then organized them to data frames that are compatible with the models we plan to use.

Because the screenplay and movie ratings are from separate sources, a crucial part is to match the screenplay text in one dataset to its rating in another dataset. The movie dialogue dataset is a folder with a number of subfolders of movie genres, under which movie dialogue text files are organized alphabetically. We created a tab-separated file of movie titles matching with dialogues. We then conducted appropriate preprocessing on movie titles to match the naming conventions in the IMDb dataset, from which we will get the corresponding unique IMDb ID. After matching the movie titles to the IMDb IDs, because the titles are matched with screenplays and the IDs are matched with ratings, we were able to match screenplays onto ratings. Our final product is a text (tab-separated) file of movie titles with their corresponding rating and dialogue texts.

We were able to match the ratings for 890 movies, discarding duplicates across genres. We averaged the ratings for the movies that have multiple versions with the same title. Later in feature engineering, we also discarded screenplays whose format presents a problem for feature extracting, namely those without consistent use of punctuations, thus resulting in an abnormally large mean length of sentence. We filtered out all the scripts whose mean lengths of utterances are over 50, a threshold we presume to be the maximum in normal speech. By the end of our data organization, we have 824 films in total that have the appropriate metadata matched. The median length of sentences is 33, with a lower quartile at 19 and a higher quartile at 55.

5.1 Linguistic Features

The first feature is the number of sentences per screenplay. To get this feature our function utilizes sent_tokenize() on each script, and we format the number of the sentence into an array. We shaped these features into a column that will be later composed into a large numpy matrix for the model. This feature alone will also ultimately be our baseline.

Our second feature is the average number of words per sentence in each screenplay. To get this, we additionally calculated, for each screenplay, the mean words per sentence, and put these numbers too into an array. As mentioned above, to maintain data consistency, we used the mean number of words feature to filter out irrelevant scripts, where due to a lack of punctuations, we are unable to identify the mean length of sentences.

Thirdly, we fitted a TF-IDF vectorizer from the sklearn package and calculated weighted TF-IDF frequency matrices for each of the texts.

Lastly, we employed pos_tag() function from nltk package to perform part-of-speech tagging to each script after word tokenization. There are three features that come out of this function: the percentages of nouns (tags that start with N), of verbs, and of adjectives respectively among all lexical items (the total number of all nouns, verbs, and adjectives) in each script. Later we combine these three individual ones into one feature matrix.

To feed into the model, which take numpy arrays, we called the numpy function .column_stack(), to select and put combinations of the above features together. Our six original features are: the number of sentences (baseline), TF-IDF, mean-words-per-sentence, TF-IDF combined with mean-words-per-sentence, POS tags, and TF-IDF combined with both mean-words-per-sentence and POS tags, (see Tables 3 & 4) for the validation stage, and for the final test stage we selected TF-IDF combined with mean-words-per-sentence as the best feature.

5.2 Models

The task at hand is not a classification task, but rather one that aims to predict a statistic correspondence (yet unknown if there is one) between the film scripts linguistic features and the outcome rating. We do not seek to get at an exact answer to the exact decimal point. To this end, we used both a Linear Regression model as well as a Random Forest Regression model. Since we will be comparing the two models, it is important to avoid data overlap and contamination. Thus, we split our entire dataset into training (60%), validation (20%), and test (20%), with an intention to improve and tune the model after evaluating with validation, before moving on to final comparison on the test dataset.

We first implemented the Linear Regression model from scikit-learn, based on the above features, and then implemented a Random Forest Regression model, also from scikit-learn, with the number of estimators set to 100. We used the number of sentences as the feature for our baseline, as it was our intuition as humans that this linguistic metadata would correlate the least with the rating. Then we tested the model on TF-IDF vectorizer, on the mean words per sentence alone, on combined POS tag percentages, on TF-IDF combined with mean words per sentence, and finally on TF-IDF frequency combined with the mean words per sentence feature and POS tag percentages (Tables 3 & 4).

With the availability of a validation dataset, we are able to fine-tune the model by selecting one best-performing feature to proceed to the final test dataset. The best-performing feature, it turns out, is the combined features of TF-IDF and mean words per sentence for both models. Taking the final models, we performed a final comparison on the test set with baseline feature and the TF-IDF combined with mean words per sentence feature (see Tables 5 & 6 below). We now turn to the results on validation data as well as test data.

6 Results

To evaluate the performance of our models, we used the basic accuracy metric from the models themselves as well as Pearsons correlation coefficient, since ratings involve densely distributed numbers on a decimal scale. We convert predictions of the two models on the validation/test to 1D arrays and fed them into the function pearsonr(), correlating the predictions with gold standard labels. Our results for both models, on both metrics, are reported below.

In our experiment with the linear regression model (see Table 3), we did not observe high accuracy in any of the individual or combined features. However, a combination of TF-IDF and mean length of utterances performed the best among all features both in model-reported accuracy score and Pearsons correlation score, at 0.0171 and 0.332 respectively.

The same combination of features also achieved the highest model-reported score in Random Forest model (see Table 4), at 0.845. However, the number is close to the rest of the scores, ranging from 0.813 to 0.844. Unexpectedly, the highest Pearsons correlation score for this model was achieved by our baseline.

After examining the validation results, we selected the combination of TF-IDF and mean utterance length as the feature to be used in our test dataset. The linear regression model (see Table 5) performed worse in terms of model-reported accuracy. The Random Forest model (see Table 6) produced results in line with our validation results in both model-reported accuracy and Pearson's score.

7 Discussion

In our model, we evaluated our model performance both on the accuracy score produced by the model itself and Pearsons correlation coefficient between the prediction and the gold standard label. What is crucial behind our choice of using two evaluation metrics is that movie scores fall on a continuum that might cause problems for accuracy checking, in which case Pearsons correlation coefficient might capture the relationship better. We used both linear (Linear Regression) and nonlinear (Random Forest) models in this study. Contrary to our hypothesis, the model-reported accuracy is the more consistent metric in choosing the right feature(s) for our model, ranking the same combination of features first in both of the models. In the meantime, using validation data did help us select one single feature to test on the held-out test data, preventing data contamination to either model.

To explain the low performance of the model, there is a number of factors that can be taken into consideration. From the perspective of feature selection, intuitively it makes sense that the features used in the paper do not accurately or directly reflect the quality or general perception of the film. First, our goal is a broad one. The labels include diverging levels of public rating that include various aspects of the film. Public ratings technically reflect the subjective opinions of a collective group of individuals from many distinct backgrounds and cultural settings in and outside of the US. These numbers are simultaneously influenced by individual tastes, which naturally vary by unpredictable factors. It is also important to consider whether or not the number of votes each film received is significant. In addition to the raters themselves, the rating may (or may not) depend on the theme/genre of the film, the time period of the film, the reputation of certain members on the production team, the marketing strategies of the film company, the language/region availabil-

	Model-reported accuracy	Pearson cor coef with gold standard
Baseline	0.00258	0.152
TF-IDF	-0.182	0.153
mean words/sent	-0.0103	0.0351
TF-IDF + mean words/sent	0.0171	0.332
POS Percentage	-0.0142	0.0691
TF-IDF, mean, POS tags combined	-0.183	0.182

Table 3: Two models on the validation dataset, respectively

	Model-reported accuracy	Pearson correlation with gold standard
Baseline	0.835	0.159
TF-IDF	0.841	-0.120
mean words/sent	0.813	-0.042
TF-IDF + mean words/sent	0.845	-0.021
POS percentage	0.838	-0.083
TF-IDF, mean, POS tags combined	0.845	-0.125

Table 4: Random Forest on validation - n_estimator = 100

	Model-reported accuracy	Pearson Correlation Coefficient
Baseline	-0.0319	-0.0568
TFIDF + mean	-0.696	-0.0330

Table 5: Linear regression on test

	Model-reported accuracy	Pearson Correlation Coefficient
Baseline	0.862	-0.0540
TFIDF + mean	0.885	-0.0330

Table 6: Random forest Regression results on Test

ity of the film, etc. Since there is currently no space to simultaneously treat all of these factors at once, only modeling the linguistic features might not generate an ideal result for such a complex and dynamic label. Secondly, even in our treatment and engineering process of the linguistic features, not many of them reflect deep textual information, and none are lexically-weighted, meaning that we did not examine certain semantics of lexical items. Most of our current features involve simply counting in one way or another, thus we also do not know whether these additional features can benefit our predictions. Again, it seems that to achieve a complex end goal, one would not lose in adding more, and more relevant semantic and lexical features on top of the metalinguistic, count features.

From the perspective of model selection, we found the Random Forest Regression model to perform better than the linear regression model after fine-tuning is implemented. This model provided the most consistent results compared to the Linear Regression model. However, more improvements to the models themselves are expected in the future.

8 Conclusion and Future Work

In this study, we tried to build models that predict movie ratings based on features extracted from movie scripts. Of the two models we built, namely with Linear Regression and Random Forest Regression, the latter one exhibited better performance in predicting the ratings. However, both of our models only outperformed the baseline by a short margin, and are not always consistent. In our future work, we plan to make use of additional movie metadata such as scene descriptions, that we think is crucial in capturing the public opinion of movies. In addition, applying neural network to train the model could possibly improve the performance of prediction.

9 References

Eliashberg, J, Hui, S. K., & Zhang, Z. J. (2007) Assessing Box Office Performance Using Movie Scripts: A Kernel-Based Approach. *IEEE Transactions on Knowledge and Data Engineering*, 26(11), 2639-2648.

Hsu, P., Shen Y. & Xie X. (2014) Predicting Movies User Ratings with Imdb Attributes. In: Miao D., Pedrycz W., lzak D., Peters G., Hu Q. & Wang R. (Eds.), *Rough Sets and Knowledge Technology* (pp. 444-453). New York, NY: Springer.

Lee, K., Park, J., Kim, I. & Choi, K. (2018). Predicting movie success with machine learning techniques: ways to improve accuracy. *Information Systems Frontiers*, 20(3), 577588.

Martinez, V. R., Somandepalli, K., Singla, K., Ramakrishna, A., Uhls, Y. T., & Narayanan, S. (2019). Violence Rating Prediction from Movie Scripts.

Peck, K. (2015). The Economic Impact of the MPAA Rating System on Types of Films Made From 2004-2014 (master dissertation). Retrieved from Digital Scholarship@UNLV.

Prag, J., & Casavant, J. (1994). An empirical study of the determinants of revenues and marketing expenditures in the motion picture industry. Journal of Cultural Economics, 18(3), 217235.

Walker, M. A., Lin, G. I. & Sawyer J. E. (2012). An Annotated Corpus of Film Dialogue for Learning and Characterizing Character Style. *In Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC)*.