The Automated Script Reviewer

Katherine Schinkel
University of Virginia
Data Science Institute
kms6bn@virginia.edu

Marcus Rosti
University of Virginia
Data Science Institute
mer3ef@virginia.edu

ABSTRACT

This paper discusses the application of text mining techniques and regression analysis to two 90s sitcoms as a means to predict the shows' viewership and rating. We collected ratings from IMDb and viewership from Wikipedia's reported Nielsen ratings. We apply TF-IDF and Latent Dirichlet Allocation to a vector space model of sitcom scripts, utilizing the results as predictors to our regression model.

Keywords

Text Mining; Regression Analysis; Machine Learning; Automated Systems

1. INTRODUCTION

Producing a television show is an enormously expensive exercise. For example, according to Priceonomics, a pilot conservatively costs $2 million to develop a typically 30 minute comedy or over $5 million to produce an hour long drama.[1] That's a substantial investment into a project that may never be seen by the public. Additionally, 98% of scripts fail to be made into pilots and of those, only about half will lead to a full season.

This creative and expensive process touches numerous individuals, from pitching ideas for shows to actual filming the season. We will focus on the inefficient script screening and development process. Typically, a production company will field many ideas and make a call for scripts based on those ideas. A producer must then screen through scripts and manually rate them using only their intuition, which could be swayed by cognitive bias towards a particular writer. In addition to the valuable time a producer wastes reading poor scripts, the chosen script must maximize the chance of developing into a hit show.

Our research addresses automating the script review process by providing assistance to the reviewer via an automated system. We apply text preprocessing and feature engineering to predict the continuous values of IMDb ratings and Nielsen viewership ratings. We chose these two values as our prediction value because a producer primarily cares about two things when developing a show: the show should be high quality and widely viewed. To measure high quality, the script needs to be well written and should receive a high IMDb value when it airs. To measure successful viewership, the show needs to be viewed by many millions people to drive advertising (or generate more subscriptions in the case of subscription services like HBO and Netflix). For our model to be successful, it will need to be able to predict these values with substantial certainty.

We used two 90s Sitcoms, both being widely viewed and rated, to train and test our models. Given that these shows have proliferated American culture, we expect to see some predictive power to our approach and conclude that there is a correlation between the script and the audience response.

[1]http://priceonomics.com/the-economics-of-a-hit-tv-show/

CS Text Mining Spring '16

DOI: 10.1145/1235

2. LITERATURE REVIEW

Researchers have explored sentiment analysis of text thoroughly. Bo Pang and Lillian Lee explored a multiclass approach to categorizing movie reviews as either positive, average or negative. They used support vector machines and a one versus all separation in order to classify each review [1]. That research expanded a traditional binary classification into a multivalued problem.

In a separate field, Saxena explored applying data mining techniques to screen resumes as part of a job application. In this case, Saxena explains that as the internet becomes more popular, people are able to apply to many more jobs, flooding online job postings with resumes. To parse through these resumes, companies can apply several information retrieval techniques. Saxena focuses on skills extraction, finding unique features and dealing with the specific setting of the resume. They experienced lackluster results and concluded that more research would be required to create a more confident parser [2].

Lastly, Pivotal, a data sciences research company, formulated a script reviewer with success. They extracted information from television scripts which included speaker and scene delegation and applied extensive feature engineering. For example, they were able to determine personalities of speakers via topic modeling and include these variables in their ElasticNet regression model [3]. However, the authors did not reveal details of their modeling and approach since it was a proprietary project for a television producing client.

3. METHOD OVERVIEW

Our model follows a standard pipelined approach and employs 10 fold cross validation to validate results.

3.1 Data Parsing

To clean the text, we follow a standard text mining pipeline of stemming, tokenizing and removing stop words. This process removes any words that are too frequent to be meaningful in predicting the outcome of an episode. Stemming combines words of different tenses and uses to further decrease the vector space model.

The two response variables are the IMDb mean rating for the episode and the Nielsen rating for viewership rating. For the IMDb rating, we paired each episode with its corresponding rating. The rating of the episode should stand on its own since it could come days weeks or years after the actual episode airs. Our claim here is that the rating is independent of the timing of the episodes air. However, for the Nielsen rating, it would not make sense to pair the episodes viewership with the actual episode. Consider someone watching the show. That person has already come to the decision to watch the show independent of the script since it is unfolding before them. We, instead, map the following episode's viewership with that episode; moreover, Script_i corresponds to Viewership_{i+1}. We chose this approach assuming that if a person were to watch an episode and enjoy the script, then they may be more likely to watch the next episode. In this case, we captured the temporal aspect of the airing.

3.2 Regression Inputs

We explored two options as predictive inputs to our regression model: a TF-IDF vector space model and a topic model.

3.2.1 TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) is a method of weighting tokens in a language model so as to highlight import words. It weights the number of times a word appears in a document (the Term Frequency) by the inverse of the number of documents that word appears in all documents in the training set. So for example, a stop word like 'a' will appear in every single document. A typical IDF metric will force that value to result in a 0 thus removing the term from our model.

Our input matrix to the regression equation is a document term matrix where the values in each cell are that weighted TF-IDF value.

3.2.2 Topic Modeling

Latent Dirichlet Allocation (LDA) is a generative language model used to formulate topics from a document corpus. For each document in the corpus, the LDA model provides a vector of probabilities representing how likely each topic applies to that document. This allows each document to contain multiple topics, mirroring the complexity of a single sitcom episode.

3.3 Model Validation

We followed a testing procedure based on 10 fold cross validation. In this procedure we compared our regression model, built on each training fold, with the mean of the response, again on the training fold. We use our model to predict the response on the test set and compare it to the mean of the response on the training set. Then we employ a two sample t-test to determine whether our model exceeds the performance of the mean.

$$t^* = \frac{\bar{Y}_1 - \bar{Y}_2}{\sqrt{s_1^2/N_1 + s_2^2/N_2}} \tag{1}$$

4. RESULTS

Our null hypothesis was that the model performed just as well as the mean and our alternate hypothesis is that the model outperforms the mean in terms of mean squared error.

Method	Target	Mean	Model	t value
TF-IDF	IMDb	0.134	0.175	-1.713
TF-IDF	Nielsen	50.018	70.269	-1.226
LDA	IMDb	0.133	0.132	0.062
LDA	Nielsen	45.858	44.872	0.096

5. ANALYSIS

Our model did not outperform the mean in any of the four tests by a statistical margin. Our TF-IDF model underperformed the mean. Our topic modeling did outperform the mean by a small margin but so small that the t-value is nearly zero. This indicates that a model based on the topics of the episode and the words used in the episode, is not more effective than a naive approach of guessing the viewership and rating by mean. Since the results did not show a stronger correlation, we do not recommend using only these values as predictors to a regression model.

6. FUTURE WORK

Although our modeling results indicate that there may not be a relationship between television scripts and ratings, applying more extensive feature engineering could supplement our model. Our modeling was limited to unigrams from transcripts. Future work could explore incorporating bigrams or trigrams to capture important phrases within episodes. Additionally, Pivotal achieved success with extracting metadata from the transcripts, such as count of scenes in an episode and count of characters in a scene. Pivotal also extracted speaker characteristics and indicated that they were significant to their model.

Our model was trained on a small sample size of around 250 sitcom scripts. In the future, we would like supplement our model with a larger training set of documents. Training an LDA model with a larger dataset could result in more meaningful topics that could be significant to the linear model.

Future work could also include exploring other modeling techniques. Pivotal implemented an ElasticNet regression model to significant variables (indicated by a linear model applied to each variable). Alternatively, we could explore this topic as a classification problem by binning episodes into categories. In this case, we could apply SVM, a method that proved to be successful by Pang and Lee when classifying multiclass ratings [1].

7. REFERENCES

[1] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 115–124, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[2] C. Saxena. Enhancing productivity of recruitment process using data mining and text mining tools. 2011.

[3] J. Vawdrey. Using data science to predict tv viewer behavior and formulate a hit tv show, 2015.