Assignment2

Sai Wu
a53212368
saw044@ucsd.edu

Tianyi Wang
A53216574
tiw163@eng.ucsd.edu

***Abstract*—We explored a dataset of movie reviews. We built a regressor from movie review text to the user ratings of the movie. We used techniques including TF-IDF, LDA and Linear Regression. We tested our predictor against common bag-of-words baselines and it outperforms those baselines.**

I. Task 1

A. What data we choose for prediction task?

We choose the dataset named "Web data: Amazon movie reviews" from http://snap.stanford.edu/data/web-Movies.html. The dataset consists of movie reviews from amazon. The original dataset has nearly 7,911,684 reviews with 889,176 number of users and 253,059 number of products. There are 8 features for each review which are productID, userID, profileName, helpfulness, score, time, sumary, and reviewText.

B. some exploratory statistics of this dataset.

We choose first 50000 reviews from the dataset as considering the runtime of program.

The total number of user in 50000 data is 36409, which has the global average rating of 4.1187800000000001. The user with the most number of reviews is userId'A3LZGLA88K0LA0' with 56 reviews.

And the total number of products in 50000 data is 1412. The most popular product with most reviews is itemId 'B002VL2PTU' with 455 reviews.The least popular product is itemId'B003VHELLI' etc with only 1 review.

The longest review text is written by user 'A3JZOITOIP90EW' with length of 4639 when he or she was rating the product productId'B001DDY6NU'. The shortest review text is of length 0, same with our expectation.

the total number of words that occured in reviewText is 112163.

the total number of words that have occured in summary is 22592.

C. analysis of the exploration

From the fig.1, we can find that the most of users would like to write reviews for just several movies, only few of them wrote more than 10 reviews, specially only one user wrote reviews for 56 movies. So the average number of review of each user may not be a useful feature.

From the fig.2, there is a interesting result that most of reviews concentrate in a small part of movies and usually the movies with higher number of reviews also has a higher ratings. So we can conclude that, people would like to write

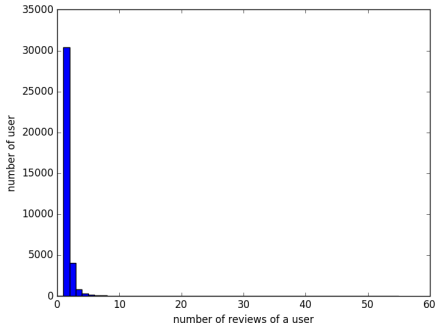

Fig. 1. histogram of number of reviews by a user

good reviews for the movies they like but barely are willing to write bad reviews for the movies they don't like. So the number of reviews is a good feature for predict the rating of movies. the average ratings of each movie is also a expressive feature for predicting the rating of each movie.

From fig.3, which is the maximum length of each movies, it is obvious that several movies have very long reviews with thousands of words, and a big part of them have around 1000-words-length reviews as their longest reviews. A small part of them have hundreds-of-words reviews as their longest reviews. The maximum length of a movie is different for different movies. We think it is a good feature for our prediction.

From fig.4, which is the maximum length of the summary of a product, We find that they are mainly the same. However, the summary text are always concise and expressive, so the text of summary is important for the prediction.

From fig.5, which is the maximum length of the summary from a user, this feature is not so expressive since most of the user are review the same movies.

From fig.6, which is the number of users in different ratings. The user that gives 5 stars are the most, they have 27957 people. Users usually tends to give high ratings for the product they have bought. So the number of low ratings is a important featrue for the prediction.

From fig.7 fig.8, We can conclude that even if there are 112163 words occurred in review, only 500 words are most common. It is same in the summary.

II. Predictive Task

A. Task

Given the dataset, we want to learn the relationship between the review and the rating of the movie. By common

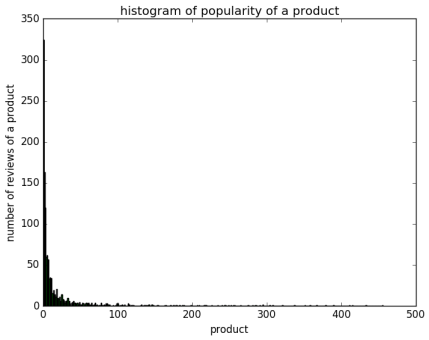

Fig. 2. histogram of popularity of product

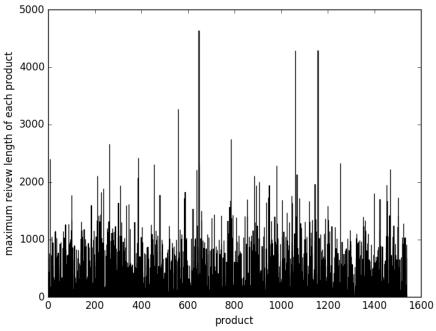

Fig. 3. maximum length review of each product

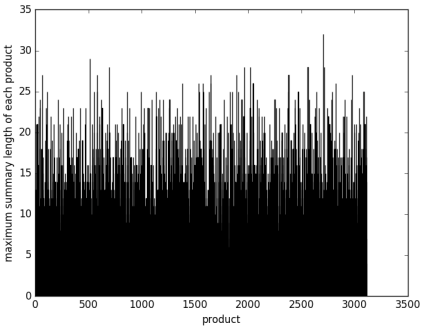

Fig. 4. maxlength of summary of each product

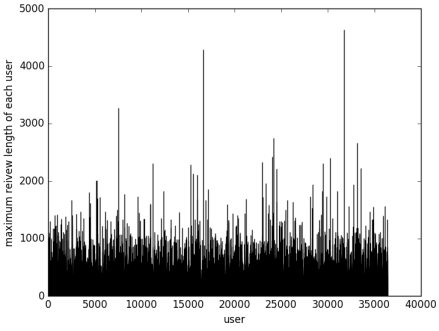

Fig. 5. maxlength of review of each user

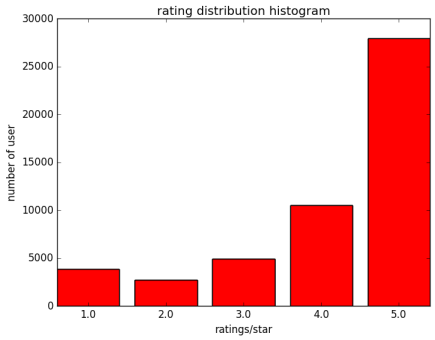

Fig. 6. rating of distribution

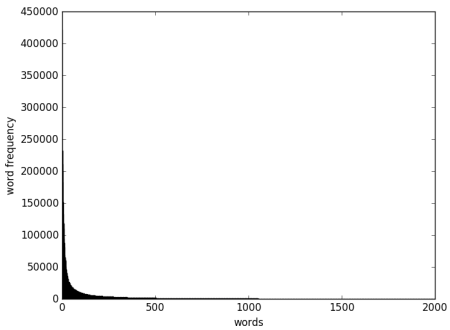

Fig. 7. words frequency in review

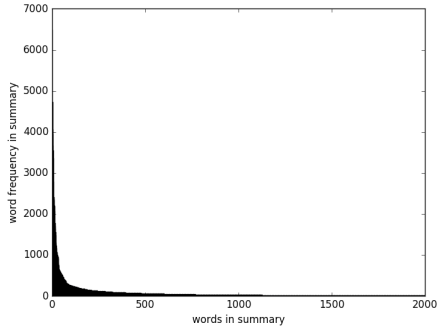

Fig. 8. words frequency in summary

sense we know that the user's rating of the movie correlates to the review text. So we want to solve this problem by some text mining techniques.

B. Baselines

Some possible baseline might be directly use TF, TF-IDF methods and then use some classifier like naive bayes or SVM. Note that directly using TF or TF-IDF generates large features, which makes training unpractical. So we only test those baselines with linear regressor, which is faster on large feature spaces. Other regressors are discussed and evaluaed with our model.

C. Evaluation

We want to evaluate the methods by testing it on a test set. We transform 50000 reviews into training set and test set with a 2:1 split. Because the label is continuous, the performance statistics we choose is mean squared error. This is not perfect, because most of the ratings concentrate on 4.0 5.0. But considering real-world usage of review prediction, a false-positive does not do much harm to businesses, so we stick with MSE.

D. Features

The features we use are review text, review summary and movie ID. Review text reflects the user's attitude towards the movie. The review summary is a more precise representation and the movie ID indicates the average rating of the specific movie. We explore different methods of text mining to generate second order features from review/summary text, as described below.

III. MODEL

We choose to combine different text mining models in lectures together. We first generate TF-IDF of review summary. This is important because the summary is supposed to contain positive or negative words directly and the number of words in summary is limited so we can afford a TF-IDF.

Then we generate TF data of review texts. This TF data is extremely large so we use LDA to reduce the dimension of features. In our evaluation we would show that this is better than directly limit the number of words in TF.

The reasoning of different choice is that the number of words used in review summary is limited, so we want a more accurate representation of it, while the number of words used in review text is extremely large so choosing top K words from review text is not representative. Treating summary and review as equal also does not sounds good because summary are shorter and more important. So here we want to reduce the number of dimension smoothly, using some topic model. In this way we can get features in summary and review text with reasonable size of features.

The mean rating of the reviewed movie is also added into features. If the movie does not appear in training set, global average rating of training set is applied. This feature helps because in our exploration of data (I-C), most reviews are of 30 movies, which makes knowledge of movie more important. On the other hand (1), the number reviews made by a specific user is mostly below 5, so we didn't added knowledge about users into the features.

After generating features as described we put them into a linear regressor for classification. After careful experiment and parameter tuning we chose linear regressor with L2 regularization and the penalty is set to 1. We also tested SVM and KNN regressor and their performance will be discussed in the evaluation section.

IV. LITERATURE

LDA is considered for dimension reduction for predictive tasks at the beginning of its invention. In [1], a SVM is used to predict the category of text documents. In our setting, the topics directly reflects the viewer's attitudes towards the movie. When the topic is positive we want the rating to go up. So a linear regression seems more promising, and our experiments show that it is.

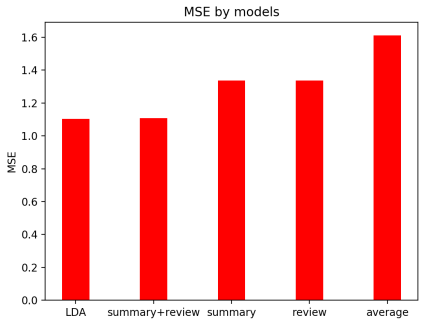

Fig. 9. Baseline Comparison

LDA is considered not suitable for supervised tasks including rating-review prediction because in review texts topics is usually about genre rather than rating. Good model would underscore "excellent, terrible, and average, without regard to genre. SLDA [3] is developed instead to address these problems. But in this assignment we are recommended to use techniques in lectures so we didn't study sLDA.

While researchers tried to describe topic using n-gram modeling, bag-of-words models are still state-of-the-art methods of predicting from texts.

This dataset is originally [2] used to study the user's 'experience' of recommended products, which model the user's taste over time.

V. RESULT

A. BaseLines

In this section we discuss the performance of different baselines compared to our model on this task.

The simplest baseline is predicting the average rating in the training set, which is around 4.15. Doing so gives us a MSE of 1.61.

The second baseline is to use TF-IDF data of summary to generate features for the predictor. The number of words in summary is limited, so we only use 200 dimensions of TF-IDF data.

Another baseline is to use TF-IDF data of review text to generate features for the predictor. The number of different words here is more than summary, so we use 1000 dimensions of TF-IDF data.

The last baseline is to combine the two above with the average rating of a movie. That is 1200 dimensions.

Our model uses 100 dimension LDA features from review text and 200 TF-IDF features from summary text. That is 300-dimension feature, which is far less than last method.

Fig. 10. Word Clouds of 9 topics(out of 10)

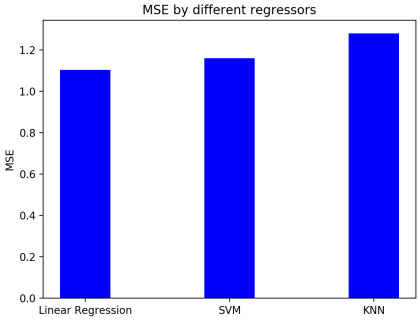

Fig. 11. MSE of different regressors

A plot 9 shows the performance difference between baselines and our model. LDA is our model. The following 3 uses TF-IDF of summary and/or review. Our model outperforms all the others. Note that TF-IDF of summary+review archieves similar MSE with much more feature dimension.

B. LDA Topics

We plotted 9 topics and their top words in 10. From the figure we can see that LDA is able to distinguish different kind of review and the genre of movie.

The coefficient of these features in the linear regression is more interesting. The coefficients are: -0.80547872, -0.14526847, 0.74437846, 0.27355388, 0.65141045, 0.16426161, -0.01606676, -0.26031115, -0.64585471. The implications is that action sci-fi movie with good effect usually receive good ratings. Movie with multiple seasons and episodeses may have better rating because the users are fans. Controversial movies with mixed review would have lower rating because the average rating of other movies are high. If the review is all about DVD disc, the ratings would be low because customers talking about disk usually indicates that there is a problem in DVD version.

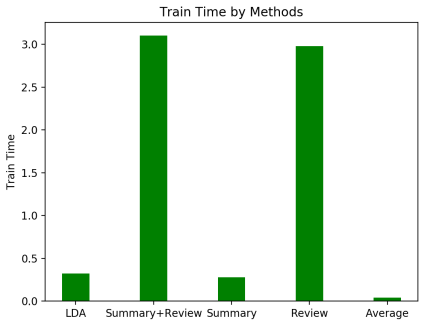

Fig. 12. Performance Comparison

C. Different Predictors

We tested SVM and KNN as substitutions of linear regression in our model. See 11 for the figure. They cannot outperform linear regressor. The reason behind this is that LDA has identified positive or negative topics, so we do not need a plane to classify the data. KNN is not so good because our feature space still has hundreds of dimensions, and we don't have enough data to cover a large part of the space.

D. Computational Performance

We plot in 12 the training time of different choice of features. We can see that the original big TF-IDF features in review text takes a lot of time. LDA reduced 75% feature space than our baseline. This performance is in the setting of linear regression. If we use TF-IDF features with other regressors like SVM with rbf kernel, our program cannot give a result in limited time.

VI. Conclusion

We showed our exploration and the development of a predictor in the movie review dataset. Our model gets similar result with full TF-IDF features but has significantly less features and better performance. The LDA can generate topic categories in a human-comprehensible way and the topic generated does not loss much information than the orignal data. Our exploration on topics also shows some interesting phenomenon in movie reviews.

References

[1] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. *Latent dirichlet allocation*. J. Mach. Learn. Res. 3 (March 2003), 993-1022.

[2] Julian John McAuley and Jure Leskovec. 2013. *From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews*. In Proceedings of the 22nd international conference on World Wide Web (WWW '13). ACM, New York, NY, USA, 897-908

[3] Mcauliffe, Jon D., and David M. Blei. *Supervised topic models*. Advances in neural information processing systems. 2008.