Amazon Fine Food Reviews' Rating Score Prediction

Jingwen Zhong and Chenglu Xia

***Abstract*— The purpose of this study is to help enhance the awareness of customer review trends for the Amazon food marketing team by analyzing information from customer reviews and predicting their rating scores. By performing text preprocessing using NLP techniques and data visualization such as world cloud, a deeper understanding of the customer's opinions is obtained. Our model of predicting the rating scores of food reviews using regression methods achieved a resulting 1.04 RMSE.**

I. INTRODUCTION

The reviewing system on Amazon is always a great resource to help users choose products and gain more information about the product. Observing and comparing and reviews and ratings is an important way to understand a product better. Higher ratings for a product from a customer usually mean it has better reviews content. The correlation between the sentiment of the reviews and rating scores should be tight.

The Amozon fine food reviews dataset from Kaggle contains 10 variables and more than 500,000 observations which were retrieved from Oct. 1999 to Oct.2021. The response variable in our project is rating "Score" and we included 20 predictors in our final models.

In this paper, we conduct several natural language processing techniques such as LDA(Latent Dirichlet Allocation), sentiment analysis, etc.on summary and text of the review. we analyze individual features such as reviews score, time of the review, helpfulness of the review, and create several data visualisations such as word cloud, heatmap to better understand the dataset. We find that the similarity between summary of the review and text of the review is not obvious, the average cosine similarity score is only 0.19 out of 1. We also perform linear regression, ridge regression and lasso regression after data reprocessing, feature selection and get the lowest RMSE score of 1.04 from linear regression model.

II. METHODS

A. Data Preprocessing and Feature Selection

As for the missing values in "Summary" and "ProfileName" columns, we implement them with a space " ".

For the features of our model, we consider 20 attributes in total. one original features, "Time", in the raw dataset is included. We convert HelpfulnessNumerator and HelpfulnessDenominator to "Helpfulnessratio" by using Helpfulness Numerator divided Helpfulness Denominator. The length of the review summary and length of the review text are also be included.

We expect the review summary and review text are highly correlated with the review score, so that we put our main focus on the text data preprocessing.

We first create clean review summary and review text by removing stop-words, words that shorter than 3 characters, links, emojis, punctuation, and several string that are not English words. We calculate the Cosine similarity score of the summary and review text, as consider the similarity score as one of our attributes. We consider the sentiment of the review summary and sentiment of review text as two of the most important features. To find the sentiment score, we use the python nltk sentiment package, and use the compound score to represent the sentiment.

In addition to the review sentiment, we believe some punctuation like exclamation mark and emojis such as :) or :(can be important factors affecting scores. We create six variables based on the exclamation mark for both summary and text, and four variables based one the smile and sad emojis. Two numerical variable are the number of exclamation marks in summary and text. The other variables are all binary variables. One example of the rule to create them is this, for the variable named Sum-pos-punc, if there is exclamation mark in summary of this customer and the sentiment of the summary is positive, then we record the Sum-pos-punc of the customer to 1, otherwise we record it to 0. Based on rules like this, we create variable named Sum-pos-punc, Sum-neg-punc, Summary-smile, Summary-sad, Text-pos-punc, Text-neg-punc, Text-smile, and Text-neg-punc.

For outliers, we calculate Z score of the summary sentiment and text sentiment and use 2.58 of Z score as the threshold to determine the outliers. we perform the regression with and without the outliers, and the RMSE result shows that included outliers can bring us the lower RMSE. Therefore we do not include this step in our final model.

This dataset is very imbalanced, there are a large amount (about 63%) of customers' reviews with score 5, but reviews with score 1, 2, 3, and 4 has only around 5% - 15% each. We use several oversampling method such as boostramp, SMOTE to balance the data, but the result does not become better. Therefore we do not include this step in our final model.

Moreover, we conduct the heatmap for the correlation between all variables and the target variable "Score", summary sentiment and text sentiment are more correlated with Score, and variables that are related to exclamation mark are also important. However, reducing features or reducing dimensions will only do little to our regression model and the result does not turn better, so we include all of the features we created in our final models.

B. Model selection - Linear Regression

We perform linear, ridge and Lasso regression to predict the score. For linear regression, we build a OLS model using all features. It is worth noting that, we use the interaction term of the summary sentiment and text sentiment. The RMSE for the final linear regression model of all data is 1.046. We perform cross validation grid search for both ridge regression and lasso regression to get the best alpha with lowest RMSE score. For ridge regression, the best alpha is 0.3 with the RMSE score 1.0540. For lass regression, the best alpha is 0 with the RMSE score 1.0529. According to the RMSE, we chose linear regression as our final model.

III. Results

A. Table

	min	mean	50%	max
Text_length	12.0	436.222083	302.0	21409.0
Summary_length	0.0	23.445744	20.0	128.0
Score	1.0	4.183199	5.0	5.0
Helpfulness_ratio	0.0	0.407862	0.0	3.0

Fig. 1. Summary Table

Fig.1. Summary Table shows the minimum, average, median and maximum for the following features: text length, summary length, score and helpfulness ratio.

B. Four line graphs

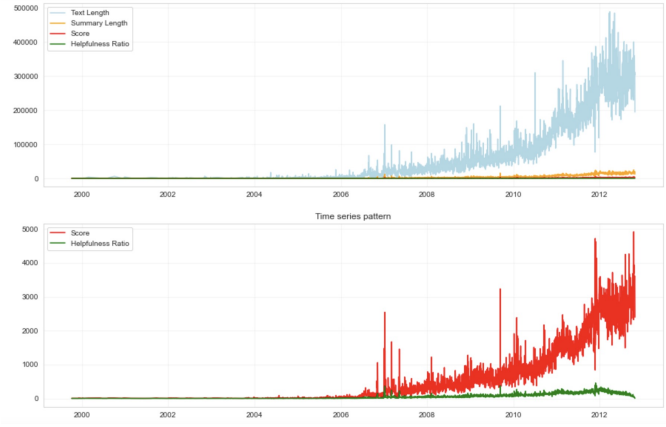

Fig. 2. Four Line Graph

According to Fig.2, compared those four lines, the aggregated values of review length, summary length, score, helpfulness ratio all have a increasing trend by day over time. Since they are aggregated value, it means more people bought the products as time goes by. The text length increased dramatically. To make the line of text length look clear, we dropped rows that have text length greater that 600,000. The line of helpfulness ratio seems more stable and flat than the other three. In addition, the slopes of these four lines are steeper after year 2006 .

Fig. 3. Word Cloud

```
Topics found via LDA:

Topic #0:
good great chocolate like amazon product taste love just price

Topic #1:
like taste good flavor just great chips water salt really

Topic #2:
product food cat just like cats use time good chicken

Topic #3:
food dog dogs treats loves like product treat love old

Topic #4:
coffee tea like flavor taste cup good drink just great
```

Fig. 4. LDA Analysis-5 Topic

C. Latent Dirichlet Allocation(LDA)

From Fig.4, there are 5 topics and each topic contains 10 words by using LDA analysis. The first topic focus on chocolate. Second one is about chips and the third one talks about cat food. The next one describes dog food and the last one mainly discuss coffee. There also included many positive words in those five topics.

D. Non-negative Matrix Factorization

Compared with the result of LDA analysis, the results of NMF have some differences. In LDA, cat and dog are in different topics, but in NMF they are in the same cluster. Besides, in NMF there are more descriptive words which don't appear in LDA clusters, such as 'sweet', 'roast' and 'green'. The two plots also show the counts of each words in clusters.

E. Similarity Analysis

The mode of cosine score between text and summary are 0 which means there is no similarity between text and summary for most reviews. The average of cosine score is 0.19, showing the similarity is not obvious in average. In conclusion, summary does not match text well from cosine scores. The maximum value is 1 and the minimum is 0.

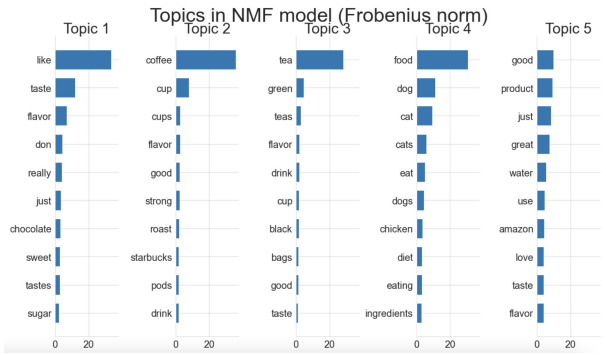

Fig. 5. NMF Analysis-Frobenius norm

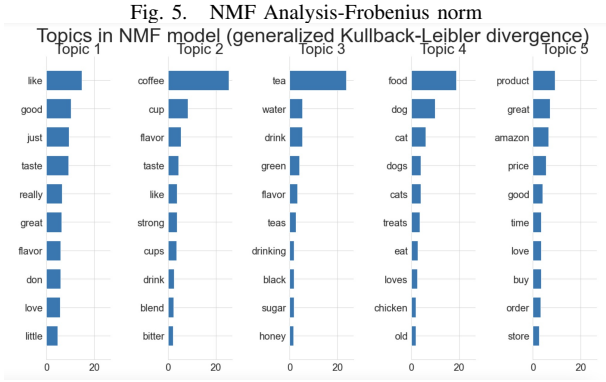

Fig. 6. NMF Analysis-generalized Kullback-Leibler divergence

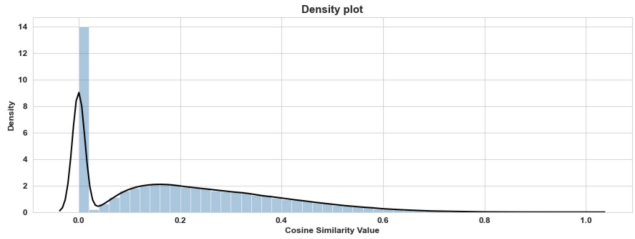

Fig. 7. Similarity Between Text and Summary

F. Modeling-RMSE Score

According to Table 1, the performance of linear regression is the best with the lowest RMSE score. Thus, we choose linear regression model to predict the rating score of amazon food reviews.

Model	Score(RMSE)
Linear Regression	1.046
Ridge Regressions	1.054
Lasso Regression	1.052

TABLE I

TABLE OF RMSE SCORES

G. Modeling-Confusion Matrix

Fig.8 shows the confusion matrix by using linear regression model. Compared the total number of each rating score between actual and predicted, the amount of predicted rating score '1' and '4' are less than the actual amount.

Predicted	1.0	2.0	3.0	4.0	5.0	All
Actual						
1	6806	10885	17567	15801	1209	52268
2	1707	4222	9557	12900	1383	29769
3	768	2838	10043	23635	5356	42640
4	226	1267	6491	40682	31989	80655
5	343	1996	13188	133480	214115	363122
All	9850	21208	56846	226498	254052	568454

Fig. 8. Confusion Matrix

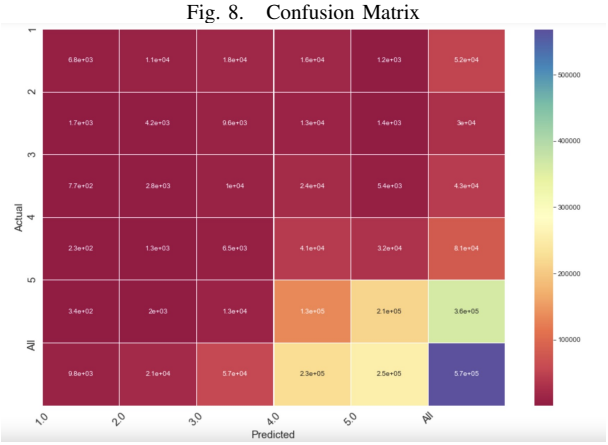

Fig. 9. Heatmap of Confusion Matrix

REFERENCES

[1] S. Kapadia, "Topic Modeling in Python: Latent Dirichlet Allocation (LDA)", Retrieved Sep.23, 2021 from: https://towardsdatascience.com/end-to-end-topic-modeling-in-python-latent-dirichlet-allocation-lda-35ce4ed6b3e0.

[2] Topic extraction with Non-negative Matrix Factorization and Latent Dirichlet Allocation"", Retrieved Sep.23, 2021 from: https://scikit-learn.org/stable/auto/applications/plot_topics_extractionw_ith_nmf_lda.html