Product Rating Prediction based on reviews from RentTheRunway Clothing Data

LING JIANG[1], JIAOYANG WANG[1], AND XINYUE JIN[1]

[1] *Rady School of Management in University of California, San Diego*

Predict the user rating on certain product based on RentTheRunway dataset. Applied methods including bag of words , linear regression, and latent factor model which over-performed the baseline method by over 30%.

Keywords: regression, TF-IDF, Bag of Words, Latent Factor Model

1. INTRODUCTION

Nowadays, more and more people tend to rent well-designed clothes for one-time uses instead of spending money on less practical clothes, so they can save money at that same time look shiny. Under this situation, online renting has become a booming business model, and how to improve the user satisfaction is the key factor for all retailers.

Rent the Runway is one of the most popular online renting websites. Founded in 2009, it has grown rapidly and had lots of data about their customers and transactions. In this report, we looked deep into the basic statistics of the RentTheRunway dataset and got a basic understanding on the characteristics of the user base. We also tried to analyze customers' reviews on clothes, and predict their ratings on clothes based on Rent the Runway clothing dataset to help retailers get a deeper understanding about their customers and products.

We applied Bag of Words, linear regression, and latent factor model on several different features, including unigrams and bigrams, and compared the results of different models. Our experiments showed that our models over-performed the baseline by 10 %-30 %, which can be considered as a reliable reference for the retailers.

2. LITERATURE REVIEW

Our dataset was collected by Julian McAuley, Rishabh Misra and Mengting Wan [1]. In the previous research, they put forward a predictive framework to solve a product fit problem, which captured the semantics contained in customers' feedback. They used a latent factor formulation to decompose the semantics of customers' fit feedback and adopt a metric learning approach. Besides, they conducted experiments by comparing the performance of five methods(1-LV-LR, K-LV-LR, K-LF -LR, K-LV-ML, K-LF-ML), which included the effectiveness of capturing fit sentiments of "true" sizes and of the proposed metric learning approach.

The method proposed in this paper is beyond our objective. In order to get a big picture of how the business has developed in recent years and the popularity of each product, we depicted the distribution of ratings during 2010-2018, and found out the most popular category of clothes and the most popular occasions that people rent clothes. In this paper, we intended to predict users' ratings based on review words, which is categorized as the problem of sentimental analysis (SA). In many studies, the bag-of-words representation of text is commonly used. However, it may be important to use the alternative bigram model to capture the sequence of words. Machine learning algorithms such as linear regression of TF-IDF and latent factor model are frequently used to make predictions.

SA is important because it helps business holders understand opinions from user' reviews and improve their services. SA can be applied in other fields as well. Such examples include predictions of stock market from business news, and predictions of election results based on social networks.

There are two major techniques to solve SA problems [?]: the lexicon-based approach which uses pre-compiled sediment terms, and machine learning approaches. The latter can be further divided into the probabilistic model such as Native Bayes, the linear model such as Support Vector Machine and Neural Networks. It is worth noting that linear model is capable of making good results despite of its simplicity, but it depends on carefully selected features.

3. DATASET

After our group discussion, we decided to use RentTheRunway Reviews dataset for analysis. This data consists of reviews of clothes renting from RentTheRunway. The dataset covers a period from November 2010 to January 2018, including 192,462 reviews from real users. It provides the information of product and user profile, ratings, the reason for renting, the review text, the summary of the review, clothes category, and review date. Based on this dataset, we did an exploratory analysis to find out the characteristics of the dataset.

A. Data summary

There are in total 192,462 reviews in our dataset which contains the reviews from 105,508 users on 5,850 products. Each record has several features, including the user ID, the item ID, the rating of the user on the product, reason for renting, the review text, the summary of the review, clothes category, and review date. The description of each feature is given in Table 1.

B. Data analysis

Before the prediction, we try to dig out more valuable facts from our dataset, such as user rating distribution, product popularity distribution and the word cloud representation of all the reviews.

B.1. The rating distribution

First, we plotted the rating distribution of the dataset to see how people tend to rate their products. The result is shown as Figure 1. The graph shows that most users rated as five and fewest users rated as one. It indicates that most customers were satisfied with the clothes that they rented. Thus, we can believe that when people rated below 4, it usually means that they were really unhappy with the purchase. We can also see the distribution of ratings over the years. The result is shown as Figure 2. As the year went closer to 2017, more ratings were given by the customers, and more top ratings tended to be given.

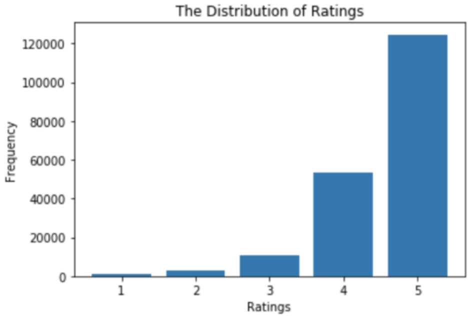

Fig. 1. Rating distribution

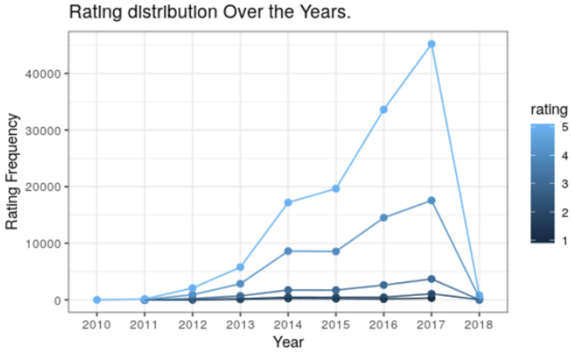

Fig. 2. Rating distribution over the years.

B.2. The product popularity distribution

We also tried to analyze the popularity of products in the dataset by comparing the number of reviews for each unique product. In other words, the product with the highest number of reviews is regarded as the most popular product. Then we extracted the 20 most popular products with the graph as Figure 3. From the graph we can see that the most popular product (item id 126335) has around 2300 reviews.

B.3. The renting reason distribution

We plotted the renting reason distribution of the dataset to see why people rent the clothes. The result is shown as Figure 4. In the plot, there is only 1 observation with the reason "party:cocktail". The graph shows that the most frequent reason for users to rent the clothes is wedding (the exact number is 57,768) and the frequency of this reason is much higher than other ones.

Table 1. Data summary of *RentTheRunway Clothing Reviews*

Feature Name	Description
fit	Whether the clothes are fit (small, fit, large)
userId	Unique identifier for the user
itemId	Unique identifier for the product
rating	Ratings from users (between 1 and 5)
rentedFor	Reasons for renting
reviewText	Texts in users' reviews
reviewSummary	A brief summary of reviews
category	Type of clothing
age	Age of each user
reviewDate	The date the users published reviews

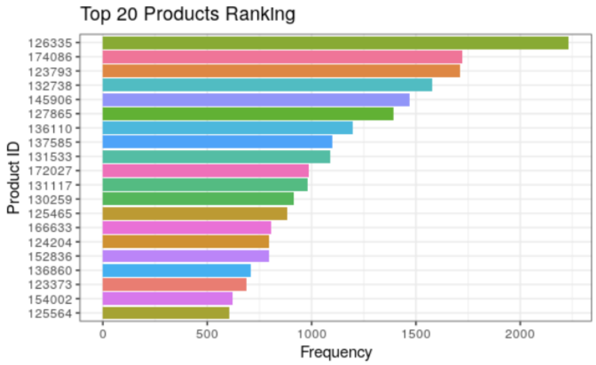

Fig. 3. Product popularity distribution

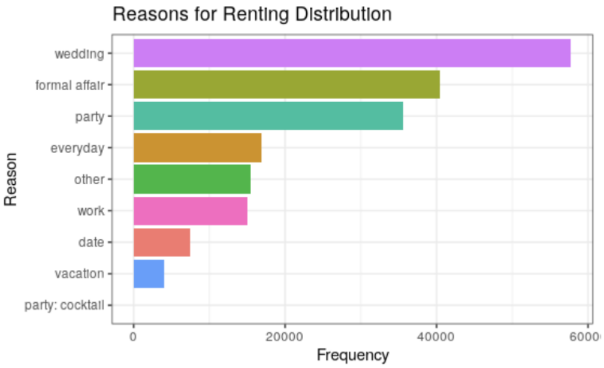

Fig. 4. Renting reason distribution

B.4. The category distribution

Finally, we plotted the category distribution of the dataset to see which type of clothes people tend to rent more. The result is shown as Figure 5. The graph shows that dress is the type of clothes which customers tend to rent mostly. It can be explained by reasons such as wedding, formal affairs and party (the top 3 reasons).

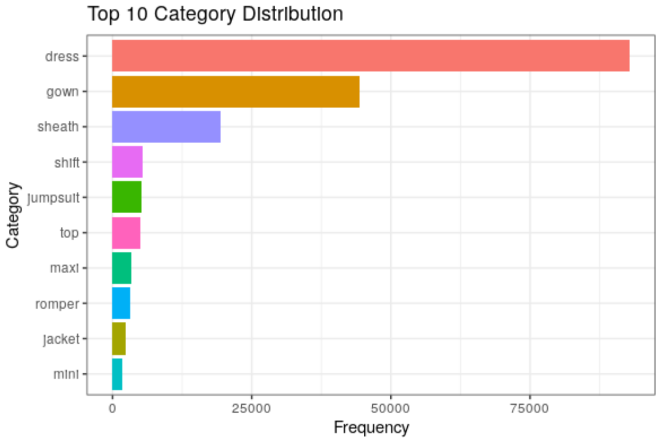

Fig. 5. category distribution

B.5. What people say when they make positive/negative rating

Besides the distribution of ratings, popularity of products, reasons, and category, we tried to find out what people would comment when they had a positive attitude or negative attitude. To distinguish the positive and negative attitude, we use the feature "rating" as the indicator. If the rating is 1 or 2, we regard it as a negative attitude; if the rating is 4 or 5, we regard it as a positive attitude. We treat rating 3 as a neutral one (neither positive nor negative).

We extracted the top 100 words that are used most frequently in each attitude group by using words cloud representation. The results are shown as Figure 6 and Figure 7. Both graphs show that the word "dress" is the top word to be mentioned in both attitudes. This is because of the extremely high frequency of dress rented by users. From Figure 6, we can find out that the words "size", "fit" and "comfortable" exist in the positive attitude graph. These words indicate the aspects of quality that customers care about. From Figure 7, we can also see words such as "fabric", indicating the aspects that the retailer needs to improve to meet the customers' needs.

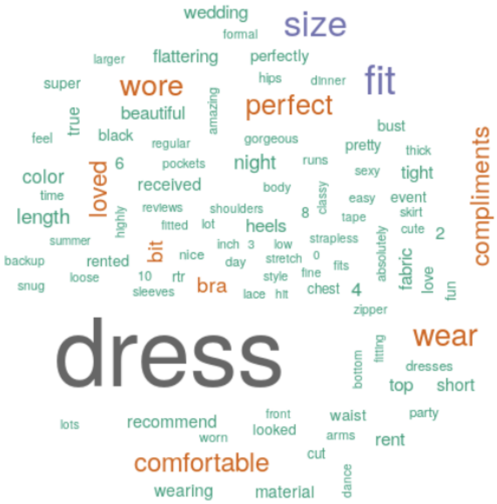

Fig. 6. Top 100 Words in Positive Reviews

4. PREDICTIVE TASK

We made predictions on rating scores from 1 to 5 by making sediment analysis on the review texts. The models will be evaluated using Mean Squared Error (MSE). To make our predictions valid, we trained and evaluated our model on the training and validation set respectively, and reported the true performance on the test set. We set 80% of data as the training set, 10% of data as the validation set, and remaining 10% of data as the test set.

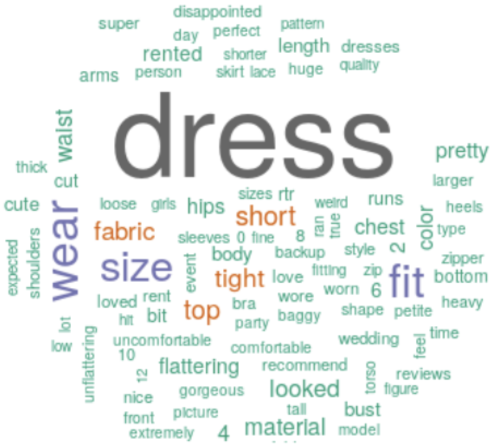

Fig. 7. Top 100 Words in Negative Reviews

A. Feature selection

Since we are modeling on SA problems, we use the following representations of words and use them as features in our models.

A.1. N-gram Model

The n-gram model indicates a sequence of n-words. Specifically, the unigram model extracts individual word from the text. It can be easily obtained, but it has obvious disadvantages of losing information on the order and combination of words. In contrast, the bigram model honors partial order of words by preserving the 2-word sequence, but it also increases the number of entries compared to the unigram model.

With n-gram models, features such as word counts or word frequency can be constructed to help us identify the importance of these n-word sequences.

A.2. TF-IDF

TF-IDF, frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is a combination of test frequency and inverse document frequency, where the two terminologies are defined below.

The definition for TF is:

$$tf(t,d) = \frac{countoft}{wordsind}$$

i.e., the number of times the term t appears in the document d.

The definition for IDF is:

$$idf(t,D) = log(\frac{N}{|\{d \in D : t \in d\}|})$$

where t is the term and D is the set of documents.

The definition for TF-IDF is:

$$tfidf(t,d,D) = tf(t,d) \times idf(t,D)$$

Every piece of word is taken into the calculation for TF-IDF. In order to conduct it precisely, we have a data pre-processing for 'Text' feature. To format our data and build the Term-doc incidence matrix, many operations will be performed on the data :

- Snowball Stemming the word
- Removing Stop-words
- Convert the word to lowercase
- Remove any punctuation or limited set of special characters like ',' or '.' etc.

B. Models and evaluation

We can apply several supervised learning models to this prediction task. And We chose linear regression on n-grams and TF-IDF, and latent factor model(LFM). We also chose a baseline models for predicting ratings. The models will be evaluated using Mean Squared Error (MSE), which is the average squared difference between the estimated values and the actual value.

5. MODELS

A. The baseline model

For our task, we set the baseline model as simply predicting the rating by using the average ratings for each user, or return the global average if we have never seen the user before.

$$f(t) = \alpha$$

B. Linear regression on N-gram words

B.1. Unigram

The words in the review texts can show users' attitude (positive or negative) and the ratings can be predicted by attitude. If a word of the review text has a high frequency shown in the positive attitude, we can predict a high rating; on the opposite side, if a word of the review text has a high frequency shown in the negative attitude, we can predict a low rating. Thus, we used the 2000 most common unigrams to generate the feature matrix, and applied linear regression model to predict the ratings.

We experimented with ridge regression techniques, which attempts to shrink the norm of the predictors by minimizing the penalized residual sum of squares.

$$argmin_x = \frac{1}{N}||Ax - y||_2^2 + \lambda||x||_2^2$$

B.2. Bigram

Similar to above, instead of using unigram words, we also used the 2000 most common bigram words to predict ratings with linear regression model. The reason is that we need to consider the situation that there may be phrases which show positive or negative attitudes.

B.3. Unigram and bigram

To be more precise, we used the 2000 most common combination of unigram and bigram words to do the linear regression instead of only using unigram words or bigram words.

C. Linear regression on TF-IDF

We used TF-IDF technique to generate the feature matrix for unigram and bigram words, such that

$$\theta^T X_i = \theta_0 + \theta_1 tfidf_1 + \theta_2 tfidf_2 + ... + \theta_n tfidf_n$$

where $tfidf_i$ means $tfidf$ for $word_i$.

We did tf-idf on both unigram and bigram.

D. Latent Factor Model

We used the Simple (bias only) Latent Factor Model to predict ratings of each product, such that

$$f(u,i) = \alpha + \beta_u + \beta_i$$

α: the average value of rating for all the training data

β_u: the user rating bias between personal rating tendency and average rating value

β_i: the item rating bias between books received rating and average rating value

The optimization procedure of the model is

$$argmin_{\alpha,\beta} \Sigma_{u,i} (\alpha + \beta_u + \beta_i - R_{u,i})^2 + \lambda[\Sigma_u \beta_u^2 + \Sigma_i \beta_i^2]$$

and we did the following procedures:

- Initialized the α, β_u and β_i to be 0. Calculated the global average rating.
- Trained the Latent Factor Model without gamma, get the new α, β_u and β_i.
- Fixed the α, β_u and β_i using the following procedure until convergence.

$$\alpha = \frac{\Sigma_{u,i}\,(R_{u,i} - (\beta_u + \beta_i))}{N}$$

$$\beta_u = \frac{\Sigma_i (R_{u,i} - (\alpha + \beta_i))}{\lambda + |I_u|}$$

$$\beta_i = \frac{\Sigma_u (R_{u,i} - (\alpha + \beta_u))}{\lambda + |I_i|}$$

For β_u and β_i, we used two different lambdas to convergence. The differentiate procedure included both MSE and Regularizer.

6. RESULTS

A. The baseline model

For the rating prediction task, the MSE of the baseline prediction model is 0.6183 and the global average for rating in the training data $\alpha = 4.55$.

B. Linear regression on N-gram words

B.1. Unigram

The model MSE for the 2000 most common unigram segment on testing set is 0.3799, we listed some significant positive and negative words used in reviews as Table 2 and Table 3. The coefficients corresponding to words represent the relatives to positive or negative rating. The larger the coefficient is, the more likely the word in positive rating will be. We can see from the chart that most significant words in either positive review or negative one are all the extreme emotional words such as happier, disappointing and matronly, which make a lot sense. But it also includes in some neutral words such as movie and miller, which are less conceivable. So we tried to use bigram features to get even more persuasive results.

Table 2. Top 5 positive unigram words

Word	Coefficient
happier	0.2574
nicole	0.2197
movie	0.2023
glove	0.1862
perfection	0.1842

Table 3. Top 5 negative unigram words

Word	Coefficient
disappointing	-0.7863
unflattering	-0.6220
unable	-0.4115
swimming	-0.3973
matronly	-0.3906

B.2. Bigram

The model MSE for the 2000 most common bigram segment on testing set is 0.4001, we listed some significant positive and negative words used in reviews as Table 4 and Table 5. We can see from this bigram chart that compared to unigram features, bigram features can use more accurate word combinations to indicate positiveness or negativeness of a product. It does not contain neutral words so that it makes the model more reliable.

Table 4. Top 5 positive bigram words

Word	Coefficient
wanted keep	0.3096
like dream	0.2977
fits perfectly	0.2604
like million	0.2564
highly recommended	0.2516

B.3. Unigram and bnigram

The model MSE for the 2000 most common combination of unigram and bigram words is 0.3859, which is larger than unigram words model. We listed some significant positive and negative words used in reviews as Table 6 and Table 7. From this chart, we can find that the top positive and negative words are all unigram words, indicating that single words,

Table 5. Top 5 negative bigram words

Word	Coefficient
unable wear	-0.7998
without wearing	-0.7474
wouldn't rent	-0.690
wasn't flattering	-0.6624
couldn't even	-0.5839

compared to bigram words, have more effects on the final ratings.

Table 6. Top 5 positive unigram and bigram words

Word	Coefficient
happier	0.2600
perfection	0.1949
incredible	0.1925
dream	0.1903
buying	0.1890

Table 7. Top 5 negative unigram and bigram words

Word	Coefficient
unflattering	-0.6394
unable	-0.4217
disappointed	-0.3727
strange	-0.3689
odd	-0.345

C. Linear regression on TF-IDF

In order to use the linear regression model on tf-idf, firstly we extract every comment from data. Then calculate $TF-IDF$ for every segment in comments. After that, we use the linear regression to assign weight(coefficient) to words. The larger the coefficient, the better the rating. Since unigram words have more effects on ratings, we only use unigram segment to train the model.

The model MSE for unigram segment is 0.5690, which is also larger than linear regression model with unigram words. We listed some significant positive and negative words used in reviews as Table 8 and Table 9. From the chart, we can see there are words with negative attitude have high coefficient, which may affect ratings positively. This might be the reason that cause the performance of this model worse than others.

Table 8. Top 5 positive words

Word	Coefficient
worried	2.9756
perfect	2.8160
compliments	2.6774
nicole	2.6729
bra	2.6690

Table 9. Top 5 negative words

Word	Coefficient
disappointing	-7.3639
unfortunately	-4.5117
strange	-4.3307
disappointed	-4.2112
odd	-4.0400

D. Latent Factor Model

The model MSE for Latent Factor Model is 0.4809. In this model, we included different λ_u and λ_i for $\beta_u ser$ and $\beta_i tem$ with 10 and 15 respectively. After the optimization procedure, the global average rating is 4.52.

Comparisons of model MSE are described in Table 10. We can see that all models work well that they perform better than baseline. Among them, linear regression on unigram words has the lowest MSE and linear regression on TF-IDF has the lowest, but it is still higher significantly than baseline model.

Note that linear regression on unigram and the combination of unigram and bigram have higher accuracy compared to other models. But Linear regression on TF-IDF is even worse than the latent factor model. This is because from the Figure 6 and Figure 7, we can find that lots of words have both positive and negative attitude in the review text, such as "fit" and "size". And TF-IDF is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus, the higher frequency the word appears in the text, the less important it is. Thus, those words, which appear in the reviewing text frequently, would have less influence on the predicted ratingsbut actually they are important. Besides, from the significant positive and negative words table of this model, we can see there are words with negative attitude have high coefficients, which may affect ratings positively. On the other hand, linear regression on unigram and the combination of unigram and bigram take the phrases into consideration, resulting in better performance.

Table 10. Model Comparison in MSE

Models	overall MSE
Naive baseline model	0.6183
Linear regression on unigram words	0.3799
Linear regression on bigram words	0.4001
Linear regression on unigram and bigram words	0.3859
Linear regression on TF-IDF (with regularization)	0.5690
Latent factor model	0.4809

7. TUNING PARAMETERS

A. Logistic regression on tf-idf

In terms of avoiding over-fitting, we add the regularization term and range parameter λ with 1, 0.05,0.005 to find the optimal value. It gains minimum MSE 0.3859 when = 0.05, which is described in the Table 11 .

Table 11. MSE on different λ

λ	MSE
1	0.5463
0.05	0.3859
0.005	0.3859

B. LFM

In latent factor model, before we do the training process, it is necessary to find optimal λ_u and λ_i for optimization process. We tried different λ pairs, and compared their performance to find an optimal model. Finally, the model gains minium MSE 0.4809 when λ_u = 10 and λ_i = 15, which is described in the Table 12.

Table 12. MSE on different λ_u and λ_i

λ_u	λ_i	MSE
2	15	0.4988
5	15	0.4873
10	15	0.4861

8. SUMMARY

In this assignment, we tried to make sentiment analysis based on the reviews in the RentTheRunway clothing dataset. We chose our baseline model and improved the performance by using alternative methods including linear regression and latent factor model. For linear regression, we applied n-gram (unigram, bigram) model and used tf-idf as the features. We found that the combination of unigram and bigram model helped increase the performance because it preserved the order of word pairs and improved both performance and scalability.

REFERENCES

1. Rishabh Misra, Mengting Wan, Julian McAuley, Decomposing fit semantics for product size recommendation in metric spaces,RecSys, 2018