Recommender System with Advanced ratings

Julia Camille Menchavez
jmenchav@usc.edu

Varidhi Garg
varidhig@usc.edu

Neelam Yadav
nyadav@usc.edu

Abstract

Yelp currently provides only a single star rating for a business. This project provides a way to generate "advanced" ratings for restaurants in particular, where each restaurant has ratings for different attributes, namely: Service, Ambience, Value, and Food using Non-Negative Matrix Factorization for topic modeling and Logistic Regression for sentiment analysis. Additionally, we have also developed a recommender system using model-based collaborative filtering to recommend restaurants to users depending on their preferred attribute.

Introduction

Currently, when a user visits the page for a business on Yelp, the user sees just a single rating on a scale of 1 to 5 stars. While this gives users a general idea of the business's quality, it does not present a more detailed breakdown of what this rating pertains to. If for example, the quality of service at a restaurant is extremely important for a Yelp user, he or she cannot immediately see whether this single rating reflects the service without having to go through numerous reviews.

In order to recommend users restaurants that meet their preferred attributes (service, quality, food,ambience), we have used topic modeling and sentiment analysis to determine "advanced ratings" for restaurants, as well as model-based collaborative filtering to recommend these restaurants to users based on their preferred attributes.

Dataset

After the initial inspection of "yelp_academic_dataset_business.json", it was concluded that all the businesses which offer food have at least one attribute which mentions restaurant in some way, such as RestaurantPrice, RestaurantParking etc. The IDs of those businesses were extracted from this file. All the reviews from given data were then filtered to keep reviews for only restaurants. For preprocessing of reviews, all the punctuations were replaced either by space or no space depending on type of punctuation and the words were lemmatized.

Method

In order to provide advanced ratings for restaurants and recommend restaurants according to users' preferred restaurant attributes (service, quality, food,ambience), the project has four components: (1) topic modeling, (2) sentiment analysis, (3) advanced rating generation and (4) model-based collaborative filtering.

1. Topic Modeling

1.1 Feature Extraction

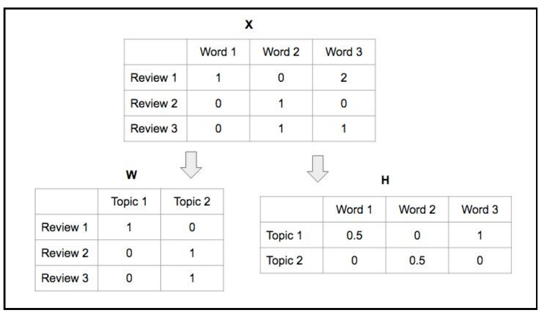
For extracting features from the restaurant reviews, scikit-learn's TF-IDF vectorizer was used, which converts a set of documents into a matrix of TF-IDF values. The default parameters were used.

1.2 Approach

Non-Negative Matrix Factorization (NMF) was used in determining the topics / attributes present in the restaurant reviews, as well as in determining the topics of new reviews.

Given an original matrix X, NMF seeks to find two non-negative matrices W and H whose product is an approximation of X.

$$minimize \, \|X - WH\|_F^2 \; w.r.t. \, W, H \; s.t. \, W, H \geq 0$$



X

	Word 1	Word 2	Word 3
Review 1	1	0	2
Review 2	0	1	0
Review 3	0	1	1

W

	Topic 1	Topic 2
Review 1	1	0
Review 2	0	1
Review 3	0	1

H

	Word 1	Word 2	Word 3
Topic 1	0.5	0	1
Topic 2	0	0.5	0

Fig1: NMF topic generation

As shown in Fig1 above, given a term-document matrix with the tf-idf values, NMF generates a document-topic matrix and a topic-word matrix, the latter of which yields the topics and corresponding word weights in the topics. In this project, scikit-learn's NMF function was used, with the following parameters: random_state=1, alpha=.1, and the number of topics set to 5.

2. Sentiment Analysis

2.1 Feature Extraction

For extracting features from the restaurant reviews, scikit-learn's count vectorizer was used, which converts a set of documents into a matrix of term counts. The default parameters were used.

2.2 Approach

For sentiment analysis, a logistic regression model was trained to determine whether sentences from reviews were positive or negative. In order to establish the ground truth during the training process, we extracted 1-star reviews and labeled them as "negative" and labeled 5-star reviews as

Fig 2: negative review

Fig 3: positive review

3. Advanced Rating Generation

In order to generate the "advanced rating" for a specific restaurant, we utilized both the (1) topic modeling and (2) sentiment analysis components. An advanced rating consists of ratings for each of the attributes of a restaurant.

Fig 4: Advanced Rating

For each review of a restaurant, we determined the topics and sentiments of each sentence and kept a count of each of the sentiments per topic.

$$result[tp\,\#][snt] \;=\; \#sentences\ under\ tp\#\ with\ that\ snt$$

$tp = topic, snt = sentiment$

We then consolidated these results of all the reviews of a particular restaurant in order to determine its advanced rating by calculating the totals per sentiment per topic. To compute the rating of a topic, we took the average of its sentiments.

$$Topic\ rating\ = \frac{\#\ positive\ sentences}{(\#\ positive\ sentences + \#\ negative\ sentences)}$$

4. Recommender System

4.1 Approach

For the recommender system, we employed model-based collaborative filtering using Spark MLlib's Alternating Least Squares (ALS) algorithm implementation. We created a user-business-topic rating matrix which was passed into the ALS model's train method. With this trained model, we could then give recommendations to new users.

```
{0: [[0, 0, 0.0026451238869648198],
  [0, 1, 0.005054705947058394],
  [0, 2, 7.8010818103285901e-05],
  [0, 3, 0.0035756574035006768],
  [0, 4, 0.0068851066496616346],
  [0, 5, 0.0028264380466669729],
  [0, 6, 0.0035602231904757848],
  [0, 7, 0.01207634987953436],
  [0, 8, 0.0022562623915937737],
  [0, 9, 0.0035679347327163139],
  [1, 10, 0.0],
  [2, 11, 0.0],
  [3, 12, 0.0],
  [3, 13, 0.0066607235183550779],
  [3, 14, 0.0084240271197718406],
  [4, 15, 0.0017778934827041458],
  [4, 16, 0.00031899582854194922],
  [4, 17, 0.0014068369719646794],
```

*Fig 5: Partial (*user-business-topic rating matrix) *which is passed to ALS model*

Experimental Approach

1. Topic Modeling

For modeling the topics in the reviews, we experimented with 3 different topic modeling implementations and algorithms, namely (1) Gensim's Latent Dirichlet Allocation (2) scikit-learn's LDA (3) scikit-learn Non-Negative Matrix Factorization

1.1 Latent Dirichlet Allocation with Gensim

Using Gensim's LDA module, we experimented with generating topics on 10,000 restaurant reviews, 100,000 reviews, and 1,000,000 reviews. We also experimented with generating 5 topics and 10 topics, as well as setting the number of passes during training to 1, 5, and 15.

The changes in number of reviews and passes did not cause a large difference in the topics generated and their corresponding words. Shown below is the result of generating 5 topics and training on 10,000 restaurant reviews.

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
place	time	order	food	time
good	service	piazza	good	place
like	go	come	place	come
service	tell	food	great	great
great	work	service	time	go
order	custom	restaurant	service	like
food	get	like	come	best
time	need	go	like	want
go	come	place	order	good
price	say	great	tri	clea

Fig 6: Topic words generated by Gensim's LDA

1.2 Latent Dirichlet Allocation with scikit-learn

We also performed similar experiments with scikit-learn's LDA module, generating topics on 10,000 restaurant reviews, 100,000 reviews, and 1,000,000 reviews as well as number of iterations. Shown below is the result of generating 5 topics and training on 100,000 restaurant reviews.

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
order	really	Like	Place	Food
ve	Ordered	Place	Staff	Good
time	Chicken	Come	Restaurant	Great
menu	delicious	People	Friendly	Service
don	Definitely	Didn	Pizza	Just
best	Fresh	Better	Service	Got
like	Sauce	Eat	Nice	Place
bar	Meal	make	Love	Went

Fig 7: Topic words generated by scikit-learn LDA

1.3 Non-negative Matrix Factorization with scikit-learn

After the results obtained by Latent Dirichlet Allocation, we took a new approach in generating topic models, this time using Non-negative Matrix Factorization. Using scikit-learns NMF module, we were able to train a model that generated more cohesive topics and corresponding words. Shown below is the result of generating 5 topics on 100,000 restaurant reviews.

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
great	good	place	food	service
atmosphere	really	love	delicious	friendly
experience	pretty	recommend	amazing	excellent
prices	pizza	definitely	excellent	customer
selection	experience	like	fresh	staff
staff	chicken	time	quality	slow
beer	overall	try	just	fast
pizza	prices	really	ok	attentive
overall	price	just	best	horrible
price	selection	nice	came	terrible

Fig 8 : Topic words generated by NMF

Another, final experiment was then done this time using better pre-processed and cleaned reviews, with the resulting topics shown in the final Results section Fig.13.

1.4 Narrowing down topics

In order to determine the number of topics or attributes that we wanted to focus on for the restaurants, we generated the topic distribution for NMF. Fig.9 below shows the topic distribution of the reviews when NMF was used to generate 5 topics. As seen in the figure, 61.4% of the restaurant reviews fell under topic 4. Upon inspection of the topic words in Fig.13, we realized that this was due to topic 4 containing words common in reviews such as “try”, “wait”, “just” and “like”, which do not necessarily indicate a specific topic or attribute.

After removing topic #4, we generated the topic distribution again show on Fig.10. This was more equally-distributed and was the basis for our taking into account only 4 topics going forward.

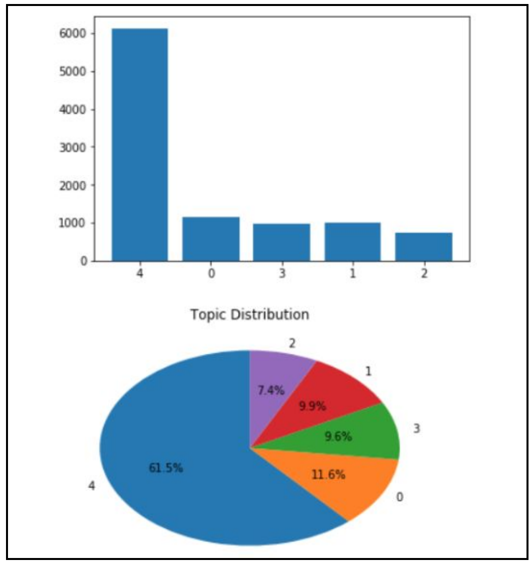


Fig 9 : Topic distribution for 5 topics

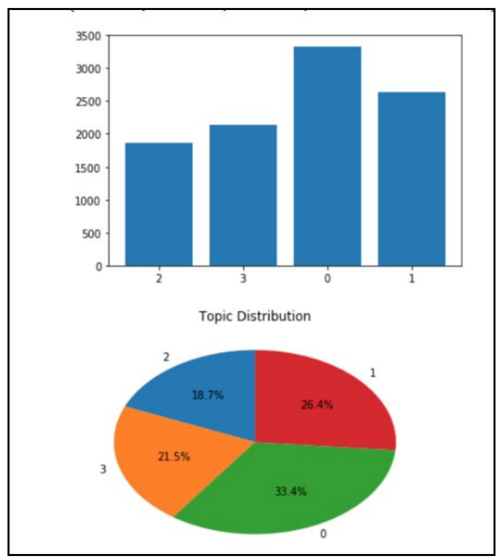


Fig10 : Topic distribution for 4 topics

2. Sentiment Analysis

2.1 Positive negative list approach

For the first approach, we made use of an Opinion Lexicon (Hu and Liu, KDD-2004), downloading lists of positive and negative words. For each of the reviews, each word appearing in the positive list was replaced with "GOOD" and each word appearing in the negative list was replaced with "BAD". Negation handling was also done where phrases such as "not good" were replaced with "bad". We then did a simple count on these words to determine sentiment.

This approach achieved an accuracy of 66% when trained on a total of 100,000 restaurant reviews with 75% used for training and 25% used for testing.

2.2 Star rating approach

For the second approach, we utilized the star ratings of reviews to establish the ground truth. Reviews with 1 star were considered negative and those with 5 star reviews were considered positive. Utilizing 25,000 1 star reviews and 25,000 negative reviews, we trained and tested the model and achieved an accuracy of 95%.

3. Recommender System

For the recommender system, we initially trained and tested on 1,000 restaurant reviews to determine approaches for handling restaurants that were not present in training. The final results in the Results and Discussion section used 200,000 restaurant reviews.

3.1 Without average rating

Without using a proper heuristic for setting values for restaurants not present in the training set, we obtained the MSEs in Figure 11.

Topic Model	MSE
Topic 0	.576
Topic 1	.375
Topic 2	.541
Topic 3	.482

Fig 11 : MSE per topic without average rating

3.2 With average rating

Given the negative results in the previous approach, we instead decided to use a user's average rating per topic whenever a new restaurant was seen in the test set, which achieved the following results.

Topic Model	MSE
Topic 0	.184
Topic 1	.079
Topic 2	.291
Topic 3	.270

Fig 12 : MSE per topic with average rating

Results and Discussion

1. Topic Modeling

When using Non-Negative Matrix Factorization on 100,000 cleaned restaurant reviews, we determined the following topics/attributes for restaurants.

Service	Ambience	Value	Food	Topic 4(N/A)
great service friendly staff atmosphere customer experience excellent price nice	place love recommend like really try amaze highly awesome nice	good really service pretty price taste pizza overall fry portion	food delicious amaze excellent price fresh quality best ok awesome	time come order definitely try wait just like eat make

Fig 13 : Topic words generated by NMF on cleaned data

To evaluate the topics, we read through the reviews classified under each topic. Some of the sample reviews and the topics they have been classified under are shown below.

Some sample output:
Input : The servers check on you often which gives you the best experience
Output : [6.764e-04, 3.321e-04, 6.188e-05,3.764e-04, 2.520e-03]
Topic Selected : Service (as Topic 4 will be ignored)

Input : The servings and portion are huge and true value for money.
Output : [0.00011222, 0. , 0.00039899, 0.00032327, 0.00060412]
Topic Selected : Value (as Topic 4 will be ignored)

2. Sentiment Analysis

As mentioned previously, we utilized 25,000 1 star reviews and 25,000 negative reviews, to train and test a Logistic Regression model. The data set was split into 75% for training and 25% for testing. We achieved an accuracy of 95% on the test set.

3. Recommender System

For training and testing the final version of the recommender system, we used 200,000 restaurant reviews with 70% for training and 30% for testing and achieved the following MSEs:

Topic Model	MSE
Topic 0	.016
Topic 1	.069
Topic 2	.095
Topic 3	.071

Fig 14 : MSE per topic with average rating using cleaned data

4. Conclusion and Next Steps

At the end of this project, we were able to devise a way to generate “advanced ratings” for restaurants with topic modeling and sentiment analysis, With MSEs of .016, .069, .095, and .071 for the Service, Ambience, Value, and Food attributes respectively for the recommender system, we were able to recommend users restaurants that aligned with their preferred attribute. To improve the study in the future, other factors could also be taken to account when generating the ratings. Other existing factors such as the “$” scale of a restaurant could also be incorporated into the advanced rating through weights, in order to better recommend restaurants to users. Additionally, the topic and sentiment analysis models could be trained on a larger number of reviews given more computing resources.

Appendix

Github link:

https://github.com/varidhigarg/INF553-AdvancedRatings

Contribution:

Julia Menchavez: Topic Modeling
Varidhi Garg: Sentiment Analysis
Neelam Yadav: Recommender System