# **“Honest or Dishonest, that is a Question”**
Restaurant Review Rating and Sentimental Analysis on Yelp Data

Caihong Rainbow Huang
Course Project for IMT 575 – Data Science III, Fall 2019

## Abstract

Restaurant reviews can provide valuable information about different aspects of the business. A valid and reliable review can generate more revenue for the business in question, therefore, for my project, I decided to explore dishonest restaurant reviews on the popular app, Yelp. I defined dishonesty as cases where the sentiment of a review does not align with the star rating given to the review and attempted different techniques that would analyze various text patterns in reviews to determine dishonest reviews. My dataset was sourced from the Yelp Dataset Challenge, round 13 released in January 2019 and it contains information about the businesses listed on Yelp.

I identified dishonest reviews by extracting polarity of the reviews using sentiment analysis with TorchText and detecting those whose polarity doesn’t correlate with the star rating. Only 8.7% of the total number of reviews are identified as dishonest reviews. I compared topics generated using topic modeling by LDA to identify data clusters indicative of dishonesty. I observed that the topics of honest reviews have more positive tones than dishonest reviews. I then used transfer learning on ULMFiT pre-trained model to develop a classification model to predict dishonest reviews. My prediction model achieved an accuracy of 84.43%. From my analysis, I conclude that though the text patterns are not obvious, as there are subtle differences in the topics of dishonest and honest reviews and a deep learning model trained on language model is able to predict dishonest reviews with good accuracy, there exists patterns that indicate dishonesty in reviews.

## 1. Introduction

Yelp reviews provide a valuable source of information for restaurant owners trying to improve their business. But identifying reliable reviews that provide constructive criticism amongst several thousand reviews can be tedious and time consuming. I decided to analyze text patterns in reviews to determine if I can identify the reviews that have been provided in good faith and can be used by the businesses to improve various aspects of their operations.

I came up with the below research question as the basis for my analysis: **Are there text patterns in user reviews that indicate dishonesty?**

I define dishonesty as cases where the sentiment of a review does not align with the star rating given to the restaurant. For example, a review that generates a positive sentiment polarity from sentiment analysis but receives a 1-star rating and vice versa.

## 2. Data Selection

My data is sourced from the Yelp Dataset Challenge, round 13 released in January 2019. It includes information about businesses, reviews, check-ins, tips, and user data.

In my project, I focused only on businesses and review data that was filtered out to the records pertaining to restaurants. I selected restaurant records using the keyword “restaurant” excluded the following keywords “casinos”, “hotel & travel”, and “hotel” from the “categories” field in the business table. I then matched a shared field, the business ids, to the review data file to select reviews for restaurants. The final dataset after selection contains 4,101,986 records with 105,413 restaurants.

## 3. Literature Review

I reviewed the paper which investigates the incentives for committing review fraud on Yelp (Luca & Zervas, 2015). The paper explores the economic incentives businesses must create fake reviews on Yelp and gave us the idea to explore fake reviews. My final project idea of identifying and analyzing dishonest reviews and execution process was finalized based on the discussion amongst the team members.

I referred to online articles in the process of conducting my own analysis. For text sentiment analysis, I referenced the article “VADER, IBM Watson or TextBlob: Which is Better for Unsupervised Sentiment Analysis?” (Intellica.AI, 2019). I consulted the case study of “LDA on the Texts of Harry Potter” (Rafferty, 2018) for topic modeling. I built the classification model to identify dishonest reviews based on the tutorial “Text Classification with TorchText” (PyTorch, 2017). I referred to the article “Using FastAI’s ULMFiT to make a state-of-the-art multi-class text classifier” on Medium to build my predictive model of dishonest reviews (Chinchure, 2019).

## 4. Analysis

### 4.1 Text sentiment analysis

I tested three open-source libraries for text sentiment analysis, NLTK-Vadar, TextBlob and TorchText. Vadar and TextBlob are both based on NLTK and pre-built dictionaries. Both of them give a continuous polarity score to each review ranging from -1 to 1. Vadar is very inaccurate as it gives more than 90% reviews polarity scores close to 1. TextBlob is relatively more accurate with means of the polarity scores being correlated with the star rating. TorchText (providing polarized positive/negative results) is a more sophisticated library with which I were able to train the model with the review text so that the predicted results fit better to my scenario of restaurant reviews. I got 95.3% accuracy with TorchText whereas the best case TextBlob could do (with 0.15 as threshold) was 85.7%. In this case, I decided to use TorchText to proceed with my analysis on dishonest reviews. Fig 1 and Fig 2 (Appendix) shows the comparison of performance of libraries.

Table 1. Configuration of learning process

| | |
|---|---|
| Criterion | CrossEntropyLoss |
| Optimizer | Stochastic Gradient Descent method |
| Initial Learning Rate | 6 |
| Number of epochs | 15 |

### 4.2 Identifying dishonest reviews

I used ‘YelpPolarityReview’ text classification dataset in TorchText to identify ‘dishonest’ and ‘honest’ reviews (PyTorch, 2017). The ‘TorchText’ package consists of data processing utilities and datasets for natural language (Torch Contributors Revision, 2017). YelpPolarityReview dataset consists of 1,120,000 lines for training and 38,000 lines for testing labeled as either positive or negative. A classification model is trained on YelpPolarityReview dataset, and my restaurant reviews dataset is passed through the model to be classified into positive or negative. The classification model is composed of the EmbeddingBag layer and the Linear layer (Fig 3 - Appendix). The data is loaded in parallel as bi-gram string into the model. The training dataset is split in a ratio of 0.95 (training) and 0.05 (validation). See Table 1 for the configurations used in learning process.

Table 2. Number of honest and dishonest reviews

| **Review Type** | **Number Identified** |
|---|---|
| Dishonest | 356,286 |
| Honest | 3,745,700 |

The learning rate is adjusted through the training. The model achieved an accuracy of 95.3% when evaluated on test dataset.

Restaurant reviews are passed through this model and polarity is predicted for each review. This is considered as predicted polarity. The reviews with stars 4 and 5 are assigned positive label and reviews with stars 1 to 3 are assigned negative label. This is considered as actual polarity. Then, the reviews for which actual and predicted polarity did not match are identified as dishonest reviews, otherwise honest reviews. For example, a review with predicted polarity Positive and Rating 2 is considered dishonest. Likewise, review with polarity negative and rating 5 is also considered dishonest. I did not investigate the reviews on which the model performed poorly and assumed that the model works perfectly. The number of dishonest and honest reviews identified are represented in Table 2. Table 3 and Table 4 (Appendix) provide examples of dishonest and honest reviews.

### 4.3 Exploring dishonest reviews

The top 10 restaurants that had the greatest number of dishonest reviews were identified and the percentage of dishonest reviews versus the total reviews was calculated. I observed that dishonest reviews formed only a small percentage (8.7%) of the total reviews and the top 10 restaurants were all based in Las Vegas, Nevada. (see Fig 4 in Appendix)

## 5. Topic Modeling

I performed topic modeling on both the honest and dishonest review datasets to identify if there were data clusters that could indicate differences.

### 5.1 Topic Modeling using LSA

I initially performed topic modeling using Latent Semantic Analysis (LSA) on a subset of the data of 50,000 lines. I made use of two libraries, sklearn and the Gensim, which have implemented LSA. Sample topics are generated using sklearn (see Fig. 5 in Appendix) and Gensim (see Fig. 6 in Appendix). LSA using sklearn produced more related topics as compared to the Gensim package. However, the results were still not satisfactory, and I then attempted with Latent Dirichlet Allocation (LDA).

### 5.2 Topic Modeling using LDA

I used Latent Dirichlet Allocation (LDA) to perform topic modeling. LDA is proposed by Blei et al in 2003, it is a generative probabilistic model for collections of discrete data such as text corpora. The main idea behind LDA is “bag of words”, it means that each document can be represented by a group of topics, and further, each topic can be represented by a group of words, the order of words does not matter.

Because LDA performs clustering operations on individual words, I conducted data preparation on the review text to convert them to the designated formats. First, I tokenized the review text, removed stop words, and merged words into bigrams and trigrams. To reduce dimension, I lemmatized the text using the Python Spacy library. And to increase modeling accuracy, I limited word types to nouns (NN), verbs (VBP), adjectives (ADJ), and adverbs (ADV) using NLTK word tags. (Bird et al, 2019).

I tested two variations of LDA model, the standard LDA and LDA Mallet (MAchine Learning for LanguagE Toolkit) made available by the Gensim library and compared their modeling results. These two variations differ in the calculation phase where standard LDA performs variational Bayes and Mallet performs Gibbs sampling. (Mallet, 2018) I evaluated the models’ performances using “c_v” ranging from zero to one, the higher the score the better the model.

To find the best model, I set the hyperparameter, the number of topics, from two to twenty with step sizes of two. The performance of both models increases as the number of topics increases with a few exceptions. (see fig. 7 and fig.8 - Appendix) The model with the highest coherence score contains “noise” category composing of French words I could not interpret. (see fig. 9 in

Appendix). After balancing the topic numbers, coherence score, and human interpretability, I selected the model of the second-highest coherence score (0.4263) generated using LDA Mallet with a topic number of eight as my optimal model.

After determining the optimal model, I applied the model to both honest and dishonest reviews. (see fig. 10 - Appendix) For better illustration, I turned the results into word clouds. (see fig. 11 and fig. 12) A color is used to represent a topic and the size of the word represents the probability of word in this topic; the bigger the size, the higher the probability.



Fig 11.(left) Word cloud of results of LDA Mallet (topic number = 8) on honest reviews

Fig 12.(right)Word cloud of results of LDA Mallet (topic number = 8) on dishonest reviews

I manually assigned topics based on the top ten common words the models have given. The topics I chose for the honest reviews are (from topic 0 - 7): experience, food, fast food, service, high-end restaurant food, bar, service, location. The topics I assigned for the dishonest reviews are (from topic 0 - 7): food, people, experience, service, fast food, bar, high-end restaurant food, price.

I can see the similarity of topics shared by honest and dishonest reviews as they can be both be summarized into these two categories: (1) food or restaurants of different categories, (2) descriptions or aspects of experience, such as service, food, price, location, people, or overall.

Though honest and dishonest reviews share similarities on the topics, the common words of two datasets show differences: honest reviews have more positive tones from the aspects of the number of positive words used and probabilities of these words in the topics. On the one hand, words indicating positive emotions occurred 15 times in the honest reviews but only occurred 10 times in dishonest reviews. I argue that these words used to describe honest reviews are more extreme, such as amazing, excellent, and favorite whereas descriptions of dishonest reviews are more general with higher occurrences of “good”. However, “good” is overused in present days which cannot accurately represent emotions. On the other hand, positive words occupy lower probabilities in dishonest reviews topic.

Combining the word differences and examples of honest and dishonest reviews, I inferred that usage of positive words usually points to positive emotions in a review, and reviews without using those words are more likely related to negative emotions. My explanation for this is that when people are showing compliments or appreciation, they tend to use positive words. I discovered that when encountering unpleasant situations, it is less common for people to use negative words, rather, they will try to use descriptive words that can be diverse and hard to categorize.

Table 5. Positive words and counts in honest and dishonest reviews

| Review | Positive Words | Counts |
|---|---|---|
| Honest | good(2), nice(2), fresh(2), delicious(2), love(2), top, great, excellent, amazing, favorite | 15 |
| Dishonest | good(3) , great, love, tasty, cool, fun, nice, delicious | 10 |

## 6. Prediction

I used transfer learning on fast.ai’s ULMFiT to build a classification model for predicting dishonest reviews. Transfer learning is a machine learning method where a model developed for a task is reused as a starting point for another task (Brownlee, 2017). ULMFiT stands for Universal Language Model Fine-tuning and this method involves fine-tuning a pre-trained language model trained on WikiText 103 corpus (Howard & Ruder, 2018). The technique involves creating a language model that can predict the next words in the sentence and building a classification model upon it (Chinchure, 2019).

I under sampled honest reviews to the number of dishonest reviews such that the dataset contains a total number of 712,572 (dishonest: 356,286 + honest: 356,286) reviews. The

dataset is split in a ratio of 0.7 (train) and 0.3 (test). The training set is again split in a ratio of 0.8 (train) and 0.2 (valid). See Table 6 [Appendix] for the number of records in each set.

The text of reviews is converted into “data bunch”, the format required for ULMFiT. Creating a data bunch automatically pre-processes the text including vocabulary formulation and tokenization. Tags like ‘xxbos’ that marks the beginning of the sentence and ‘xxmaj’ that indicates that the next word starts with a capital letter are appended to the text. An example of text in data bunch is given below:

“xxbos xxmaj update : xxmaj chef xxmaj chris is awesome . xxmaj he not only sent me a message late last night , but he called me this morning to personally apologize , talk with me about the experience , and insist i take a refund ( which i did not want to take because there were a lot of positives about the experience that i thought were valuable )”

One data bunch is created for language model in which labels are ignored and the model learns to predict the next words in a sentence given a starting word. Another data bunch is created for classification model in which labels play a key role. The language model is created and trained with optimal learning rate for one epoch. In this step, only the last few layers of the neural network are trained. Next, the whole neural network is trained for one epoch after unfreezing all layers. The language model achieved an accuracy of 35%. This accuracy represents how well the model performs at predicting the next word given one word.

Next, text classifier is created, and language model is loaded into the classifier. The classifier is trained using a technique called “Gradual Unfreezing”. In this technique, last few layers are trained first, then the model is trained again after unfreezing a few layers and lastly whole neural network is trained after unfreezing all layers. For my model, I initially trained the model for two epochs with optimal learning rate, then trained again for two epochs after unfreezing two layers and finally, after unfreezing all layers trained the model for fmy epochs. The model achieved a validation accuracy of 84.33%. On test set, the model achieved an accuracy of 84.43%. I generated a confusion matrix to see the model’s performance on each class.

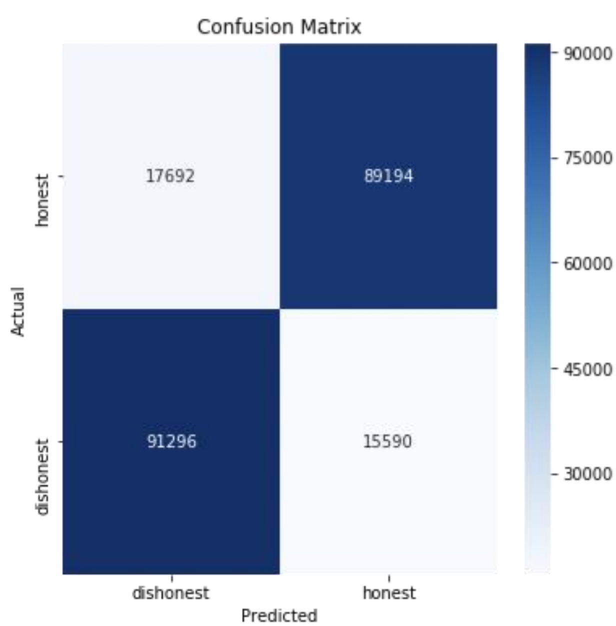



Fig. 13 Confusion Matrix

The model performed similarly in predicting both dishonest and honest reviews.

## 7. Limitations & Assumptions

- All the reviews with ratings from 1 to 3 are considered negative and 4 to 5 are considered positive while identifying dishonest reviews. A better threshold can be defined as reviews with rating 3 usually means neutral rather than negative.
- Topics were manually assigned with the support of common words generated by LDA modeling, but potential biases or errors could be introduced in the assigning process.
- My model to identify dishonest reviews has an accuracy of 95.3%. I did not eliminate reviews on which the model performed poorly from my analysis but assumed that the model predicts perfectly.
- User behavior is not considered in the analysis. Some users may always give higher ratings while others may always give lower ratings. Rather, I assumed that every user gives correct rating.

## 8. Conclusion

The primary purpose of this project was to identify if specific text patterns exist in reviews that can predict its dishonesty. Yelp itself scours the reviews to flag down fake ones, but dishonest reviews also hamper a business's potential to grow and might be harder to detect due to context and language nuances like humor, sarcasm etc.

Topic modeling did show differences in the topics generated for honest and dishonest reviews and my prediction model has an accuracy of 84.43%. This supports my assumption that there are specific text patterns in dishonest reviews that differentiate them from honest reviews. However, this conclusion is made disregarding user behavior which is a major factor in the way the reviews are written.

## 9. Future Work

For future scope, the accuracy of the predictive model for dishonest reviews might be improved with hyper parameter tuning. User behavior can be included in the analysis to better identify the dishonest reviews. A time series analysis can be conducted for restaurants or locations to see monthly or yearly trends in dishonest reviews by enlarging the data sample.

## References

Bird et al.(2019).Categorizing and Tagging Words. Retrieved from https://www.nltk.org/book/ch05.html
Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of machine Learning research*,

*3*( Jan), 993-1022.
Brownlee, J. (2017, December 20). *A Gentle Introduction to Transfer Learning for Deep Learning.* Retrieved from

Machine Learning Mastery: https://machinelearningmastery.com/transfer-learning-for-deep-learning/

Chinchure, A. (2019, July 26). *Using FastAI's ULMFiT to make a state-of-the-art multi-class text classifier*. Retrieved from Medium: https://medium.com/technonerds/using-fastais-ulmfit-to-make-a-state-of-the-art-multi-label-text-classifier-bf54e294 3e83

Howard, J., & Ruder, S. (2018, May 23). *Universal Language Model Fine-tuning for Text Classification.* Retrieved from arXiv.org: https://arxiv.org/pdf/1801.06146.pdf

Intellica. AI. (2019, March 1). VADER, IBM Watson or TextBlob: Which is Better for Unsupervised Sentiment Analysis?. Retrieved from https://medium.com/@Intellica.AI/vader-ibm-watson-or-textblob-which-is-better-for-unsupervised-sentiment-analys is-db4143a39445

Luca, M., & Zervas, G. (2015, July 20). *Fake it Till You Make it: Reputation, Competition, and Yelp Review Fraud*. Retrieved from Management Science 62: http://nrs.harvard.edu/urn-3:HUL.InstRepos:22836596

Mallet. (2018). Topic Modeling. Retrieved from http://mallet.cs.umass.edu/topics.php PyTorch. (2017). *Text Classification with TorchText.* Retrieved from PyTorch:

https://pytorch.org/tutorials/beginner/text_sentiment_ngrams_tutorial.html

Rafferty. (2018). LDA on the Texts of Harry Potter. Retrieved from

https://towardsdatascience.com/basic-nlp-on-the-texts-of-harry-potter-topic-modeling-with-latent-dirichlet-allocation-f3c00f77b0f5

Torch Contributors Revision. (2017). *Torchtext.* Retrieved from PyTorch: https://torchtext.readthedocs.io/en/latest/#
Yelp Dataset Challenge. (2019). Retrieved from: https://www.yelp.com/dataset/challenge.

## Appendix Figures:

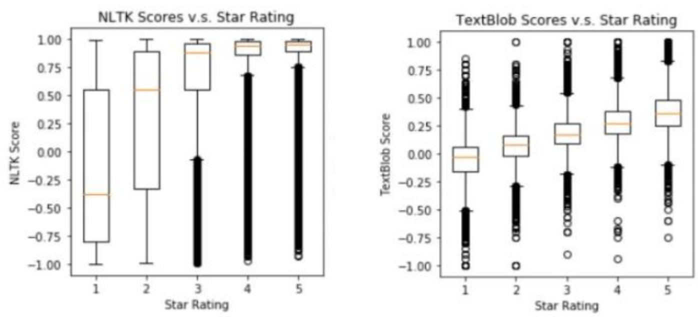



Fig 1. Vadar & TextBlob Scores vs Star Rating

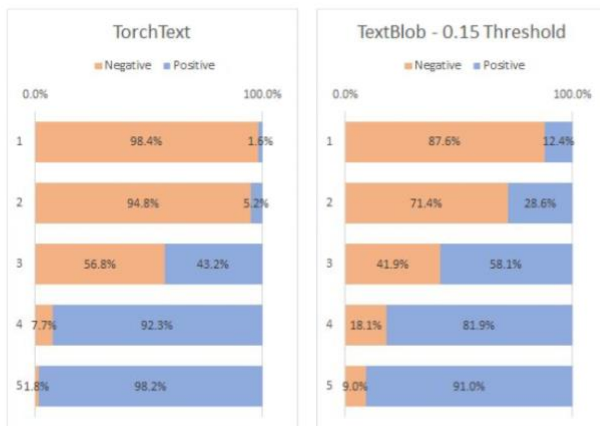



Fig 2. TorchText & TextBlob Results vs Star Rating

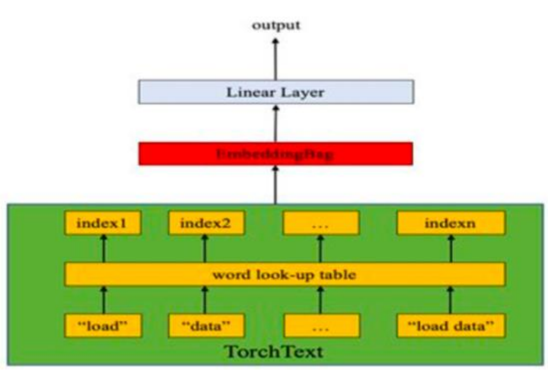



Fig 3. Classification Model [Source: PyTorch Tutorial - Text Classification with TorchText]

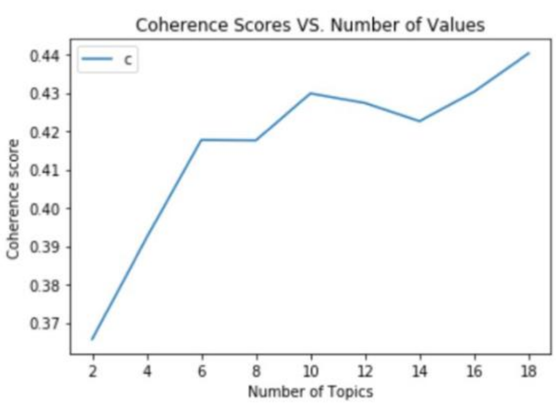

Fig 4. Comparison of Dishonest Reviews Versus Total Reviews

| Topic 0: | Topic 1: | Topic 2: | Topic 3: | Topic 4: |
|---|---|---|---|---|
| food | chicken | pizza | chicken | food |
| good | good | crust | order | chicken |
| place | sauce | cheese | minutes | service |
| service | rice | toppings | ordered | pizza |
| like | fried | sauce | said | wait |

Fig 5.Top 5 topics LSA using sklearn

```
Total Number of Documents: 7042
[(0, '0.325*"food" + 0.199*"place" + 0.194*"time" + 0.193*"like" + 0.187*"good" + 0.179*"one" + 0.176*"u" + 0.158*"would" + 0.1
57*"get" + 0.152*"order"'), (1, '0.370*"u" + -0.283*"food" + -0.271*"good" + -0.264*"place" + 0.226*"table" + -0.200*"like" +
0.186*"minute" + -0.176*"chicken" + 0.164*"order" + 0.156*"asked"'), (2, '-0.802*"food" + 0.216*"pizza" + 0.187*"like" + 0.149
*"one" + 0.126*"place" + -0.112*"u" + 0.110*"would" + -0.092*"service" + -0.084*"restaurant" + -0.083*"table"'), (3, '-0.454*"d
e" + -0.424*"le" + -0.239*"et" + -0.216*"la" + -0.178*"table" + -0.174*"un" + -0.153*"est" + -0.149*"l" + -0.145*"je" + 0.138
*"time"'), (4, '0.281*"time" + 0.262*"place" + -0.233*"chicken" + -0.201*"ordered" + -0.200*"u" + 0.189*"get" + 0.184*"de" + -
0.182*"came" + 0.177*"le" + 0.167*"service"')]
```

Fig 6.Top 5 Topics LSA using Gensim



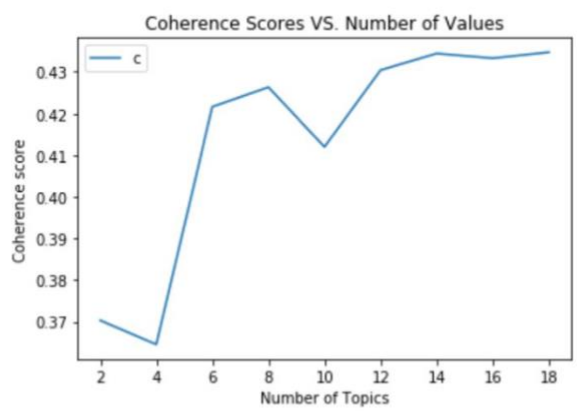




Fig 7. (left) Coherence Scores and Number of Topics using Standard LDA

Fig 8. (right) Coherence Scores and Number of Topics using LDA Mallet

```
[(0,
  '0.037*"sushi" + 0.029*"dish" + 0.025*"roll" + 0.024*"noodle" + 0.021*"soup" '
  '+ 0.018*"chinese" + 0.017*"rice" + 0.014*"pho" + 0.013*"thai" + '
  '0.013*"raman"'),
 (1,
  '0.164*"pizza" + 0.046*"wing" + 0.025*"italian" + 0.022*"pasta" + '
  '0.021*"cheese" + 0.021*"crust" + 0.020*"slice" + 0.019*"delivery" + '
  '0.015*"pie" + 0.014*"sauce"'),
 (2,
  '0.023*"table" + 0.015*"server" + 0.014*"ask" + 0.008*"tell" + 0.008*"seat" '
  '+ 0.007*"see" + 0.007*"leave" + 0.007*"sit" + 0.006*"customer" + '
  '0.006*"take"'),
 (3,
  '0.018*"salad" + 0.017*"dish" + 0.016*"dessert" + 0.015*"order" + '
  '0.013*"steak" + 0.011*"taco" + 0.010*"delicious" + 0.008*"appetizer" + '
  '0.008*"plate" + 0.008*"entree"'),
 (4,
  '0.042*"chicken" + 0.029*"fry" + 0.025*"meat" + 0.025*"sauce" + 0.017*"side" '
  '+ 0.016*"flavor" + 0.016*"sandwich" + 0.013*"spicy" + 0.012*"taste" + '
  '0.008*"cheese"'),
 (5,
  '0.027*"go" + 0.026*"order" + 0.017*"place" + 0.017*"come" + 0.016*"get" + '
  '0.015*"food" + 0.014*"time" + 0.014*"eat" + 0.012*"really" + 0.012*"good"'),
 (6,
  '0.056*"breakfast" + 0.042*"coffee" + 0.021*"egg" + 0.020*"brunch" + '
  '0.016*"pancake" + 0.014*"cafe" + 0.013*"sandwich" + 0.012*"waffle" + '
  '0.012*"crepe" + 0.010*"sweet"'),
 (7,
  '0.060*"good" + 0.040*"food" + 0.028*"place" + 0.027*"service" + '
  '0.026*"great" + 0.018*"nice" + 0.015*"pretty" + 0.013*"drink" + '
  '0.012*"menu" + 0.012*"price"'),
 (8,
  '0.010*"store" + 0.008*"shop" + 0.006*"find" + 0.005*"old" + 0.005*"park" + '
  '0.005*"mall" + 0.005*"space" + 0.004*"street" + 0.004*"locate" + '
  '0.004*"wall"'),
 (9,
  '0.006*"pour" + 0.005*"est" + 0.005*"une" + 0.005*"que" + 0.005*"service" + '
  '0.004*"tre" + 0.004*"etait" + 0.004*"qui" + 0.004*"mais" + 0.004*"les"')]
```

Fig 9. Topic Modeling Results of Standard LDA (Topic Number = 10)

```
[(0,
  [('good', 0.14982486682218585),
   ('food', 0.13041746457296055),
   ('place', 0.09430033773197312),
   ('price', 0.02790501723477595),
   ('eat', 0.02488840917795341),
   ('service', 0.02426656453466105),
   ('pretty', 0.020924062532641622),
   ('nice', 0.014555899864210856),
   ('quality', 0.014496709724591762),
   ('bad', 0.011293478639323144)]),
 (1,
  [('salad', 0.021166196497910197),
   ('dessert', 0.013852328447731176),
   ('meal', 0.013524841818618683),
   ('side', 0.01225998404095956),
   ('steak', 0.011477457113786772),
   ('serve', 0.01042855066402067),
   ('bread', 0.010044703111419613),
   ('menu', 0.01002571837929715),
   ('top', 0.00894299537543791),
   ('dish', 0.008719924772998964)]),
 (2,
  [('order', 0.043345728979181736),
   ('chicken', 0.02943352087334848),
   ('taste', 0.021941363369953684),
   ('dish', 0.017523250492143025),
   ('sauce', 0.016080344434015063),
   ('sushi', 0.015845525952478533),
   ('meat', 0.01582034434587944),
   ('spicy', 0.011994628763312414),
   ('fresh', 0.011606202481521423),
   ('eat', 0.010677630738179912)]),
 (3,
  [('drink', 0.030297139010777473),
   ('burger', 0.025271094090620515),
   ('fry', 0.02110365189810482),
   ('bar', 0.02085888319178393),
   ('beer', 0.01260240128299597),
   ('place', 0.012505513670077281),
   ('friend', 0.012383129316916833),
   ('night', 0.00924384317465012),
   ('menu', 0.009152692328285828),
   ('people', 0.008249470305221477)]),
 (4,
  [('place', 0.048581710414902626),
   ('love', 0.037667059523537084),
   ('pizza', 0.03466771174222487),
   ('good', 0.029662362404741745),
   ('sandwich', 0.02184789573654942),
   ('delicious', 0.019682415268782755),
   ('breakfast', 0.01675458269446415),
   ('fresh', 0.014824387257706479),
   ('lunch', 0.014570508272879146),
   ('favorite', 0.012369985353685607)]),
 (5,
  [('great', 0.11696477071692221),
   ('service', 0.06527805861660087),
   ('food', 0.0581114432641777),
   ('amazing', 0.033935741415667214),
   ('staff', 0.020710757461642897),
   ('excellent', 0.02053391707101586),
   ('delicious', 0.020209827112097786),
   ('experience', 0.018677445241126453),
   ('love', 0.018262469185033527),
   ('nice', 0.017073904487870932)]),
 (6,
  [('order', 0.05561118157487442),
   ('table', 0.02949271677850849),
   ('wait', 0.022937538064400777),
   ('time', 0.01700479960664458),
   ('server', 0.014823464991926157),
   ('leave', 0.013449514731272314),
   ('waitress', 0.013321562470316751),
   ('give', 0.011821894922697838),
   ('back', 0.011584987679705878),
   ('bring', 0.01041609628267951)]),
 (7,
  [('location', 0.01926511301290277),
   ('eat', 0.01298107526447141),
   ('time', 0.01293750189449434),
   ('find', 0.01268111365955684),
   ('make', 0.012150655242444757),
   ('taco', 0.01152294611552879),
   ('give', 0.01135559911489224),
   ('review', 0.010686211112346041),
   ('work', 0.010304786250517829),
   ('thing', 0.010031347566458862)])]
```

Fig 10. Topic Modeling Results of LDA Mallet (Topic Number = 8) on Honest Reviews

## Tables:

Table 3. Examples of Dishonest Reviews

| Review | Rating | Polarity Predicted |
|---|---|---|
| Love the crust on this pizza. The sauce is decent, cheese is okay, the pepperoni was not the best quality. My friend ordered an everything slice, said it was good, just wished so cheese was on the toppings to hold it together. | 4 | Negative |
| We love the all you can eat king crab legs on Tuesdays. Service is good. The food is good. I enjoy the quick soup and salad for lunch. | 3 | Positive |

Table 4. Examples of Honest Reviews

| Review | Rating | Polarity Predicted |
|---|---|---|
| This place has gone down hill. Clearly they have cut back on staff and food quality. Many of the reviews were written before the menu changed. I've been going for years and the food quality has gone down hill.The service is slow & my salad, which was $15, was as bad as it gets.It's just not worth spending the money on this place when there are so many other options. | 1 | Negative |
| You can't really find anything wrong with this place, the pastas and pizzas are both amazing and high quality, the price is very reasonable, the owner and the staff are very friendly, if you're in downtown check this place out, a lot of people think just because it's downtown there are lots of options around but that's not always the case as there is also a lot of poor quality food in downtown as well. | 5 | Positive |

Table 6. Number of Records in Training, Test and Validation sets

| Dataset | Number of records |
|---|---|
| Training Set | 399,040 |
| Testing Set | 213,772 |
| Validation Set | 99,760 |