Cafés in Portland

Listen to the customer's voice to improve performance

06.22.2021

—

Valentina Rizzati
Data Scientist
Yelp, Business Team

Opportunity

I am a Data Scientist for the Business team at Yelp. I've recently been approached by a Product Manager who intends to build a new tool for business owners.

In particular, I will focus on cafés because their relative low variance in pricing and customer preferences across regions will likely make the insights more applicable at a national scale. I will build the prototype of a tool that will enable cafés owners to gather useful information from reviews and optimize their performance.

Impact Hypothesis

If cafés owners have more visibility on what are the most valued elements in a café by a customer (e.g. decor, matcha latte), they will be able to invest in what really matters and align to customers' expectations. As a result, their performance and popularity will eventually improve.

Data

I have collected data from Yelp, which was in the format of five json files (*business*, *review*, *user*, *tip*, and *checkin*). I used MongoDB to store the data locally and then reshape it in a tabular form in Python. I focused on the *business* and *review* tables.

It's important to note that the Yelp dataset provides data only on a subset of cities in the US. By filtering for reviews for the café-related categories of interest (i.e. *Cafes*, *Coffee & Tea*, *Coffee Roasteries*), I noticed that ~16% of reviews were about coffee establishments in **Portland**. This is not surprising, knowing the strong coffee culture that is so pervasive in this city. Therefore, the fact that reviews on Portland cafés alone represent such a significant size of the available cafés reviews on Yelp and the fact that Portland is definitely a representative market for coffee, enhanced my confidence in the generalizability potential of using Portland as the test market for my prototype model.

Algorithm

DATA EXTRACTION & CLEANING

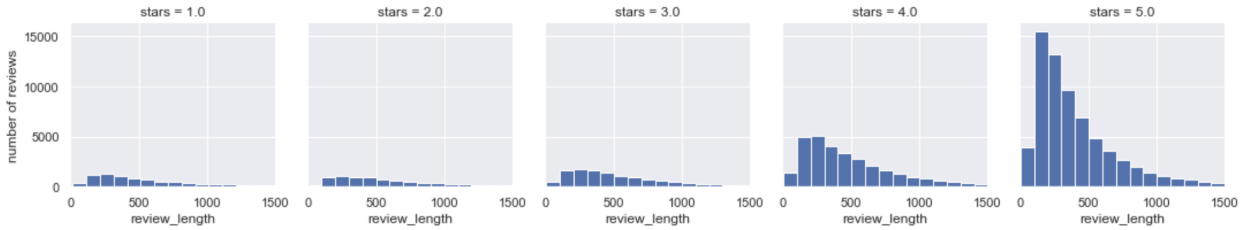
I started by building a new instance of MongoDB, called *yelp*. I then included 5 collections, corresponding to the 5 json files included as part of the Yelp dataset.

Then, I created two pandas dataframes corresponding to respectively the *business* and the *reviews* collections. Finally, I extracted data on the three cafe-related categories (i.e. *Cafes*, *Coffee & Tea*, *Coffee Roasteries*) and for Portland as the geo of choice. My final dataframes were then ready for use in the modeling phase.

EDA

As part of the EDA phase, I analyzed the distribution of stars across reviews as well as the distribution of reviews' length by stars number. Interestingly, 4 and 5 stars (i.e. positive sentiment for the purpose of this analysis) make up nearly 78% of the Portland cafe reviews.

Also, as we can see from the image below, 1, 2 and 3 stars reviews are characterized by similar distributions of review length. Differently, 4 and 5 stars reviews have a more right-skewed distributions of review length.



Finally, as shown in the output below, average review length is higher the lower is the number of stars. This could be due to the fact that, behaviorally, users tend to share more content when they are dissatisfied with a service.

```
Average review length for 1 stars reviews: 636.91
Average review length for 2 stars reviews: 651.68
Average review length for 3 stars reviews: 595.45
Average review length for 4 stars reviews: 526.35
Average review length for 5 stars reviews: 436.12

Average review length for all reviews: 499.81
```

PREPROCESSING

In this step, I followed a pretty standard pipeline:

- Data cleaning to remove numbers, capital letters and punctuation
- Tokenization
- Lemmatization
- Parts of speech by sentiment with:
    - Positive sentiment: 1 or 2 stars

    - Neutral sentiment: 3 stars
    - Negative sentiment: 4 or 5 stars
- Dependency parsing to understand which adjectives and nouns were mostly used in correspondence of core terms like *coffee* or *service*

UNSUPERVISED LEARNING | TOPIC MODELING

Topic modeling constituted the core of the unsupervised learning modeling phase.

First, I created a class, called *Tokenizer_pos_sel_lemma()*, that allowed me to both lemmatize with WordNetLemmatizer and filter the corpus for only adjectives and nouns which, because of their highly descriptive nature, were the tokens I was primarily interested in for my analysis.

I then iterated through various combinations of TF-IDF and NMF by testing different *n_components*, *max_df* and *min_df*. Then, I finalized the topic modeling phase with a NMF model with 9 components. Since I was satisfied with my result, I did not see the need to continue testing with more complex topic modelers.

My 9 final topics are:

- Savoury food
- Doughnut/Voodoo (topic related to landmark doughnut and coffee shop Voodoo)
- Coffee/Barista
- Tea
- Ice Cream
- Books/Powell's (topic related to landmark bookstore and coffee shop Powell's)
- Location/Atmosphere:
- Service
- Pastries/Dessert

SUPERVISED LEARNING | CLASSIFICATION

In this phase I am planning to build a classification model to predict the customer sentiment (positive or negative) based on the strengths of the 9 topics identified as part of the topic modeling phase.

Since interpretability is my main objective, I will likely choose Logistic Regression and F1 to be respectively my model and metric of choice.

VISUALIZATIONS

For the topic modeling component, I have generated wordclouds for both positive and negative reviews (see below).

Positive Sentiment

Negative Sentiment

I am also working on a scattertext visualization.

Tools

To store and query the data I used MongoDB.

To handle the text data I used NLTK and spaCy.

To manipulate the data and build the classification model I used pandas, numpy, and scikit-learn.

For visualizations I used scattertext, stylecloud, seaborn, and, depending on the need, plotly.

Next Steps

To finalize the project, I will build the classification model based on the topics generated as part of the unsupervised learning phase.

I will also refine a few visualizations like scattertext.