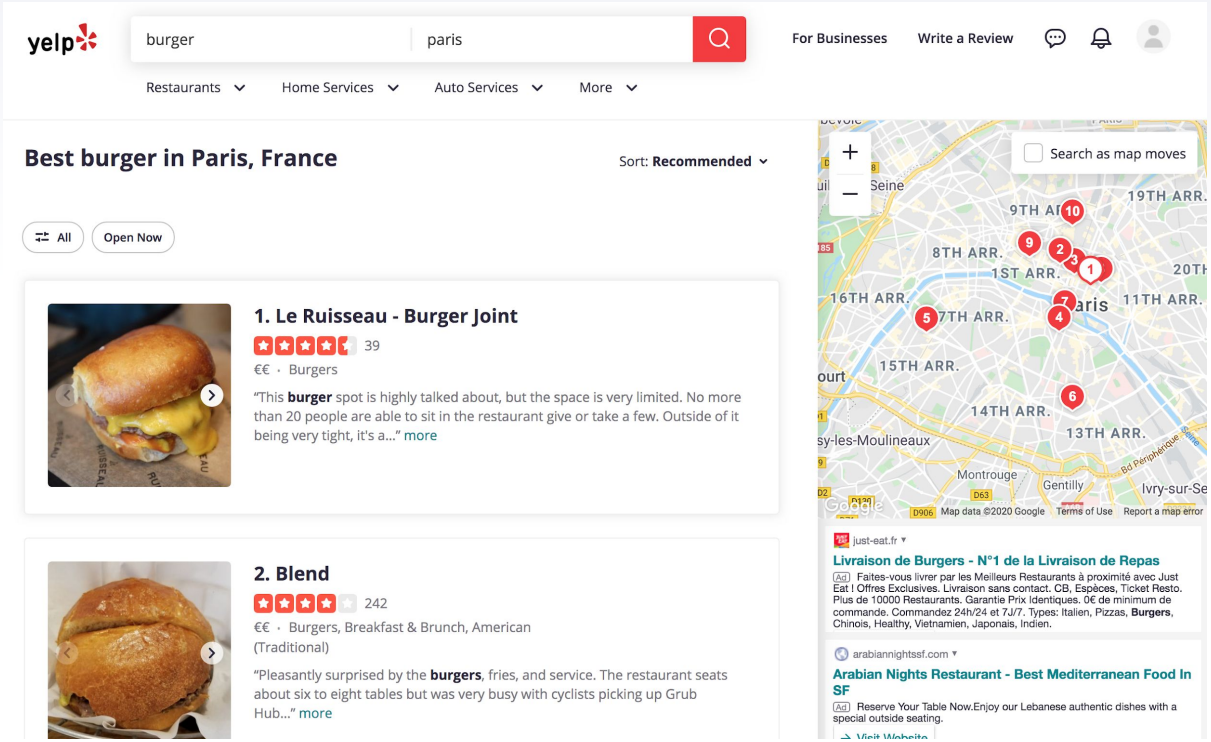
Yelp for Restauranteurs

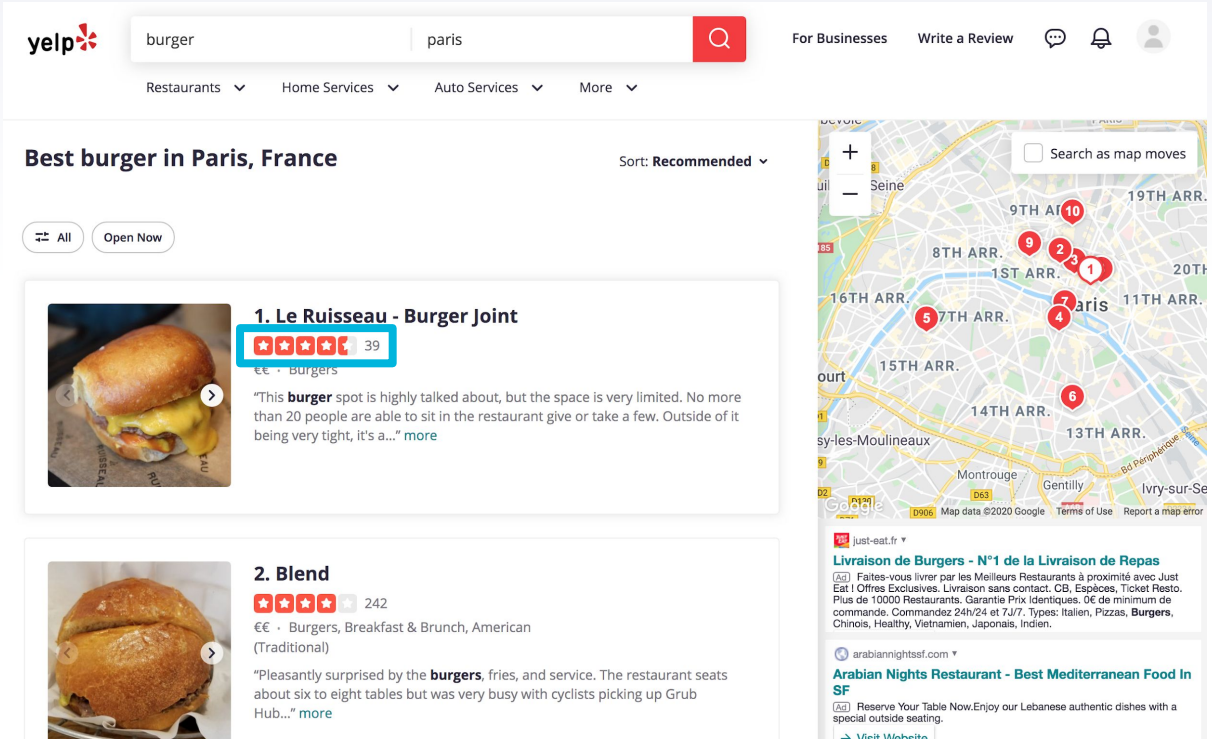
Mirae Kim

Currently, Yelp is primarily geared towards Restaurant Seekers and not for Restaurant Owners

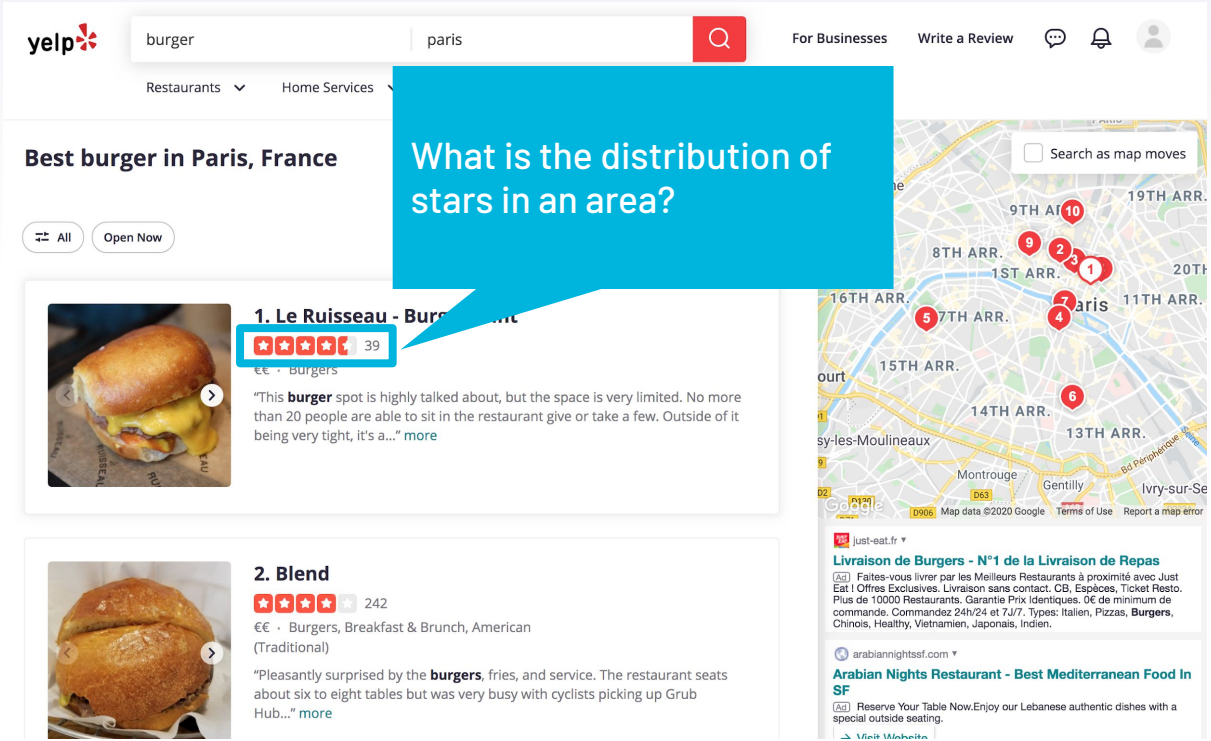
Yelp Search view has limited information



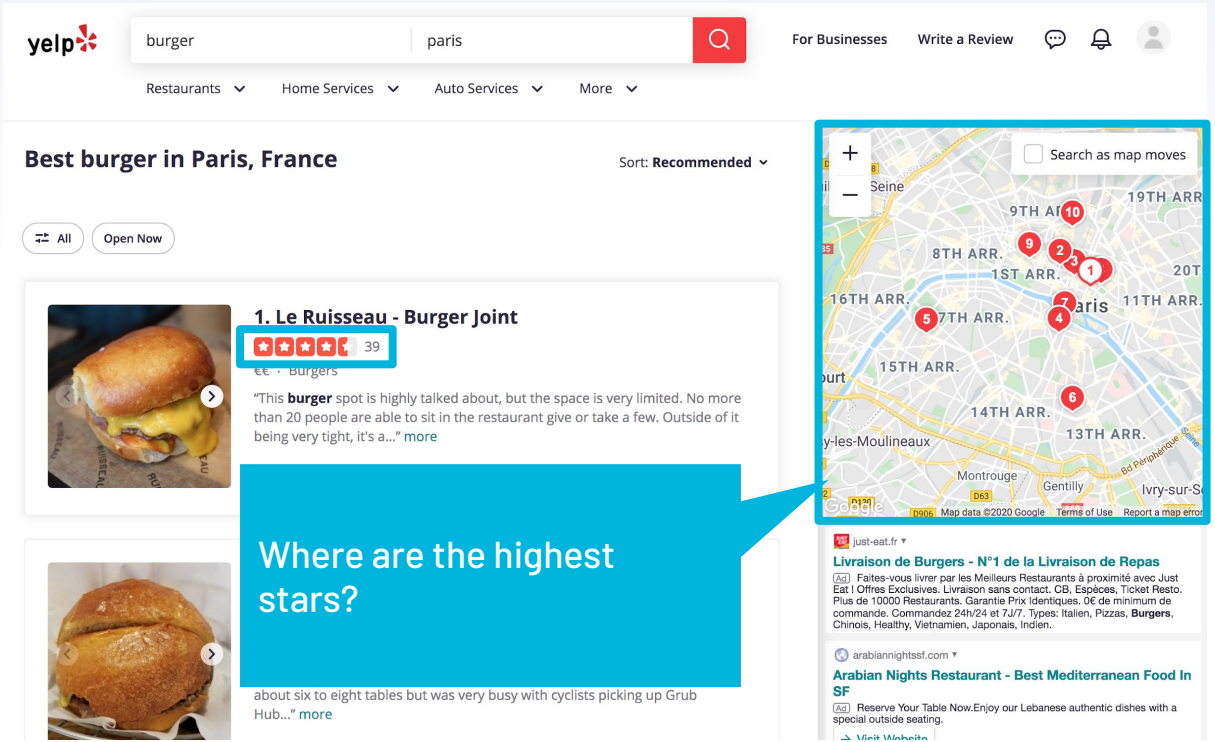
Yelp Search view has limited information



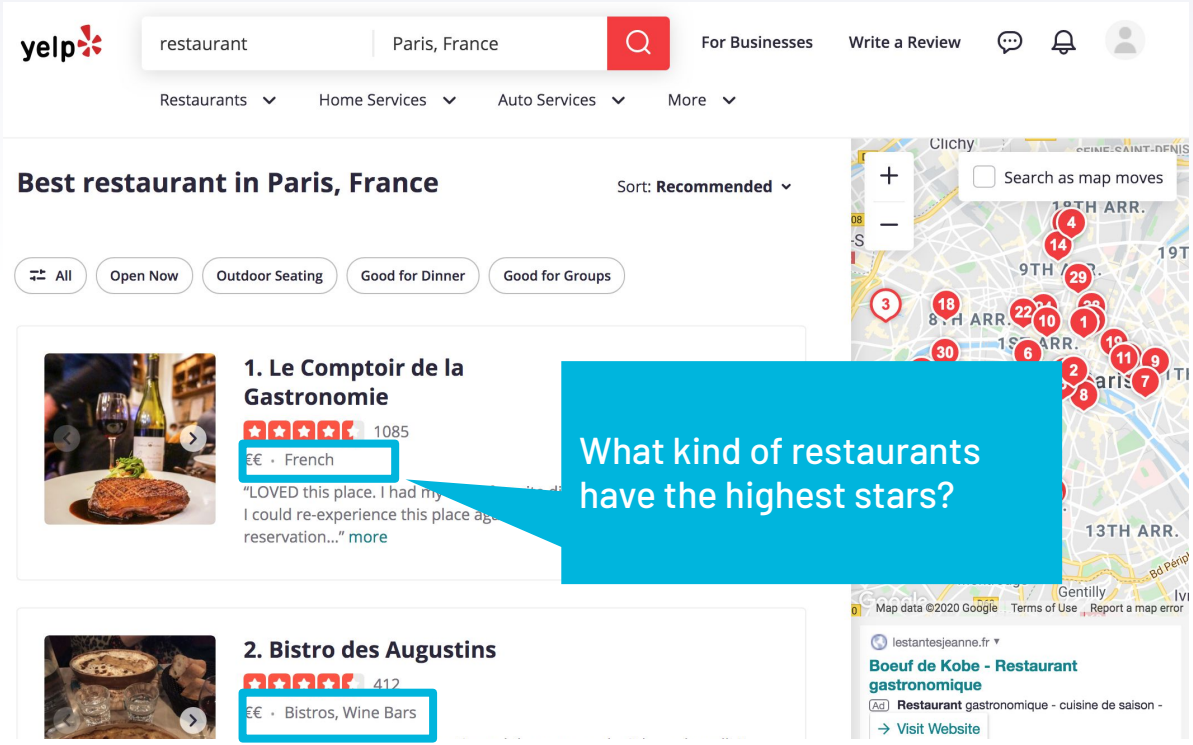
Yelp Search view has limited information



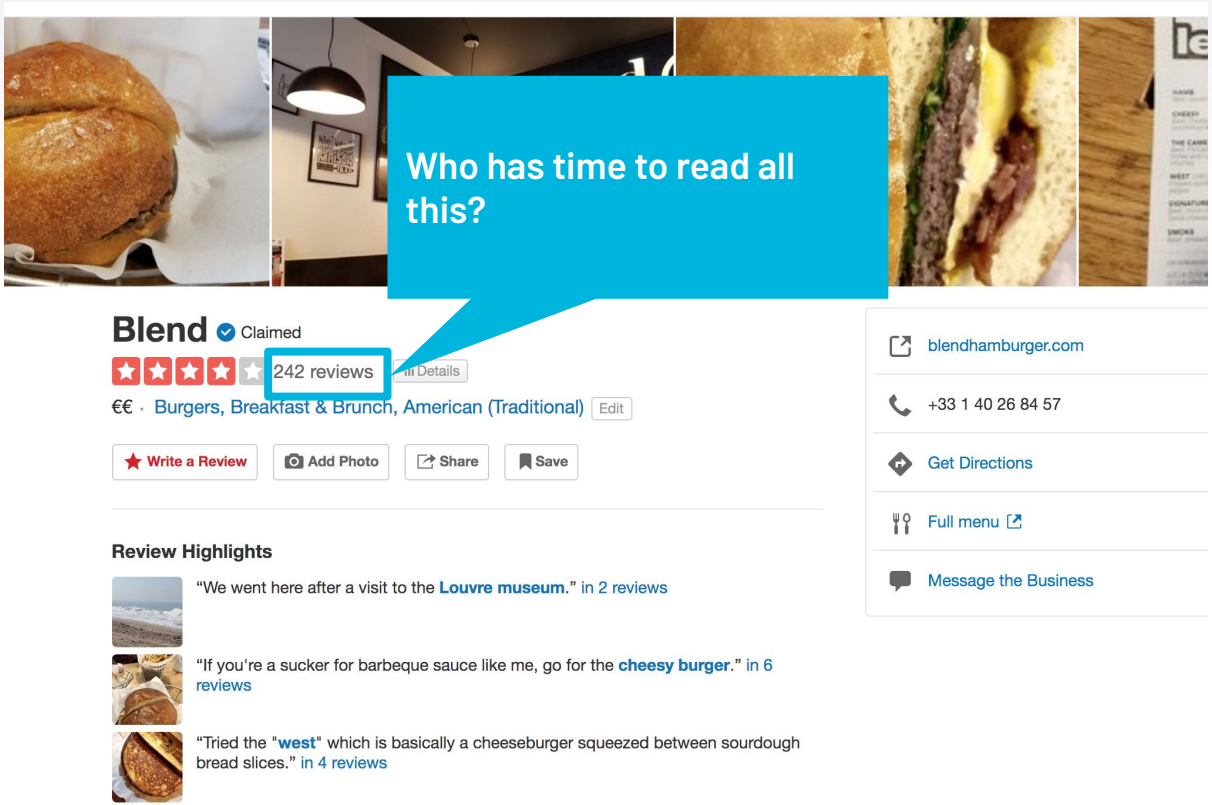
Yelp Search view has limited information



Yelp Search view has limited information



No holistic view of Restaurant Reviews



Project Objectives

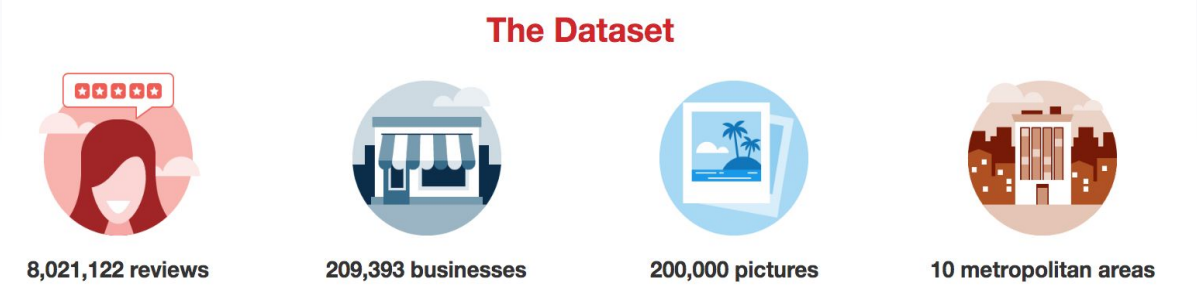
- Create a view for **potential** Restaurant Owners
  - Optimize Location
  - Optimize Category
  - Optimize Characteristics
- Create a view for **current** Restaurant Owners
  - Understand Reviews
  - Optimize Characteristics

Personal Objectives

- Familiarize with **Natural Language Processing** (NLP)
- Learn how to develop an **interface** from scratch
  - Data Storage and Serving
  - Model Storage and Serving
  - User Interface Creation

Data Sources

- **Yelp Reviews Dataset:** Downloadable Dataset with large corpus of text



- **Yelp Fusion API**: Extra information about restaurants

Yelp Reviews Dataset: https://www.yelp.com/dataset

Yelp Fusion API: https://www.yelp.com/fusion

Use Latent Dirichlet Allocation to summarize the texts by Topics

Goal: Finds latent topics in a corpus of text

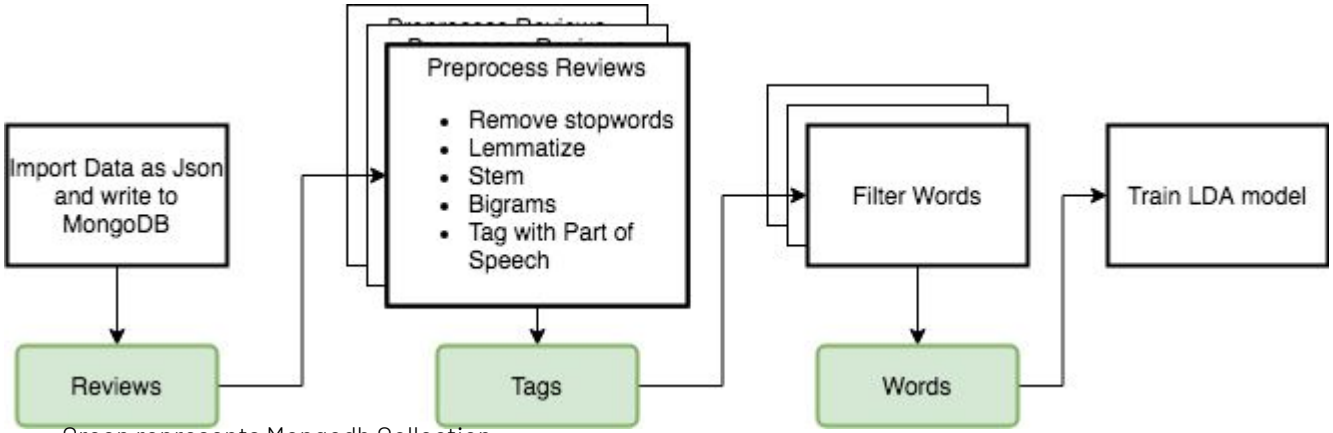
Main assumption: Text documents are a mix of topics and each topic is a mix of words

Technologies:

- Gensim
- NLTK

Data Processing

- Failed to use spark implementation of LDA due to data engineering issues
- Multiprocessing python library used to create threads of concurrent processing
- MongoDB NoSql database used to store intermediary results



Green represents Mongodb Collection

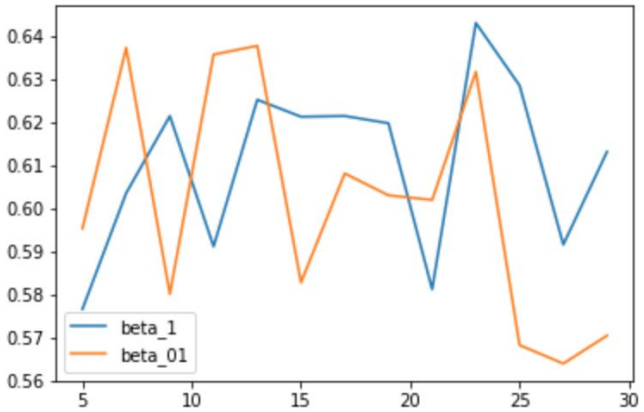
Finetuning and Results

Main Hyperparameters:

- **Alpha:** document-topic density
- **Beta:** topic-word density
- **Number of Topics:** Number of topics LDA should produce

Chosen Hyperparameters:

- **Alpha:** "Asymmetric"
- **Beta:** 0.1
- **Number of Topics:** 13



Evaluation Metric:

- **Coherence:** measures the degree of semantic similarity among the highest scoring words in each topic

Final Coherence Score: 0.6377

Topics produced

```
#0: (0, '0.034*"place" + 0.018*"coffee" + 0.018*"breakfast" + 0.016*"food" + 0.015*"time" +
0.012*"staff" + 0.012*"service" + 0.010*"day" + 0.008*"love" + 0.008*"spot"')

#1: (1, '0.061*"food" + 0.049*"service" + 0.032*"place" + 0.022*"restaurant" + 0.022*"time" +
0.014*"experience" + 0.013*"dinner" + 0.012*"staff" + 0.010*"menu" + 0.010*"quality"')

#2: (2, '0.017*"salad" + 0.013*"sauce" + 0.011*"meal" + 0.011*"bread" + 0.010*"chicken" +
0.010*"menu" + 0.010*"dinner" + 0.009*"restaurant" + 0.009*"side" + 0.008*"steak"')

#3: (3, '0.045*"bar" + 0.037*"place" + 0.032*"beer" + 0.027*"drinks" + 0.024*"night" +
0.023*"food" + 0.021*"music" + 0.017*"selection" + 0.015*"drink" + 0.014*"patio"')

#4: (4, '0.063*"food" + 0.046*"place" + 0.028*"service" + 0.022*"chicken" + 0.017*"staff" +
0.016*"tacos" + 0.015*"chips" + 0.014*"lunch" + 0.014*"time" + 0.011*"love"')

#5: (5, '0.062*"cream" + 0.049*"ice" + 0.039*"chocolate" + 0.018*"cake" + 0.015*"dessert" +
0.014*"flavors" + 0.013*"la" + 0.011*"vanilla" + 0.010*"cookies" + 0.010*"desserts"')

#6: (6, '0.030*"selection" + 0.030*"quality" + 0.025*"price" + 0.022*"prices" + 0.013*"variety" +
0.013*"foods" + 0.012*"place" + 0.010*"options" + 0.010*"wall" + 0.010*"items"')

#7: (7, '0.068*"fries" + 0.057*"burger" + 0.031*"burgers" + 0.021*"sandwich" + 0.019*"bbq" +
0.017*"meat" + 0.015*"ribs" + 0.013*"bun" + 0.012*"cheese" + 0.011*"beef"')

#8: (8, '0.025*"order" + 0.024*"time" + 0.019*"minutes" + 0.016*"service" + 0.015*"people" +
0.014*"food" + 0.013*"customer" + 0.011*"place" + 0.010*"manager" + 0.009*"location"')

#9: (9, '0.049*"vegas" + 0.027*"strip" + 0.023*"buffet" + 0.023*"hotel" + 0.017*"las" +
0.015*"room" + 0.011*"casino" + 0.011*"club" + 0.011*"night" + 0.010*"floor"')

#10: (10, '0.108*"pizza" + 0.029*"toppings" + 0.028*"store" + 0.027*"crust" + 0.020*"slice" +
0.015*"wings" + 0.014*"pizzas" + 0.013*"place" + 0.012*"delivery" + 0.012*"shopping"')

#11: (11, '0.033*"rice" + 0.025*"chicken" + 0.023*"soup" + 0.021*"spicy" + 0.018*"beef" +
0.017*"pork" + 0.017*"noodles" + 0.016*"sauce" + 0.015*"food" + 0.014*"thai"')

#12: (12, '0.020*"watch" + 0.018*"tables" + 0.017*"room" + 0.015*"tv" + 0.015*"game" +
0.015*"area" + 0.015*"bar" + 0.014*"games" + 0.013*"chairs" + 0.012*"sports"')
```

Topics produced

Topic0  **Breakfast**

Topic1  **Dinner**

Topic2  **Main Dish**

Topic3  **Bar**

Topic4  **Mexican**

Topic5  **Dessert**

Topic6  **Value**

Topic7  **Burgers**

Topic8  **Service/Wait Time**

Topic9  **Vegas**

Topic10  **Pizza**

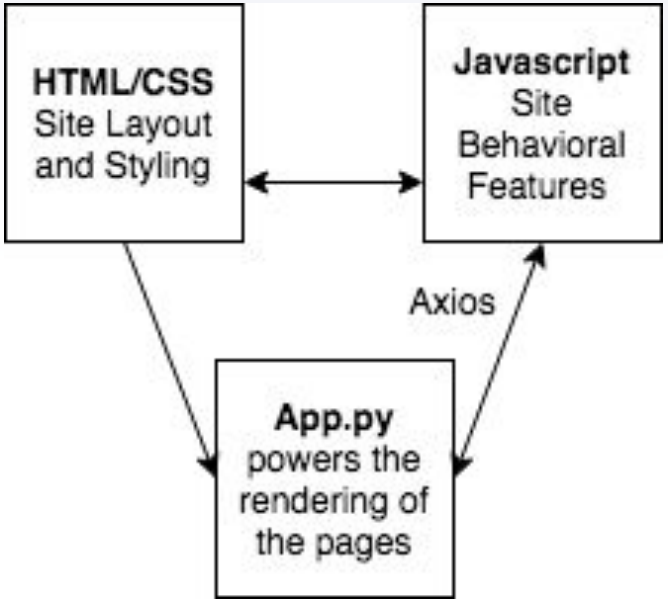
Topic11  **Asian**

Topic12  **Sports Bar**

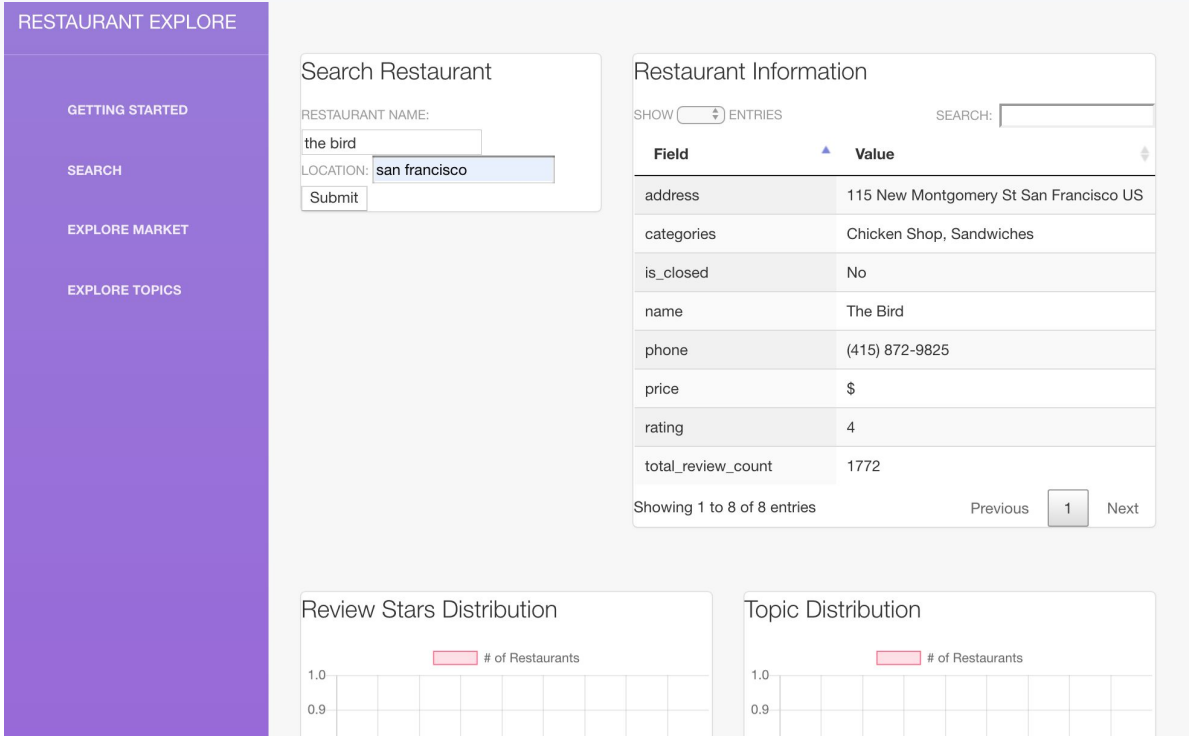
Create a User Interface to showcase results

Technology:

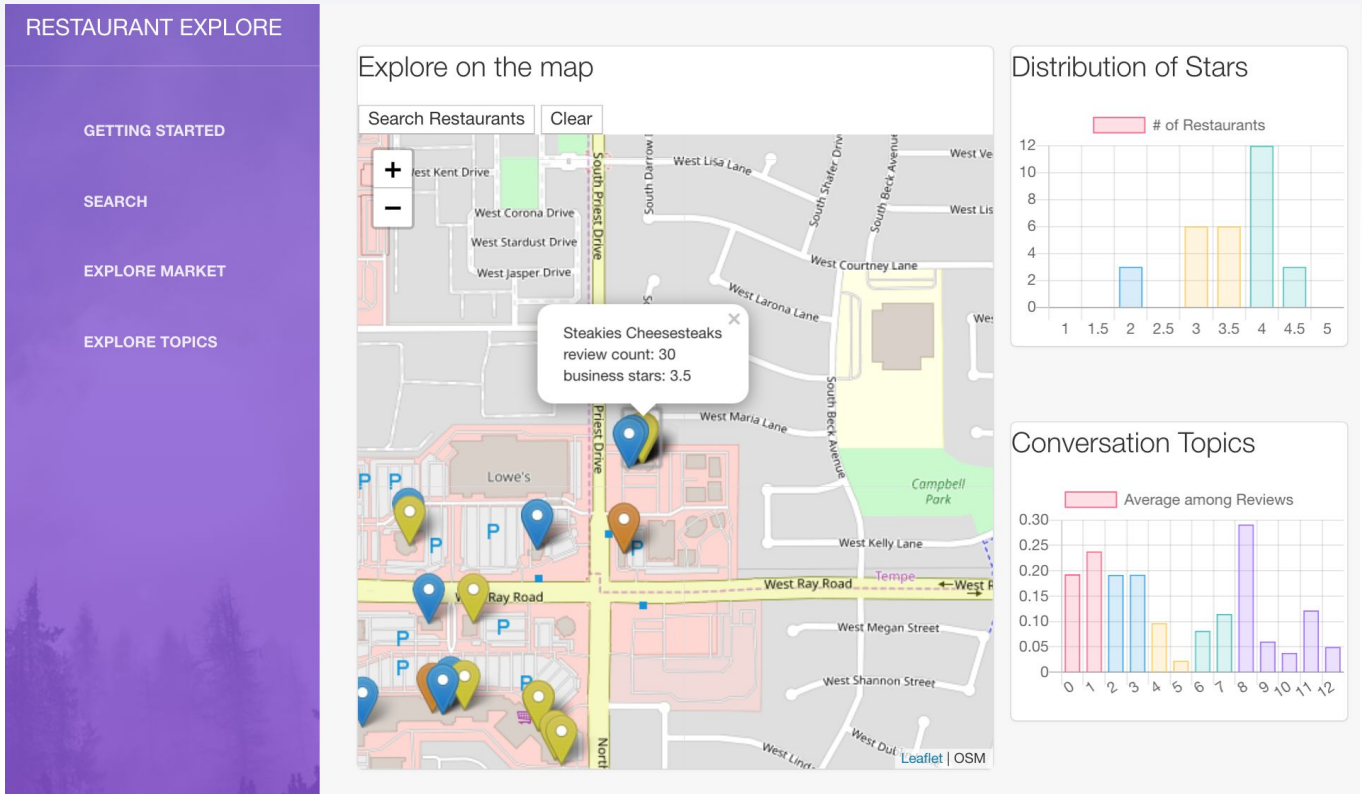
- Flask: Framework for developing web applications
- Leaflet: Map API
- Bootstrap: HTML/CSS framework
- MongoDB: Database



Search restaurant for in depth analysis on your review content



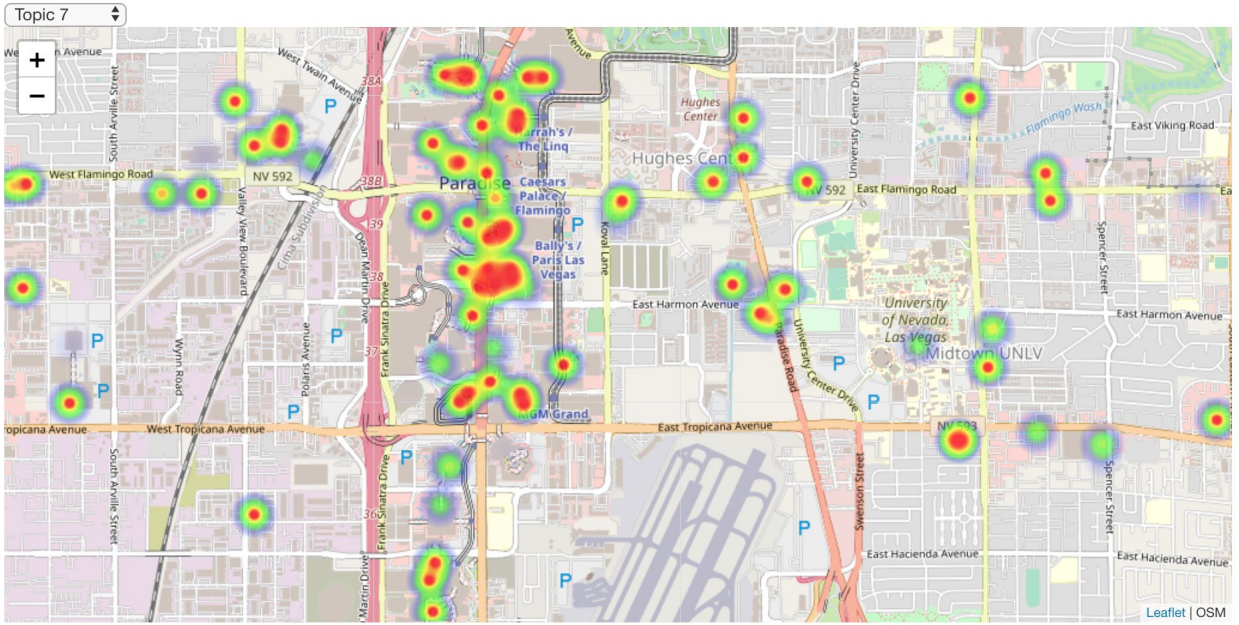
Explore the review stars on a map



Explore Topics on a map

RESTAURANT EXPLORE

- GETTING STARTED
- SEARCH
- EXPLORE MARKET
- EXPLORE TOPICS



Next steps

Data

- Make a model for each city or for each category to have more precise topics

Application

- Database Optimization
  - Slow Record Retrievals
- UX/UI Design
  - More intuitive UX
- Page that analyzes per category and topic