[야놀자] DI팀 채용과제

Office Location Picker Project

by Bolat Ashim
(May 1, 2019)

Overview

---

1. Idea & Background
2. Target User Group
3. Approach
    a. Software
    b. Data Analysis
    c. Application Demo
4. Discussion / Conclusion
5. Running the application
6. References

IDEA

- Socializing among workmates outside work is significant to the spirit of the employees and consequently success of a company
- Lunch breaks & Dinners are a great way to promote socializing and office location is important for maintaining
- Travellers look for spots tagged as “Great Location” that is oftentimes associated with places that have a great variety of high-quality food

Best Restaurants in San Francisco, CA

Showing 1-30 of 4285

All Filters | $ | $$ | $$$ | $$$$ | Open Now | Delivery | Takeout | Reservations | Cash Back

Sponsored Results

People's Bistro

151

$ · Chinese, Asian F

Offers takeout and delivery | Start Order

New Tsing Tao Restaurant

148 reviews

$$ · Chinese

Most viewed **Chinese place** in West Portal

Offers takeout and delivery | Start Order

Daeho Kalbijjim & Beef Soup

157 reviews

Korean, Soup, Noodles

Japantown

1620 Post St
San Francisco, CA 94115

View more photos

IT IS GREAT TO HAVE A LOT OF GOOD RESTAURANTS CLOSE BY

TARGET USER GROUPS

COMPANIES

Companies choosing places to rent office space

REAL ESTATE AGENCIES

Agencies looking to purchase real estate for subsequent rental or sale

APPROACH

- Analyze publicly available YELP academic datasets providing data on crowdsourced reviews and ratings of different businesses
  - Yelp_academic_dataset_business.json
  - Yelp_academic_dataset_review.json

- Make use of the data to build a data-powered web application to assist target users in finding the best locations for office lease or real estate purchase.

APPROACH - SOFTWARE

- Apache PySpark (Local)
  - Data analysis
  - Data manipulation
- Flask micro web framework
  - Mediate communication between web-interface and Spark engine
- Chart.js, KoolChart.js, Leaflet.js
  - Demonstration of maps, graphs and charts on the web

Chart.js

Leaflet

KOOLCHART

PySpark

APPROACH - DATA ANALYSIS

- Star Ratings **Distribution**
- Ratings Distribution **across regions**
- Business **Categories** Analysis
- Distribution of **Review Length**
- Analysis of features such as “diversity”, “count”, “star ratings” to **evaluate areas**
- **Extraction** of significant **review highlights** by applying **TF-IDF** on the review corpus

APPLICATION DEMO (1/3)

- **HOME PAGE :** DATA ANALYSIS
  - ANALYSIS
  - BACKGROUND
  - LOGIC
- **APPLICATION PAGE :** MAP-BASED INTERFACE
  - MAP - good spots on the map, businesses in the selected area
  - SUMMARY - business categories word-cloud in the selected area, area evaluation radar, review highlights

APPLICATION DEMO (2/3) - HOME PAGE

Business Star Ratings Analysis

Number of reviews by stars

The given bar chart refers to the numbers of stars given to different businesses by Yelp users. As can be seen, the most frequent star rating is 4.0, follwed by 3.5 and 4.5 and the rest. Based on the this information, I chose to categorize the star ratings into 3 groups shown below,

GREEN : 4.0 - 5.0 stars range for "Best" businesses

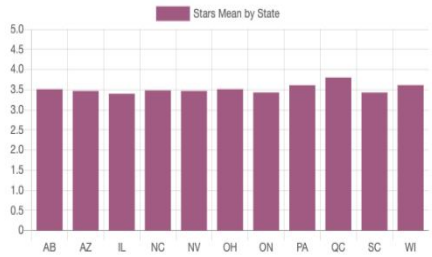
YELLOW : 3.0 - 4.0 stars range for "Good" businesses

RED : 0.0-3.0 stars range for "Unsatisfactory" businesses

As can be seen on the chart, these 3 categories would have relatively significant numbers of businesses in them.

Stars count by state

Average stars by state

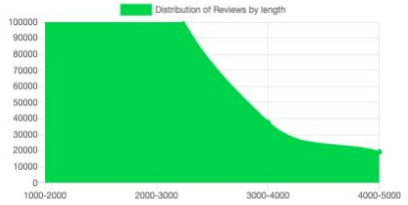


Business Categories Analysis

Frequency word cloud across all businesses

Frequency word cloud across food/drink businesses

Distribution of reviews by length



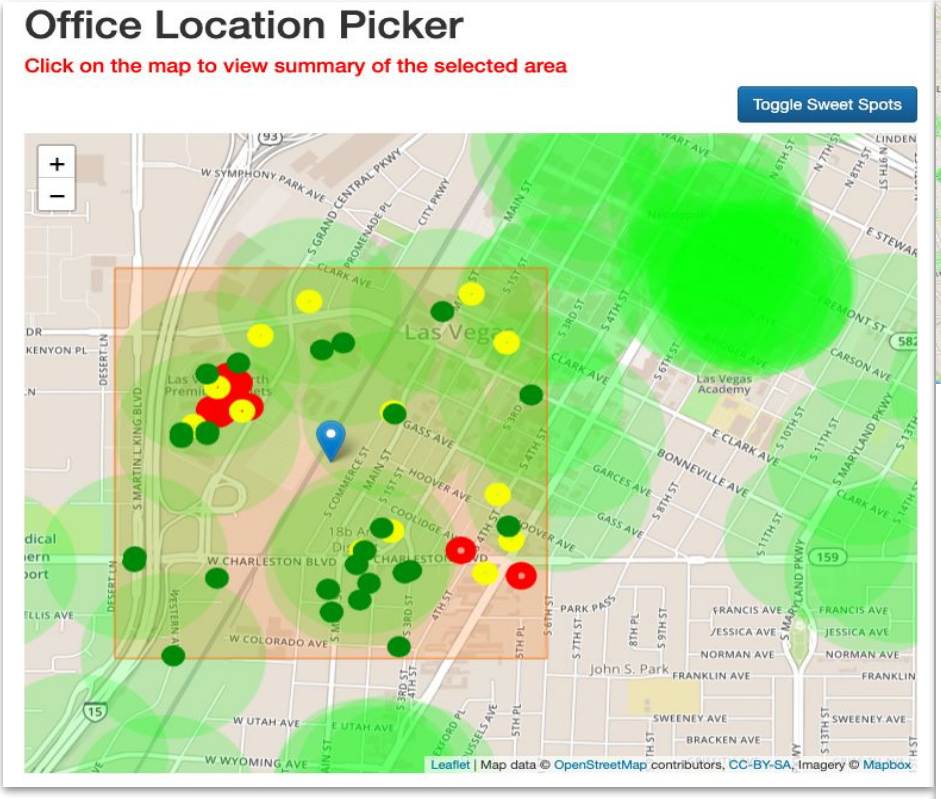
The longer a review, the more information one can retrieve out of it. As can be seen on the chart to the left, the vast majority of the reviews given are on the shorter side of the spectrum, but still, a large number of reviews (some ~50k) belong to the 3000-5000 charaters group. With this, it could be possible could gain some useful insight into the businesses.

I decided to make use of TF-IDF information retrieaval technique in my application. ML library of spark provides functionality to operate this technique. By separating each review into its corresponding sentences, tokenizing them, clearing redundant words and evaluating their rank across the corpus, I include a number of sentences from reviews for a given location selected by the user in the application.

With this, let's proceed to the demonstration of the application by clicking on the button below!

GO TO APPLICATION

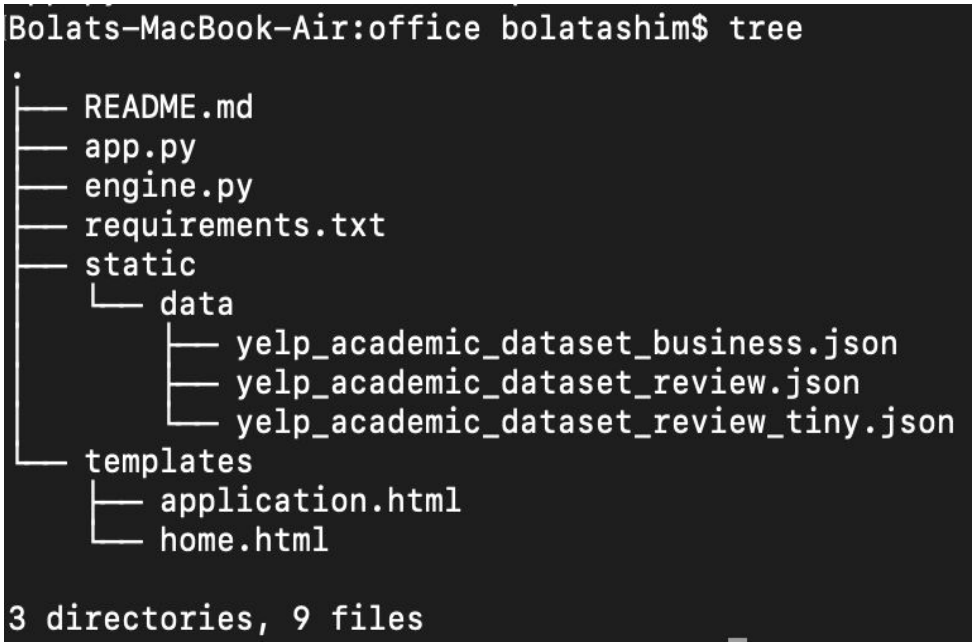
APPLICATION DEMO (3/3) - APPLICATION PAGE



DISCUSSION/CONCLUSION

- Certain parts of the application require extensive data processing, which sometimes leads to delays in the user interface. This limitation could be solved by employing more computers to utilize parallel processing power of Spark or by optimizing the application.
- The application might does not provide information on the pricing of the areas, which is a limitation that stems from the difficulty of extracting that information from the give data. AirBnB could serve as a good estimate, but it happens so that only a couple of regions that match both datasets. This challenge could be overcome by employing data from more sources.
- Overall, the application runs smoothly with only minor potential errors that could be exterminated through more thorough cleaning of data.

HOW TO RUN



- PySpark version 2.4.2
- Python3 dependencies in requirements.txt
- Javascript Libraries linked through cdn
- Datasets download links in README.md
- Launch : `python3 app.py`

NOTES

- Yelp_academic_dataset_review_tiny.json
  - This dataset (~40MB) was generated from the original yelp_academic_dataset_review.json (~6GB) dataset to optimize runtime of pyspark engine.
  - The dataset is the filtered version of the original and the code used to produce yelp_academic_dataset_review_tiny.json is provided at the bottom of the engine.py file commented out.
- Initialization Delay
  - The application was built using a MacBook Air (4GB RAM, 1.6GHz) computer, and a significant delay of ~3-7 minutes might be observed upon first initialization of the program.

THANK YOU !