

1. Web Application Description and Website Design

Please see my web application here: <http://agnedil.pythonanywhere.com/>

The function of the system is to provide recommendations to users, who pick a cuisine and a dish from that cuisine, about what restaurants the users can visit in order to try this dish. The scenario is very simple: if anyone has a certain cuisine in mind, but doesn't know where to start, they visit my website, pick preselected dishes of that cuisine and see the rating and the sentiment polarity of the Yelp reviews about those restaurants that serve this particular dish.

So far, I have 5 cuisines with 5 dishes in each cuisine, but this was done in a very short period of time, and the list can be greatly expanded if needed. First, the user visit the main page where some instructions are given about what will follow next. Then the user gets to the cuisine selection page followed by the dish selection page, after which the user sees a horizontal bar chart based on a list of the restaurant serving this dish from this cuisine. Each restaurant has 2 bars: average rating and sentiment polarity.

All data was taken from the Yelp dataset. I preprocessed it in advance collecting information about the restaurants matching the specific cuisines and specific dishes that I selected for the website. The preprocessed files are stored at the website as tsv files. The website is written using Flask. Python scripts collect user input (cuisine + dish), process the tsv file to select restaurants, and the result is displayed using D3.js scalable vector graphics. There is a scale at the bottom of each bar chart, but in addition the actual ratings and sentiment polarities are shown on each corresponding bar for convenience. There are also some elements of interactivity when you mouse over a bar or a restaurant name.

Each page has two buttons 'Main page' and 'Restaurants' which allow the user to return either to the very beginning of the website or to the page where the restaurant selection process starts.

2. Ranking Function and Sentiment Analysis

I used a special function to calculate the rating which I described in my Task 4&5 report. For ranking, the main idea was to use a balanced function in the range of [-1, 1] which is based on the star rating of each review in which a particular dish is mentioned to have been served in a particular restaurant. A completely negative rank is -1, and a completely positive rank is 1 with 0 being neutral. So, for each preselected dish I found all the reviews where it is mentioned and created a list in which each element is represented by the following data structure:

```
[restaurant,
 dish,
 [list of all star ratings from all reviews for "this restaurant + this dish"],
 [list of all sentiment polarities for all reviews for "this restaurant + this dish"]
]
```

The combination "restaurant + dish" is unique and was used as the key. Then, I used information from the third element (list of star ratings) to calculate the average rank **for each pair "restaurant + dish"** using the following formulas:

negative rating = count of **1&2-star** ratings / count of all **1,2,4,5-star rating**

positive rating = count of **4&5-star** ratings / count of all **1,2,4,5-star rating**
 (as you can see, I did not use 3-star ratings in the numerator as I consider them too neutral, sometimes even noise. Per Professor Zhai's suggestion, I also did not 3-star ratings in the denominator as they would make results less intuitive)

average rank = positive rating - negative rating

The division by the number of all review is necessary to normalize the ratings because now they do not depend on the number of reviews. Otherwise, the dishes/restaurants with more reviews would have had higher ratings. If all the reviews for this dish in this restaurant are negative, the rank is -1, and if all the reviews are positive, the rank is 1, with everything else in the middle. I found this to be a very convenient way to represent an attitude towards a particular dish cooked in a particular restaurant - as soon as the cumulative rating in reviews shifts to either negative or positive side, the rank of the pair "restaurant + dish" will go either below or above zero, i.e. it will literally become negative or positive, respectively.

Because when implementing this assignment I operated at the level of a single restaurant serving a single dish, I noticed that, for example, this function yielded the same good results both when the dish was mentioned positively in 1 review and when it was mentioned positively in 20 reviews. In order to differentiate between these cases, I introduced a minimum cutoff of at least 3 reviews for a specific dish in a specific restaurant, and I also added a third member to the formula which now looks like this:

average rank = positive rating - negative rating - 1/count of reviews

So now, if there are only 3 positive reviews, the restaurant rating will be $1 - 1/3 = 0.66$, and if there are 20 positive reviews, it will be $1 - 1/20 = 0.95$ which is a lot higher.

The sentiment polarity has the same range of $[-1, 1]$, therefore it was easy to plot both the rank and sentiment on the same chart.

Please explore my web application to get a better feel of it, and as this is the last assignment in this class, I would like to sincerely thank the Instructors and all of my fellow students for this very interesting and useful experience! Good luck to everyone!