Capstone - The Battle of the Neighborhoods Report

November 5, 2020

Table of contents

- Section 1: Introduction
- Section 2: Data
- Section 3: Methodology
- Section 4: Results
- Section 5: Discussion
- Section 6: Conclusion

1 Introduction

The target of this project is to find an optimal location for a restaurant. In detail, this report targets stakeholders interested in opening a **Chinese restaurant** in **Düsseldorf, Germany**. Since there is an abundance of restaurants in **Düsseldorf**, we are aiming to find locations that are **not already crowded with restaurants**. We are also particularly interested in areas with **no or few chinese restaurants** in the vicinity. Another preference is that the location is **as close to city center as possible**, assuming that first two conditions are met. We will use data science to generate a few most promissing neighborhoods based on these criteria. Advantages of each area will then be clearly expressed so that the best possible location can be chosen by the stakeholders.

2 Data

The data is aquired via different sources. The geojson file containing the boundaries of the 50 ditricts in Düsseldorf is downloaded from https://opendata.duesseldorf.de/sites/default/files/Stadtteile_WGS84_4326.geojson. Venue information such as type of restaurant, exact location, etc. is extracted from the **Foursquare API**. For simple geocoding tasks we use the **OpenStreetMap API** and for more complex tasks where we wish to extract each street name in a given area (poylgon) we use the **Overpass API**. In view of the problem statement, the decision on the choice of location is based on the following parameters:

- the number of restaurants in the district (any type of restaurant),
- the number of chinese restuarants in the area of interest,
- the distance to the next chinese restaurants,
- the distance between the district/street and the city center.

These parameters are therefore calculated before the exploratory analysis. The city center is chosen as *Königsstraße 1, Düsseldorf, Germany.* To calculate the distances between locations we convert the spherical cordinates (latitude/longitude) to cartesian coordinates.

3 Methodology

First, we need to figure out in which district we should locate our restaurant. For this purpose we perform an analysis that facilitates our decision making. We calculate the relevant parameters needed for the analysis. Figure 1 shows all districts of Düsseldorf and their distance to the city center chosen as *Königsstraße 1, Düsseldorf, Germany*. There are 50 districts in total. Some are located really close to the center (< 500m) and other are extremely far out (> 10km).

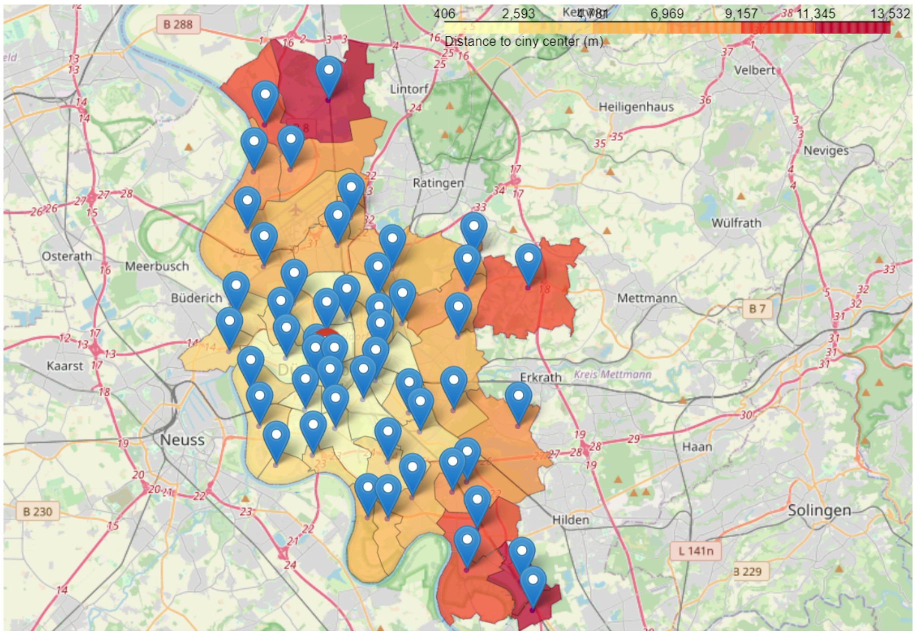

Figure 1: Map of Düsseldorf displaying the distance of each district to the city center. Blue markers are the respective district centers.

In the next step, we use the **Foursquare API** to get all venues in a district. Our interest, primarily, lies in food venues, but only those that are proper restaurants. *Bakeries, Coffee shops, Pizza parlors* or *fast food restaurants* are not direct competitors so we can neglect these. We are also going to make sure to detect all *chinese* restaurants and its subcategories. Foursquare allows for us to search for venues inside a given polygon. The polygon information is extracted from the geojson file. However, the API only accepts polygons with 4 to 15 lat/lng points. The geodata provides much more complex shapes. We therefore simplify the polygons using proper algorithms to meet the API criteria. Figure 2 shows the map of Düsseldorf with updated/simplified district boundaries. We can see that the boundaries are drawn well and should suffice for our purposes.

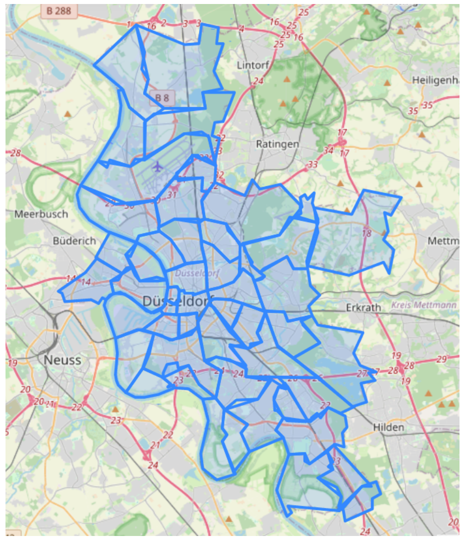

Figure 2: Polygons before and after simplification.

Figure 3 shows each restaurant in Düsseldorf, with Chinese restaurants being marked as blue dots. We can see that there is an abundance of restaurants especially towards the city center. We can aslo see that some districts consist mainly of uninhabited areas such as parks, industrial areas, sewage treatment plants, etc or are not densely populated. We drop these districts from our candidate list as they will lead to outliers in our analysis.

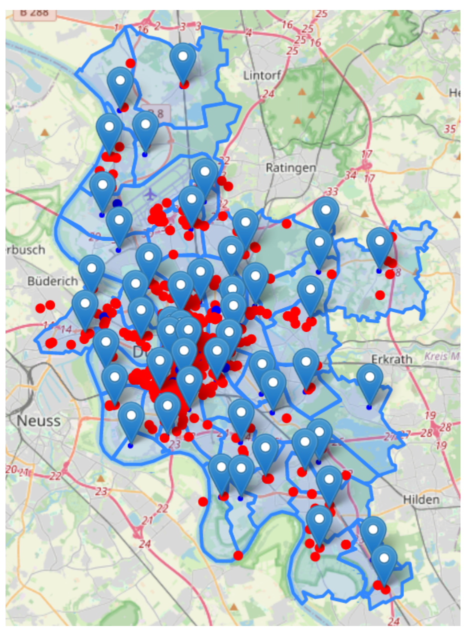

Figure 3: Restaurants in Düsseldorf. Red dot: Restaurant, Blue dot: Chinese restaurant

The results of our analysis are shown in Figure 4. We used a scatter plot to display the number of restaurants in each district over the distance of the respective district to the city center. The plot also visualizes the number of Chinese restaurants in each district by the size and color of each marker. If we look at this as a multi-objective optimization problem we can see that the resulting pareto front depicts optimal locations considering all predefined criteria. *Altdstadt, Carlstadt* and *Niederkassel* for example being pareto optimal solutions. These three districts have no Chinese restaurants and show a good trade-off between the number of restaurants in the district and their distance to the city center. However, considering that Niederkassel is on the left side of the *Rhein* river, which is a far less popular part of town, it might be a better idea to leave this one out in the further analysis. The other two districts are not only optima according to our criteria but also popular districts of Düsseldorf.

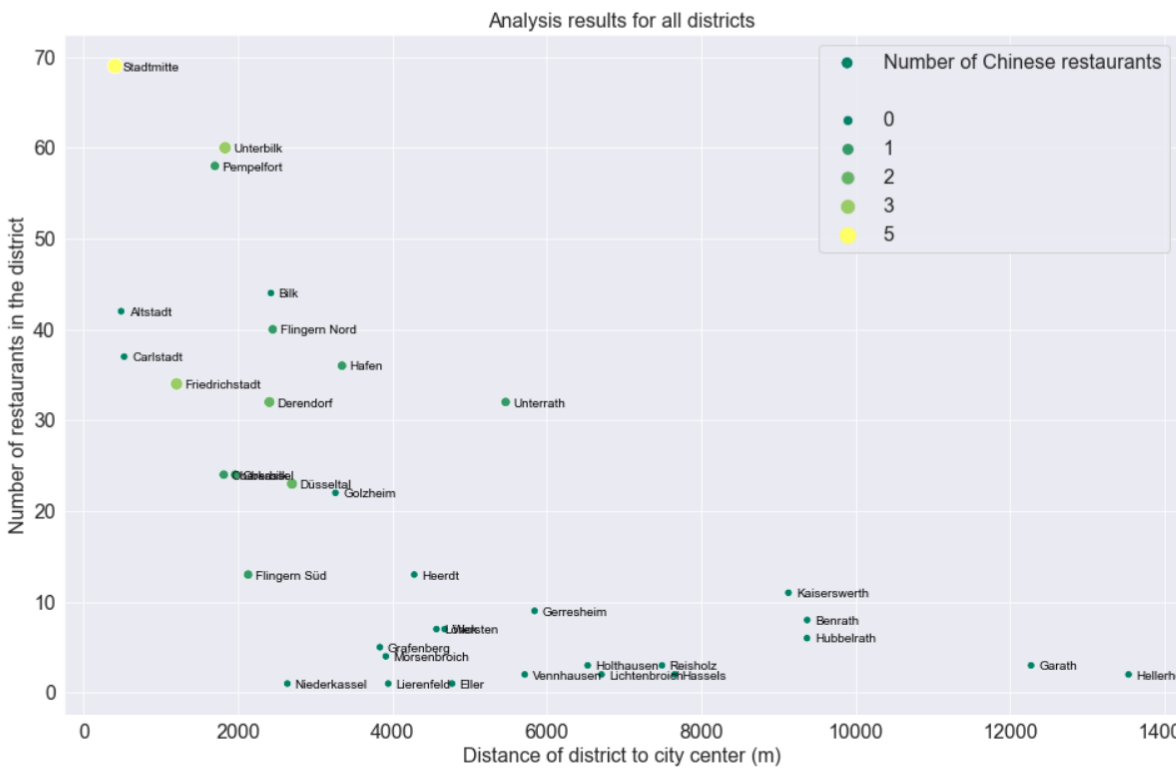

Figure 4: Results of the district analysis.

In the following, we focus our search efforts on these above-mentioned districts. We generate a list of possible candidate locations (street names) for both districts using the **Overpass API**. The candidate locations are shown in Figure 5. We then calculate their distance to the city center and to the next chinese restaurant. We also get the number of restaurants within a radius of 200m for each location. In the next step, we conduct a second analysis using K-Means to create **three** clusters of candidate locations with the three afore-mentioned parameters.

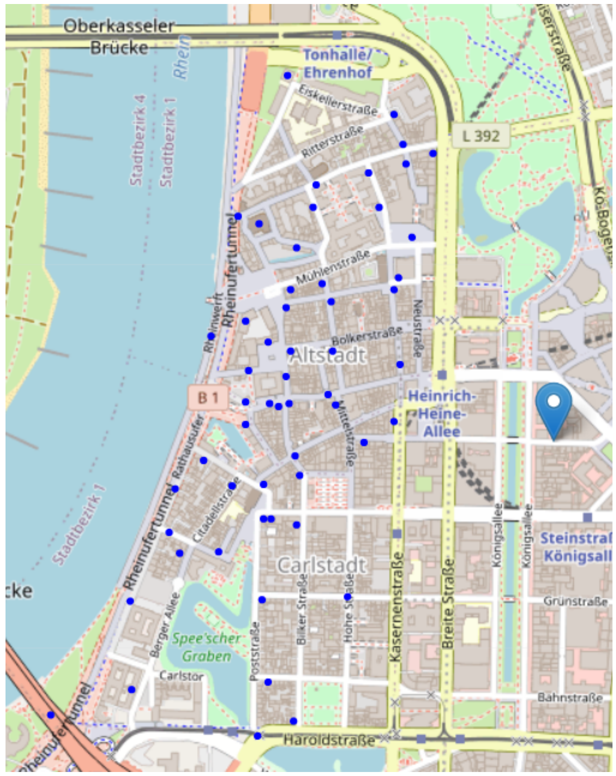

Figure 5: Candidate locations.

4 Results

The results of the analysis are shown in Figure 6. The visualization shows a clear separation between the clusters. Furthermore, in the first cluster, we have locations with no restaurants in the vicinity, a high distance to the next chinese restaurant and a higher distance to the city center. In the second cluster, we have locations with high restaurant denisty, a low distance to the next chinese restaurant and a medium distance to the city center. And in the last cluster, we have locations with medium restaurant denisty, a medium distance to the next chinese restaurant and a medium distance to the city center.

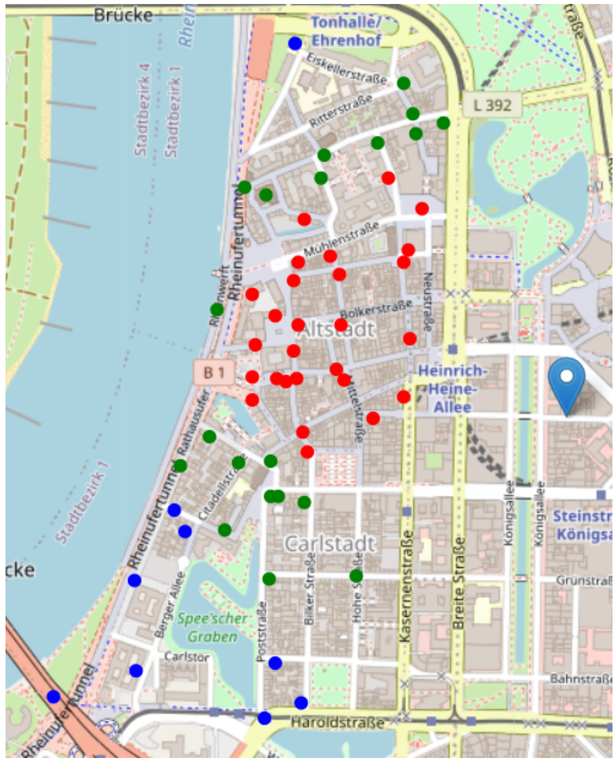

Figure 6: Results of the K-Means algorithm.

5 Discussion

Depending on the stakeholders prefererences, the locations in the first and second cluster can be considered optima. The locations in the first cluster have almost no restaurants nearby and are the furthest away from nearby Chinese restaurants. However, this could also be due to the fact that these locations have certain disadvantages such as few parking spaces, far from the city center, etc. The areas in the second cluster on the other hand have some restaurants nearby and are closer to the city center. These areas could prove to be better candidates. A deeper analysis with more parameters could clarify this.

6 Conclusion

In this study, we generated location proposals for a new Chinese restaurant Düsseldorf, Germany. We performed a two step analysis to reach our goal. In a first step, we identified the candidate districts in which we might be willing to open the restaurant. In the second step, we used unsupervised learning to generate clusters of candidate locations (street namess). Advantages of each cluster were clearly expressed so that the best possible location can be chosen by the stakeholders. Using the candidate locations the stakeholders can search for places to rent or buy.