Final Project Proposal

Sophie Ngo, Ting Chen, Michael Peeler

We will be using data of airport and metro station locations and Airbnb locations, price, and distance to nearest airport and metro stations. We will collect Airbnb listing and metro station data from three U.S. cities: New York City, Chicago, and Washington D.C., as well as the airport data from across the U.S. We are also recording the latitude and longitude coordinates of the city hotspots of the three cities (one per city) and are collecting this information ourselves via Google and Google Maps. The original datasets can be found at the following links:

Airports - data.humdata.org/dataset/ourairports-usa

Airbnb - <http://insideairbnb.com/>

DC metro stations - <https://opendata.dc.gov/datasets/DCGIS::metro-stations-regional/explore?location=38.903376%2C-76.740656%2C10.00>

Chicago Metra stations - <https://github.com/ChicagoCityscape/gis-data/blob/master/stations_metra/metra_entrances.geojson>

New York City Metro - <https://catalog.data.gov/dataset/subway-stations>

Our goal is to predict the nightly price of an Airbnb. We chose to do numeric estimation to achieve this because it is the most appropriate method of learning for this type of goal. We will want to see if the distance from an Airbnb to the city hotspot, nearest airport, and nearest metro station, alongside its other data such as number of reviews, type of living space, days available per year, etc., has an influence on its nightly price.

We will need to perform preprocessing for our data. We need to clear irrelevant attributes, since the datasets store a lot of information we don’t need, such as the name of the person listing the Airbnb and ID numbers of any kind. We’ll also need to comb through the data to find any missing values or noise, likely by using a Python script (since there are thousands of instances). We will delete instances that are detected to have noise since we have a plentiful number of instances, and the benefit of getting rid of any potential noise outweighs the cons of keeping potentially misleading data.

After cleaning the data, we must integrate the data. We need each instance to be an Airbnb, and alongside the original Airbnb data, it will store three additional attributes: the distance (in miles) from the Airbnb to the nearest airport, to the nearest metro station, and to the city hotspot. We are not actually carrying over any data from the airport and metro datasets because their only use to us is to serve as a checklist to find distances for.

Before finding any distances however, we need to find the latitude and longitude coordinates for all locations: the Airbnbs, city hotspots, airports, and metros. We have all this information except for the Chicago metro stations, so we need to get this data ourselves. We do this by getting JSON files containing this information from the Chicago metro station website, then writing a JSON Parser in Java that collects the latitude and longitude data, then saves it to a new .csv file. The city hotspots won’t need to do this, since we are only using 3 cities and add coordinates manually.

After that, we would be able to calculate distance. We plan to do this by writing a Python program that, for each Airbnb, uses the Euclidean distance between the latitudes and longitudes to find the shortest distances. We attach each shortest distance to the corresponding Airbnb instance in the integrated dataset we are building. The distance from the Airbnb to the city hotspot can be added separately from this searching algorithm.

As for splitting up the work, Michael plans to do the data collection and data integration using Python, Sophie plans to write the JSON Parser, and Ting plans to perform the data cleaning. We will then come together to do the data mining together.