Date:11/20/2019

Week 13 ETL Project Report

Extract:

Our original data was acquired through [www.kaggle.com](http://www.kaggle.com) in the format of two CSV files on New York City property sales and Air BNB listings.

New York City Property Sales Data :

https://www.kaggle.com/new-york-city/nyc-property-sales/data

New York City Air BNB lists Data:

<https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data>

Transform:

Airbnb: The clean-up/ transformation procedure for the Air BNB dataset, was a lot simpler compared to the Property dataset, mainly because there were no missing data. We started off by selecting a few columns that we thought were relevant (i.e airbnb id, price of the room, neighborhood group,...) and stored them in a new dataset called ***airbnb\_transformed***. Next, we renamed some of the columns, so that they’re more meaningful, and more readable to us. Even though the values for **airbnb\_id** are unique, we double checked for duplicates by using the .**duplicates()**function.

Property: For the sales property data, we noticed the sales data column included a date and a default timestamp of 00:00:00. Because all the rows had the same time stamp, we knew this value was a carry-over from the csv file and wasn’t capturing any additional information. So we transformed the sales date information to exclude the default timestamp by splitting the data/time string and just keep the date. In addition to those transformation efforts, we also wanted to convert our numerical-heavy columns, like sales price, into an integer datatype. However, several of the columns used a special character to represent the cell not having a value. We had to replace the special character with a null value so we could successfully submit our dataframe to SQL. This was successfully done by locating and storing the special character as a variable, then using that variable with **.replace()** function to replace all cells with the variable with a null object.

Load:

The schema was well laid out in the csv files, so we were able to use the formatting in the csv files to determine how to create the schema for our SQL database. This dataset isn’t particularly complex or unstructured, which would require using Mongodb. Using SQL will allow us to join and merge our tables later in the future, in order to look for interesting trends between property sales and Air BNB listings. If time permits, we would like to determine the zipcode of an area by examining the longitude and latitude coordinates from the Air BNB dataset. This would enable us to draw meaningful conclusions on the sale price of New York Cities properties based on the area.