ETL Project Technical Report

Recycled Materials in NY State in 2017

Group 9: Samuel Okunola, Phoebe Yaheng Wu, Maria Sierra Lizarazo

GitHub repository: https://github.com/mariasierralizarazo/ETL-project-Group-9

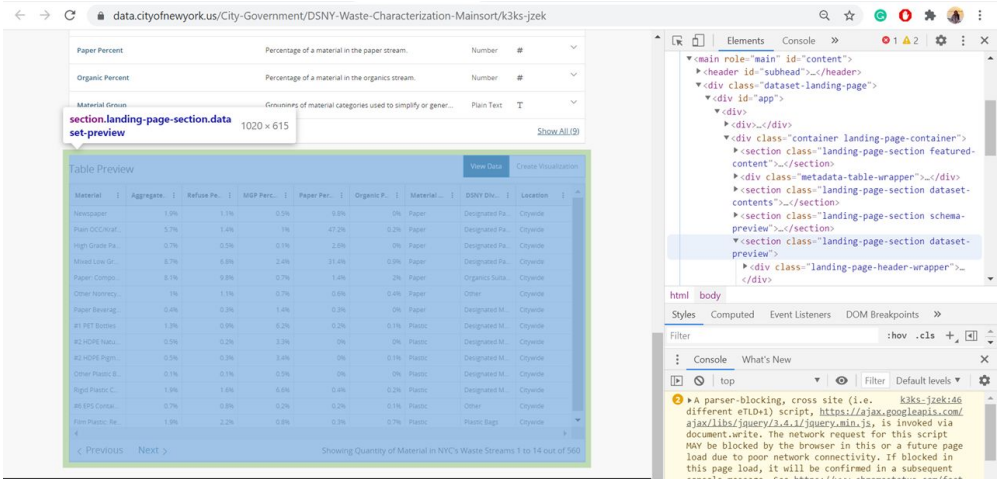
ETL (extract, transform, and load) tools are becoming more popular in the data world day by day. With the amount of available data nowadays, the possibilities of collecting, reading, merging, or migrating large volumes of data for easier access are useful and necessary. The main goal of this project is to apply the acquired knowledge in Python tools (especially Pandas) for loading a new database taking information out from different sources and in different formats.
The chosen databases were:

- DSNY Waste Characterization: Citywide Subsort 2017 (https://data.cityofnewyork.us/Environment/DSNY-Waste-Characterization-Citywide-Subsort/phkb-tkts)
- DSNY Waste Characterization: Mainsort 2017 (https://data.cityofnewyork.us/City-Government/DSNY-Waste-Characterization-Mainsort/k3ks-jzek)
- Recycling Diversion and Capture Rates (https://www.kaggle.com/new-york-city/nyc-recycling-diversion-and-capture-rates)

EXTRACT

- DSNY Waste Characterization: Citywide Subsort 2017 database was extracted by downloading the CSV file with the complete data and reading it, with Pandas.read_csv tool, into a Jupyter Notebook

- DSNY Waste Characterization: Citywide Mainsort 2017 database was extracted by doing a web scraping. *browser.visit* was used to go into the URL of the database and then with *BeautifulSoup* module the HTML code of the website was explored. There the information of the database was reading by taking the data in all the 40 pages of the table of the website.



- Recycling Diversion and Capture Rates data were extracted by calling the NYC Open Data API. In this case, it was necessary to request an API Token that allowed the exportation of the data in the main Jupyter Notebook.

TRANSFORM

After analyzing the datasets, it was possible to define the structure of the new database and the necessary transformations. One predominant point was to clarify what information would be kept and which one would be no necessity for the project. Also, it was needed to determine if the new database would be relational or non-relational.

For determining the database type, the following considerations were taken into account:

- All recyclable groups have different types of materials. For example, MGP is a very important group for recycling in the city, but this group contains plastic, glass, and metal materials found in items like bottles, bags, containers, organic containers, and many others. So, this irregular number of materials is key for a non-relational database.
- There is no record for all the materials in every single location for the Mainsort and the Subsorts. For example, there are no records for the paper group in the main sort, there are records for the MGP group in this sort. So, there is another key to have a non-relational database.
- Recycling Diversion and Capture Rates just have very important and useful information for the paper and the MGP groups, but it does not have information for other materials.

Each database had different transformations for getting the right format and creating our new database. The transformations are listed below:

- Recycling Diversion and Capture Rates
    - Filter just to take the information for the 2017 year.
    - Rename locations for grouping some of them.
    - Rename some columns for easier management.
    - Converting to numeric: the API call for extracting and reflecting the information in a Pandas DataFrame sent all the columns as object type. So, it was necessary to convert the data of the numeric columns in float datapoints. The tool use for this conversion was pd.to_numeric from the Pandas module.
    - Filter per location
- DSNY Waste Characterization: Citywide Subsort
    - Percentage columns were in decimal format, so multiplication for 100 was necessary to get the values in percentage.
    - For creating all documents for the non-relational database, it was necessary to filter per material, group by location, and calculating the average values in the parameters we were interested in.
    - Merging data with the Mainsort dataset and delete duplicates.
- DSNY Waste Characterization: Citywide Mainsort
    - As the information was taken from the HTML code of the website, all of the columns were read as text values (str). So, for the numeric columns, it was necessary to take out the ‘%’ symbol and then convert that value into numeric (pd.to_numeric). In that way, all the numeric data would be float64 data type.

- Same as the Subsort database, it was necessary to filter per material, group by location, and calculated the average values in the parameters we were interested in.
- Merging some data with the Subsort dataset and deleting duplicates.

For filtering the information the tool use was DataFrame.loc and for grouping DataFrame.groupby.

LOAD

The new non-relational database was loaded to MongoDB using *pymongo* module in the Jupyter Notebook.

Here some of the documents MongoDB shows after loading the information: