ETL Project

Data Analytics Bootcamp

Taylor Breychak, Indira Poovambur, Maria Wisco, & Tasneem Talawalla-Vora

10/22/2019

The purpose of this project was to use ETL (Extract, Transform, Load) strategy using at least two separate datasets. In this project, the team decided to focus on the data analyst and data scientist job market to keep the data as relevant as possible to the course and help the teammates analyze the job market following course completion. The topic was also chosen as something that could easily be analyzed for a larger project in the future. The project had a deadline of one week; as such, the team had to work together in order to complete the project using the best code variations and within the time frame.

The first step of the project was to find at least two sources of data for the "Extract" phase. The team decided to use Glassdoor.com and Indeed.com for the most up to date and relevant information related to job postings in the data market. The data was scraped from the chosen webpages and then downloaded as CSV files using Python code in Jupyter notebooks. The format was chosen after finding a CSV on the site, Kaggle.com, for the job market of data scientist jobs in highly populated areas, including New York, California, Washington, amongst others.

While extracting data from Glassdoor.com, the script had to account for pop-ups such as several screens to "sign-up" for the site's service. Selenium event handlers were used to close the pop ups to continue webscraping in a timely manner. Additionally, to ensure the script would complete, try and except rules were added into the code. Due to the amount of time to scrape the pages, it was noted that only data for Washington and Ohio could be scraped for this project.

The process for Indeed.com was more efficient due to the lack of pop-ups on the site. Due to this, data was able to be identified for the five states chosen of Ohio, Washington, Massachusetts, California, and New York was able to be scraped. However, again, due to the timing of the project and the scraping process, not all postings related to the chosen job titles of data analyst, data scientist, and data engineer were able to be pulled into the data.

The second step of the project including cleaning the data upon web scraping. This fulfilled the "Transform" phase of the data. Using the resulting CSVs from the code, the team looked through the job titles which had populated in the job searches. It was noted that all jobs with just one keyword of "data" "analyst" returned; for example, the job titles "research analyst" and "data grinder" were returned in the CSV files.

To filter the Glassdoor data, a new script was written using Pandas in a Jupyter notebook, to import the created data files in the prior step and pull only the titles of “data scientist,” “data analyst,” and “data engineer,” into a new data frame. Additionally, the returned locations in the data files had combined city and state into one column of “location.” Within the code, the column was separated to create a new column for both state and city. The new Glassdoor data frame additionally had commas; as such, the data could not be exported as a typical comma separated file. To avoid any complications for the later load, the data frame was exported as a pipe (|) separated file.

To filter the Indeed data, a schema was written using Postgres in SQL. This data frame also required job title filtering. A table was created in the database in order to load in the extracted CSV. Within the table job titles of “data analyst,” “data scientist,” “data engineer,” “business analyst,” were chosen as the most relevant in the data returned. Within the code, a wildcard search was established to delete any rows that did not match the chosen job titles. Once deleted, the Indeed data was cleaned as the location columns were pulled separated (city and state) through the initial web scraping.

To complete the project, the cleaned data then needed to be loaded into a final database. The team decided to use SQL for this step due to the team’s knowledge of the system as well as the efficiency of loading and exporting data within the code. A separate table was created for each the indeed and glassdoor data as there were likely duplicates of position entries if the data were to be directly compared. The tables were created to match the cleaned files as well as to add a primary key for future query comparisons. The two data files were loaded to one schema file.

In project reflection, it can be noted that webscraping for data can be quite challenging regarding popular webpages. As the team chose two popular job search webpages, the team encountered issues regarding pop-ups and queries taking long period of time. This was indicated to be a result of the page popularity-of many job searchers trying to access the parameters at once as well as the pages requiring accounts for full service. The team did try to engage in pulling data from Glassdoor.com specifically using an API. However, an API code for the site was only accessible upon company outreach. While such outreach was attempted, no response was received. While the API would have made for cleaner datasets to start with, the unpredictability of webscraping on the sites made for a greater challenge in cleaning the data and further learning in the project.

The ETL (Extract, Transform, Load) method was used to pull data from the two popular job search sites of Indeed.com and Glassdoor.com. The data was cleaned in order to pull specific positions related to the data analytics job market, most relevant to the team related to the bootcamp course. The team was able to utilize a single repository to easily share information and worked together in order to appropriately pull, clean, and eventually load a single data file. Additionally, the team was able to take a topic that will be useful following the completion of the

course, as the project members will be able to utilize such methods to hone future job searches in the data analytics market. The project members will also be able to utilize the data and code for future projects. Overall, the method is extremely useful for such large datasets and creating a broad code to handle future projects as well. The team was able to collaborate to share responsibilities to most efficiently complete this project.