ETL Project Report

**Extract**:

We pulled our datasets from Kaggle. The data was formatted in both CSV and JSON files. The datasets referred to the Top 200 trending Youtube Videos Statistics for the U.S, Great Britain, France, Germany, and Canada for every day since 2011. In all, the datasets provided a unique Video ID, the date on which the video trended, the title of the video, the YouTube channel from which the video came, a Category ID, a thumbnail, whether or not comments or ratings were disabled, whether or not the video has been removed, total counts for views, likes, dislikes, and comments, and a description of the video. Out of this information, we decided that the most pertinent information was the trending date, the video title, the channel title, the category ID, and the counts for views, likes, dislikes, and comments.

**Transform**:

In order to clean and transform our data, we used Pandas and Sqlalchemy in Jupyter Notebook. First, we imported Pandas and Sqlalchemy. Then, we created variables that stored the filepaths of our CSV files for the US, Great Britain, France, Canada, and Germany. Next, we read those CSV files into Dataframes for each country. Then, we created new versions of each Dataframe by copying only the columns in which we were interested by using the .copy() function. The columns that we copied contained information on the trending date, the video title, the channel title, the category ID, and the counts for views, likes, dislikes, and comments.

**Load**:

Once the transformation process was complete, we then established a connection to SQL via the create\_engine function. In order to create this connection, we passed a string into the create\_engine function that contained our PGAdmin username and password. After establishing this connection, we used the to\_sql() function to pass our Dataframes into the 5 SQL tables we had previously created. From there, we were able to access our data in SQL.

**Possible Uses**:

* Sentiment analysis — parsing through comments, likes, and dislikes.
* Categorizing YouTube videos based on type.
* Documenting the kinds of comments that each video receives.
* Tracking the kinds of videos that receive the most views, likes, and dislikes.
* Training Machine Learning algorithms like RNNs to generate their own YouTube comments.
* Analyzing what factors affect how popular a YouTube video will be.
* Statistical analysis over time of likes, dislikes, views, or comments.