While working through my dataset, I was pretty fortunate as it was mostly already clean. The structure of the data was a little odd, however and that posed some challenges. While the file was a CSV and structured in a tabular format, several of the columns within the dataset had multiple values. These values were stored in a JSON structure within the CSV. Loading the dataset into Pandas made it hard to visualize the columns in this condition. To work around this, I first turned each of the JSON values from the string they had been imported as to a dictionary object. From there, I converted the row into a list of the values only, removing the keys as they would not be necessary for this assignment.

An example of this transformation would be the Genres feature, which listed the genres of film each movie belonged to. The keys were an id number for the genre, while the value was the actual genre name. As we were focused more on text analysis here, I removed the ids and kept only the names. Had this been a deeper project, I likely would have created a second genres table containing the ids as a primary key and names as an attribute.

After performing the necessary transformations, I took a look at a summary of the numerical data to search for outliers. While there are values for features like budget and revenue that are out of line with the mean, they all appeared to be accurate. Depending on what kind of further analysis we were doing, we might want to consider how to best treat the outliers. If we’re looking at this data to see how a small-budget film would perform, we may not want to include data from very high budget films such as Pirates of the Caribbean or Avengers.

My next step was to look for duplicates. I wanted to do this slowly, by first starting by looking for films with duplicate titles and flagging those as potential duplicates. From there I was going to filter down further using values such as the production company, genres, or runtime. I found only two cases of potential duplicates within the data and both were not actual duplicates but just films that happened to have the same title as another film.

The final step of the project was to conduct fuzzy matching. I decided to do this with the plot of each film. To ensure the best performance possible, I first took the plot data and converted everything to lowercase, then stripped all punctuation. I experimented with four different methods of fuzzy matching and found the best performance was from token sort ratio. I analyzed the performance both by the scores returned and also by using the extract method to see which films were most similar in plot to a specific film. While it certainly shouldn’t be the only factor I can see how an algorithm like this could be a useful part of a recommendation system for a company like Netflix.