I. Code:

The code for my solution is at: https://github.com/susiehuynh/CS4400

II. Solution Outline:

The solution includes five steps:

1. Data reading and EDA
   - In the first step, we read three tables: left, right, and training. We read the data set as pandas in order to help understand all the tables and design the solution. In total, we have 56376996 pairs (2554 rows for left table and 22074 rows for right tables). In order to efficiently examining every pair, we need blocking step to remove unnecessary pairs and reduce number of pairs that needed to be handle.
2. Blocking (by Branch)
   - In this step, we perform blocking on an attribute "brand." This blocking process will pair ids from two tables in a list based on brand. By doing so, we could likely get two products with the same entity. This blocking also filters out unnecessary pairs and allows us to easily handle the problems. After blocking using "brand," we left with 256606 pairs to examine.
3. Feature engineering
   - For each pair in the common set, we would apply the Jaccard similarity and Levenshtein distance of pair on five attributes in left and right table to generate a feature vector of 10 dimensions. By doing so, we obtain a feature matrix X_c for the common set. Then, we repeat the same process to those pairs in training set to obtain a feature matrix X_t. The label for the training set is denoted as y_t.
4. Model training
   - In this process, we first test the data using cross validation with random forest classifier, KNN, SVM, and Naive Bayes machine learning method to find the best model parameter. We find that the random forest classifier is the best method with highest "f1" score. GridSearchCV were also used to obtain the optimal value for the parameter inside random forest classifier like "n_estimator" and "criterion". Further, a random forest classifier method was rerun using the optimal value for the n_estimator and criterion from GrindSearchCV to increase the "f1 score" value as high as possible.
5. Generating output
   - According to the indication, pairs with $y_c = 1$ are those predicted matching pairs in our M set. Since we want to exclusive any matching pairs in training tables, we will obtain a new predicted matching pair M^- set where all matching pairs in training tables get filtered out from M. Once the process complete, M^- will be exported to CSV file called "output.csv."