INFSCI 2725: Data Analytics
Final Project: Introductory report

Home Depot Product Search Relevance

A competition by

kaggle

0. Overview

0.1. Group Member

	Name	Pitt email
1	Mohammed Alharbi	maa271@pitt.edu
2	Yixuan Edison Wang	yiw72@pitt.edu

0.2. Group Name

Our group name in Kaggle is MAYW, which is the initials of our first and last names.

0.3. Introduction

Home Depot become an important store that home owners rely on to find solutions to their home improvement needs, ranging from installing a new ceiling fan to remodeling an entire kitchen. Customers expect the correct results to their queries when they use Home Depot's system. Speed and accuracy are essential.

This project is a competitive project published by Kaggle, which is a platform for data prediction competitions. In this competition, Home Depot encourages Kagglers to improve their customers' shopping experience by developing a model that can accurately predict the relevance of search results. They provide three datasets about products, their attributes, and some search terms and their relevance.

Technically, Kagglers have to used data that were stored in file train.csv to predict the search relevance in another file, test.csv. They have to fit some model such as Linear Regression Model to learn from train set, and apply or predict the test data set.

To tackle the problem, we firstly bind the necessary datasets. Working in one dataset containing all needed data is more efficient. Then, we try to clean the data to eliminate missing values and unnecessary records. Next we fit some models such as Linear Regression Model and Generalized Boosted Model. Next, we predict the relevance of search terms using that model. Before generating the submission file, it is essential to test the accuracy of each model by computing Root Mean Squared Error (RMSE), highest accurate will be applied.

0.4. Programing Language

Before processing data, it is crucial deeply to think which programming language is going to help us to tackle the problem. We primarily use R programming for couple important reasons. R is essentially build to tackle problem involving exhausted statistical operations. R also has a great environment to adopted statistical graphs. It eases the way by which we draw, edit, and extract the graphs with fully control. Therefore, it gives us a way to understand the features of some main variables in the datasets.

1. Data

In this project, data is only the subject we are going to deal with. To take advantage of these data and learn their patterns, it requires a clear understanding of data features. This part will introduce the data features.

1.1. Data files

There are six files that produced by Kaggle. The following is their descriptions:

- **train.csv** - the training set, contains products, searches, and relevance scores.
- **test.csv** - the test set, contains products and searches. You must predict the relevance for these pairs.
- **product_descriptions.csv** - contains a text description of each product. You may join this table to the training or test set via the product_uid.
- **attributes.csv** - provides extended information about a subset of the products (typically representing detailed technical specifications). Not every product will have attributes.
- **sample_submission.csv** - a file showing the correct submission format.
- **relevance_instructions.docx** - the instructions provided to human raters.

1.2. Data fields

The following is the descriptions of all fields that the files contain:

- **id** - a unique Id field which represents a (search_term, product_uid) pair.
- **product_uid** - an id for the products.
- **product_title** - the product title.
- **product_description** - the text description of the product (may contain HTML content).
- **search_term** - the search query.
- **relevance** - the average of the relevance ratings for a given id.

- **name** - an attribute name.
- **value** - the attribute's value.

2. Data Exploration

2.1. Exploring Train and Test data

There are 74067 rows in the train.csv file and 166693 in the test.csv file. Both have the same fields: id, product_uid, product_title, search_term, and relevance. However, the relevance is not included in the test.csv file.

2.2. Exploring search terms

- Number of search terms in train: 11795.
- Number of search terms in test: 22427.
- Number of terms in train not in test: 2174.
- Number of terms in test not in train: 12806.
- Number of common terms in test and train: 9621.

2.3. Exploring product data

- Number of unique product ids in train: 54667
- Number of unique product ids in test: 97460
- Number of common product ids: 27699
- Number of product id in train not in test: 26968
- Number of product id in test not in train: 69761
- Number of product id in product descriptions: 124428

Examining the column product_id in both train.csv and test.csv files, there are 54667 unique values in the train.csv file and and 97460 in the test.csv file. There are only 27699 common product_uid values among these. The total number of unique product ids across both the train and test data is 124428, which equals the total number of product ids and rows in the product_descriptions.csv file. This is displayed below in a Venn Diagram.

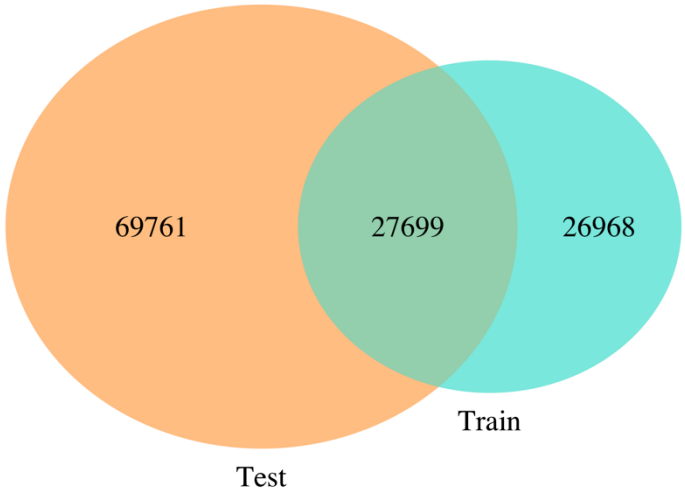


Figure 2-1: A Venn Diagram of product ids in both Test and Train

Examining the attributes.csv file, there are a total of 2044803 rows and a total of 86264 unique product ids. The intersection of product ids across the train, test, and attributes files is displayed in the Venn Diagram below. There is only one value in attributes.csv file that is neither in the Train nor Test files. On examining this, there are 155 rows in attributes.csv file that do not have a product_uid value. As it is unneeded, these rows can be **removed**.

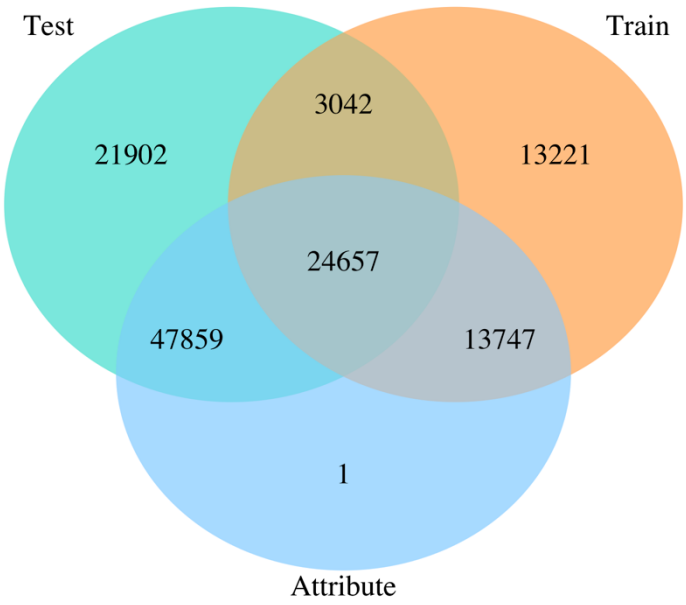
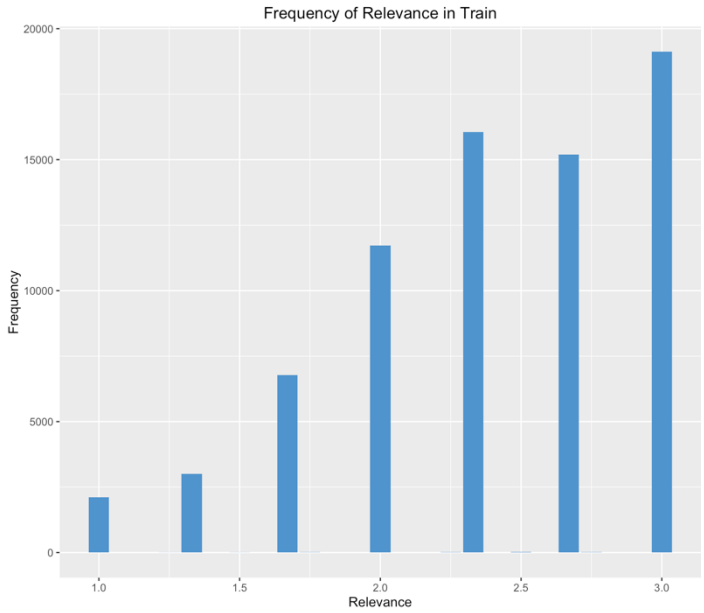


Figure 2-2: A Venn Diagram of product ids in the train, test, and attributes files

2.4. Train data

On investigating Train data, there are a total of 13 unique relevance values. A bar-chart of their frequency is displayed below. Note there are sets of relevance that occur with very low frequencies.



2.5. Attributes data

Examining Attributes data, there are 2044803 rows. Of these only 86264 are unique products. Moreover, there are 5344 unique categories (or name) of attributes. A sorted count of the top 10 for each category are displayed below.

Attribute's Name	Frequency
mfg brand name	86,250
bullet02	86,248
bullet03	86,226
bullet04	86,174
bullet01	85,940
product width (in.)	61,137
bullet05	60,529
product height (in.)	54,698
product depth (in.)	53,652
product weight (lb.)	45,175

Table 2-1: Count of top 10 Frequency of categories in attributes data

On investigating this data, 63 names refer to a color aspect. Similarly, 10 names refer to brand aspect. A sorted count of the top 10 for each are are displayed below.

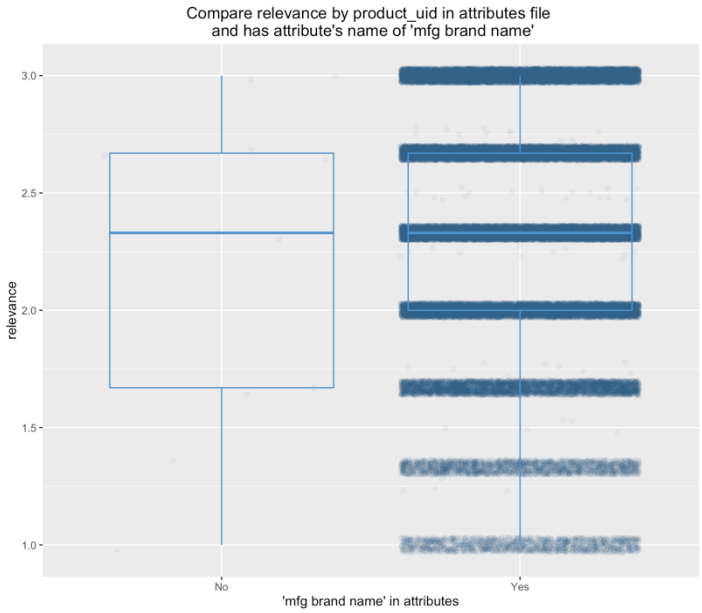
Attribute's Name	Frequency
color family	41,508
color/finish	28,564
color	6,222
color/finish family	4,630
fixture color/finish	4,119
fixture color/finish family	2,256
shade color family	2,006
actual color temperature (k)	1,421
color rendering index	1,118
top color family	996

Table 2-2: Count of top 10 frequency of color categories

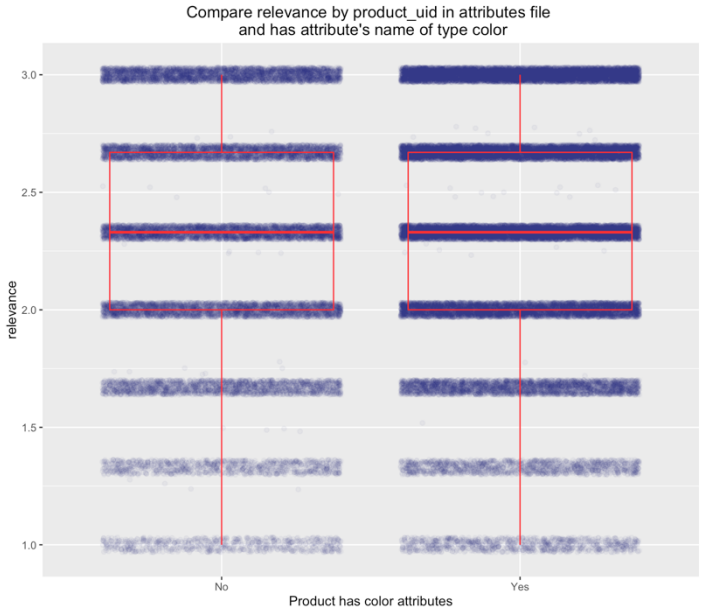
Attribute's Name	Frequency
mfg brand name	86,250
brand/model compatibility	681
brand compatibility	609
fits faucet brand	550
fits brands	407
fits brand/models	168
fits brands/models	119
pump brand	87
fits brand/model	6
brand/model/year compatibility	4

Table 2-3: Count of top 10 frequency of brand categories

The **mrg brand name** looks like the standard for brand attributes. In terms of color, the information is spread across multiple fields. A comparing relevance by product_uid in Attributes data and having attribute's name of 'mrg brand name' is displayed below.



The following is a comparing relevance by product_uid in Attributes data and having attribute's name of type color.



3. Tackling the problem

3.1. Binding data

We firstly bind the datasets that we think will help us to fit models. Working in one dataset including all needed data is more efficient. It gives us an opportunity to load data in

memory with the minimum code lines. In this step, we merge product_descriptions.csv with both train.csv and test.csv files. Then, we bind train.csv and test.csv files in one file, called all.csv.

3.2. Cleaning data

It's necessary to make training dataset more complete and uniformed in the beginning, which could make the later learning more accurate. We start, in this step, with eliminating missing values from attributes.csv file. Now we work on correcting misspelling and making data uniform. For instance, toilet word may misspell as 'toliet'. Also, numbers may exist in search terms as letter, but it was inserted in product_descriptions.csv as a number. We have started to uniform the numbers in all fields and change them to numbers. In addition, in cleaning data aspect, we include the 'SnowballC' package to our work as it helps us to gain the word stem. We think that swapping words by its stem will increase the accuracy. We sill have a great idea in this step, and we try to apply to our project.

3.3. Data Prediction

Prediction requires preparing explanatory variables that major to predict the relevance scores. Five explanatory variables are computed as the following: numbers of occurrence of each searching term's word in the product title, description, brand, and color. All these variables are appended to the major dataset, all.csv. There are couple models that are considered to tackle the problem. As far, we are able to apply two models: Linear Regression Model and Generalized Boosted Model. Fitting both of them gives almost same RMSE; It is 0.5357882 in Linear Regression Model and 0.5339374 in Generalized Boosted Model.

4. Result and Conclusion

To sum it up, Home Depot Product Search Relevance competitive project urges us to learn deeply and practically data analytics. As applying some technique being thought in this course, the first submission for us in Kaggle got 740 place. It is achieved 0.50291 score right now in Kaggle, resulting in ranking 1071. We will continue working hard to improve our place in the competition, hopefully winning the competition.

5. Reference

[1] “Home Depot Product Search Relevance | Kaggle” <https://www.kaggle.com/briantc/home-depot-product-search-relevance/homedepot-first-dataexploreation-k/notebook>
[2] “Using a GBM for Classification in R on Vimeo” <https://vimeo.com/71992876>
[3] “R-bloggers | R News And Tutorials Contributed By (573) R Bloggers” <http://www.r-bloggers.com/>
[4] “inside-R | A Community Site for R | A Community Site for R – Sponsored by Revolution Analytics” <http://www.inside-r.org/>