MIDAS Task 3

Task 3: NLP Assignment Details- Use a given dataset to build a model to predict the category using description. Write code in python. Using Jupyter notebook is encouraged.

1. Show how you would clean and process the data
2. Show how you would visualize this data
3. Show how you would measure the accuracy of the model
4. What ideas do you have to improve the accuracy of the model? What other algorithms would you try?

About Data : You have to clean this data, In the product category tree separate all the categories, figure out the primary category, and then use the model to predict this. If you want to remove some categories for lack of data, you are also free to do that, mention this with explanation and some visualization.

Dataset link: https://docs.google.com/spreadsheets/d/1pLvofNE4WHokpJHUIs-FTVnmI9STgogo5e658qEONoI/edit?usp=sharing

1) Cleaning and processing the data:

After importing the standard libraries (Pandas, Numpy, Matplotlib, Scikit learn) and the dataset, the dataset was observed to study the available features.

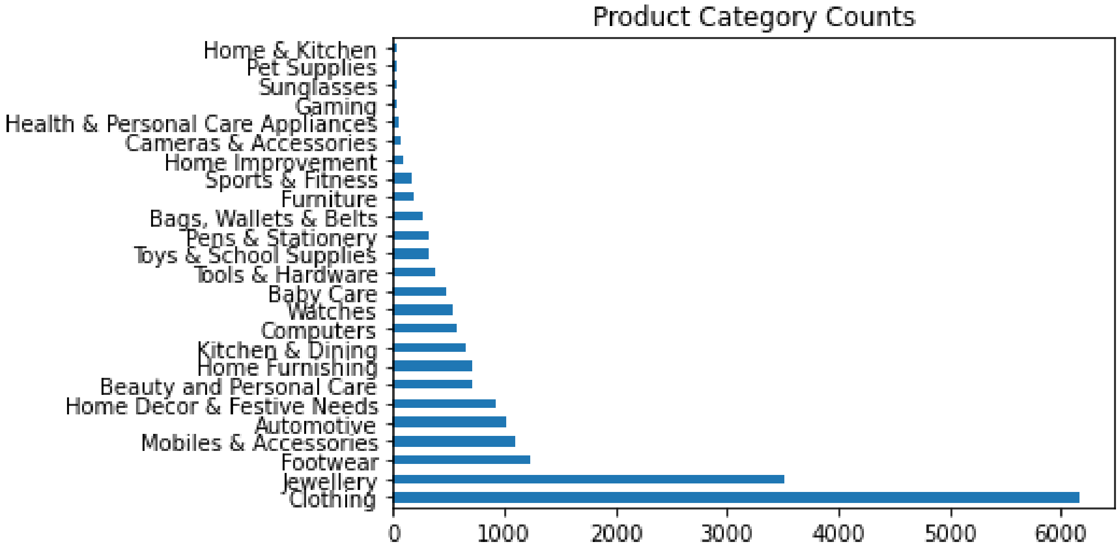
The features 'product_category_tree', 'description', 'product_name' had categorical data and hence were processed to the required format using functions and converted into the string datatype. The features, uniq_id, crawl_timestamp, product_url, pid, retail_price, image, product_rating, overall_rating, which were insignificant to the problem were dropped from the data frame.

Further, the 'discounted_price' column was converted to 'int' datatype from the 'float' datatype.

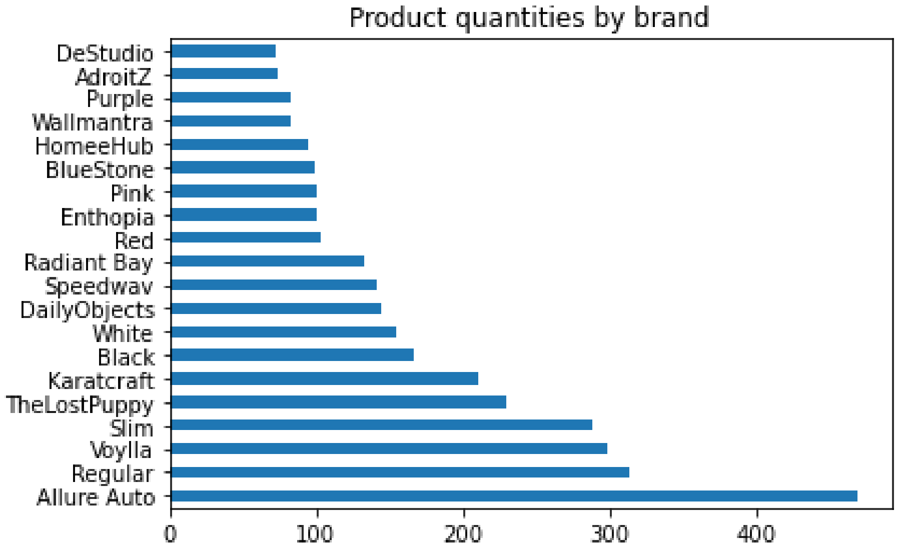
2) Visualizing the data:

Initially, the dataframe had 20,000 rows and 15 columns. After processing the data, the dataframe had 19,922 rows and 7 columns.

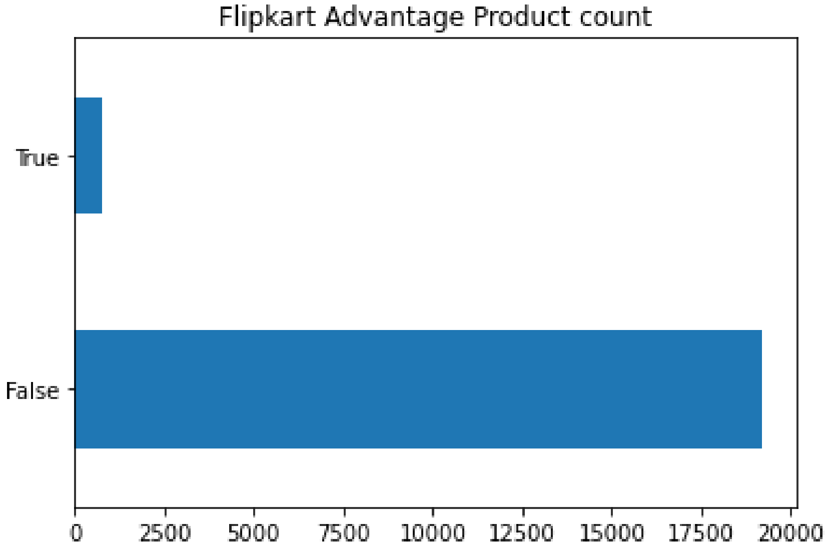
The value counts of some significant features were taken and plotted for some insights:



As can be observed from the graph above, the category with the most products is 'Clothing', followed by 'Jewellery', 'Footwear' and others. The primary category is 'Clothing'.



The most selling brand is 'Allure Auto' followed by 'Regular', 'Voylla' and others.



It can be seen that most of the products are without Flipkart advantage.

It was also found that the products with a **maximum price of Rs.5,71,230** and the **minimum price of Rs.35** had been sold.

3) Training a model:

The model was trained on the feature 'description' to classify the products into the top three product categories. As was observed from the graph above, the primary category was 'Clothing', followed by 'Jewellery' and 'Footwear'. Since the target variable had to be encoded, a label encoder was used to achieve this.

To create the test and train sets, a new dataframe was created consisting of the columns, 'description' as the feature and 'product_category_tree' as the target, and the products which belonged to the categories in the top three category list were chosen.

Before fitting the model, since 'description' had categorical data, CountVectorizer and TfidVectorizer were used to convert a collection of text documents to a vector of term/token counts or to normalize the data.

For this particular problem, Naïve Bayes Classifier was used as kNN Classifier and Random Forest Classifier could not be used with categorical features. Firstly, the model was trained after using CountVectorizer and afterwards, using TfidVectorizer.

4) Accuracy of the models:

The accuracy of the models was measured by printing the classification report using the sklearn library.

Naïve Bayes Classifier:

Classification report (using CountVectorizer):

	precision	**recall**	**f1-score**	**support**
0	1.00	0.99	1.00	1180
1	0.98	0.98	0.98	257
2	0.99	1.00	0.99	747
accuracy			0.99	2184
macro avg	0.99	0.99	0.99	2184
weighted avg	0.99	0.99	0.99	2184

Accuracy: 0.9935897435897436

Classification report (using TfidVectorizer):

	precision	**recall**	**f1-score**	**support**
0	0.99	1.00	1.00	1180
1	1.00	0.94	0.97	257
2	0.98	1.00	0.99	747
accuracy			0.99	2184
macro avg	0.99	0.98	0.99	2184
weighted avg	0.99	0.99	0.99	2184

Accuracy: 0.9963369963369964

5) Alternate approaches:

The classification report looked promising for both the models. Although, an attempt was made to add another feature in addition to description. The feature 'product_name' was chosen for this purpose. However, some error occurred and the model couldn't be completed.

Another approach could be to not drop the missing 'discounted_price' values that were dropped earlier and training the model in this way with more data.

6) References:

- scikit-learn 0.24.1 (https://scikit-learn.org/stable/modules/naive_bayes.html)
- pandas (https://pandas.pydata.org/docs/user_guide/index.html#user-guide)
- Product Category Prediction (https://www.kaggle.com/arindamroy23/product-category-prediction-99acc)