AMAZON MACHINE LEARNING CHALLENGE

TEAM - RANDOM_STATE - METHODOLOGY USED

1) Data cleaning

1.1 **Handling missing values:** Removed all rows which had even one NaN value in any column. This reduced number of points from 2,903,024 to 2,026,638.

1.2 **Removed duplicates:** Kept a single row for a unique title. This further reduced the number of points to 2,003,527.

1.3 **Reducing the number of classes:** A lot of BRAND_NODE_IDs has only one or two representative points. We discovered that 95% of points are covered by just 2759 classes. We removed all the points that did not fall in these classes. The number of points was finally reduced to 1,903,408. This data was saved into a file named 'filtered_data.csv'.

2) Data Pre-processing

Most of these steps were performed using the regular expressions (re) library.

1.1 **De-contraction of words:** Words like "won't" and "can't" will be converted to "would not" and "can not".

1.2 **Removal of HTML tags:** Using the 'Beautiful Soup' library.

1.3 **Removal of characters that are not AlphaNumeric**

1.4 **Stemming:** Used NLTK's SnowballStemmer.

1.5 **Stopword removal:** Used NLTK's collection of stopwords.

1.6 **Lowercase conversion:** Converted all words to lowercase.

3) Feature Engineering

- We **combined all text features** into one (Title + Description + Bullet Points + Brand) after pre-processing.
- We then created hashed vectors of dimension 200,000 using sklearn's **HashingVectorizer**

The shape of the data at this point was **(1903408, 200000)**

4) Training

We trained a Multinomial Naïve Bayes model with sklearn's **MultinomialNB** class, setting with **alpha = 0.0001**.

5) Submission

1.1 **Pre-processing of test data:** We concatenated all columns as done in for the training data and used the word "empty" wherever the value was NaN.

1.2 **Feature generation:** We transformed the data with the same HashingVectorizer used for the train data and got the desired shape of (110775, 200000).

1.3 **Prediction:** The submission file was created with the predictions generated by the MultinomialNB model.