Predicting E-commerce Marketplace Listing Attributes with Random Forest Classification

Laura Campbell, *School of Computing, Dublin City University, Ireland.*

Abstract – Attribute prediction is a machine learning task that aims to predict the value of specific target attributes. There are many applications for this task such as stock price predictions for financial analysis, and medical diagnostics based on patients' symptoms. There are many machine learning techniques that can be used to achieve this task such as support vector machines, neural networks, and regression. This paper will review the use of a Random Forest Classifier (RFC) on product attribute predictions. The paper discusses a review related works, the methodology used, and an analysis of the results obtained. The performance of the RFC model was evaluated using F1 Score.

CCS Concepts*:* • ***Computing methodologies → Random Forest Classifier; Information extraction; Text representations, Ensemble***

.

Additional Key Words and Phrases: attribute prediction, text analysis, text representation model, text classification model, classification

.

I. INTRODUCTION

Classification of product listings on e-commerce platform is an example of machine learning application being used in industry. The implementation of an effective machine learning classifier can play a significant role in inventory management, search results and product recommendations. This can help reduce resources required for manual tasks and help promote efficiency. Attribute prediction is a task in machine learning that aims to predict the value of specific target attributes based on other attributes provided in a dataset. There are various classifiers available to perform such a task such as support vector machines, decision trees, and logistics regression. This report will research the use of the RFC for product attribute predictions. The dataset used for this research was acquired from an e-commerce marketplace that hosts its platform to over 5 million active sellers, with almost 100 million active sale listings selling goods and services. The benefit of this type of machine learning task in e-commerce is it could help generate predicted product attributes required for active and new listings, while reducing the manual workload that would be required by a human performing the same task. The dataset contains the variables such as the title, description, and tags for listings, along with product identification number, colour identification and various other corresponding categories.

The objective was to build a classification model to accurately predict colour identification, top categories, and bottom categories of a given listing based on other descriptive data being fed into the model. The performance of the model is evaluated using F1 score and accuracy.

II. RELATED WORK

Random Forest (RF) can be an adequate choice for attribute predictions due to ability to handle non-linear classification tasks effectively [1]. RF is an ensemble learning method. The classifier is created by combining a collection of tree-

structured classifiers. Each tree is independently created from one another and has a random subset of features on a subsection of the dataset provided. This allows each independent tree to make its own predictions, the final prediction is a combination of all the trees predictions and based on the most popular class from the input provided [2]. The benefit of RFC is its ability in handling imbalanced, noisy data, and large datasets. RFC can be a faster alterative to other classifiers as the multiple independent trees within the RFC architecture allows for parallel processing [1].

Breiman [2] who first provided a general definition of RF, had noted in his paper that increasing the number of trees in RF would improve the performance of the classification. However, other research indicates less suggestive improvement in classification performance when increasing the numbers of trees pre-determined limits [3]. More recent studies suggest that classification performance is improved by determining the optimal number of trees and using feature importance [4].

Random forests-based feature selection (RFFS) has demonstrated resistant to overfitting, as its aggregates results of multiple decision trees with one study showing favorable results when used for text classification [2] [5]. However, the feature selection of individual trees in RFC can be biased in performance by the ordering of features selected during construction of the tree. Some features may be selected more frequently due to correlation with other features or by chance. Other features may be more dominant. This can result in over-representation of some features in the final feature set, while other important features may be left out [6]. By evaluating feature importance, this can help reduce impact of such biases.

III. Methodology

3.1 Data

The data employed for this research was acquired from an e-commerce marketplace that hosts its platform to over 5 million active sellers, with almost 100 million active sale listings selling goods and services. The training data has 245,485 products and the test data provided 27,119 products. Both datasets contained the following attributes: product_id, title, description, tags, type, room, craft_type, recipient, material, occasion, holiday, art_subject, style, shape, pattern. The additional data in our training dataset included our variables that we were trying to predict and their corresponding text columns: bottom_category_id, bottom_category_text, top_category_id, top_category_text, color_id, color_text.

3.2 Pipeline

The pipeline for RFC is illustrated in Figure 3.2.1. There are four steps in the pipeline process. Step 1 uses a custom column selector that pulls our numeric data (*product_id*) into a numeric pipeline. Step 2, a text custom selector selects specific text data columns from the data frame. Two methods were trialed for this. Method 1, each text column was pulled into a separate text pipeline. The text data was preprocessed to transform all text to lowercase, removal of stop words, special characters and emojis. Term Frequency – Inverse Document Frequency (TF-IDF) was then used as a weighting factor for

information retrieval of the text data. Method 2 concatenated the multiple text columns into a single text column which was then preprocessed using the same techniques. Step 3, Feature unions combined the pipelines together. Step 4 implements a Multiple Output Classifier, which is a wrapper classifier, that allows the RFC model to handle multiple target columns concurrently.

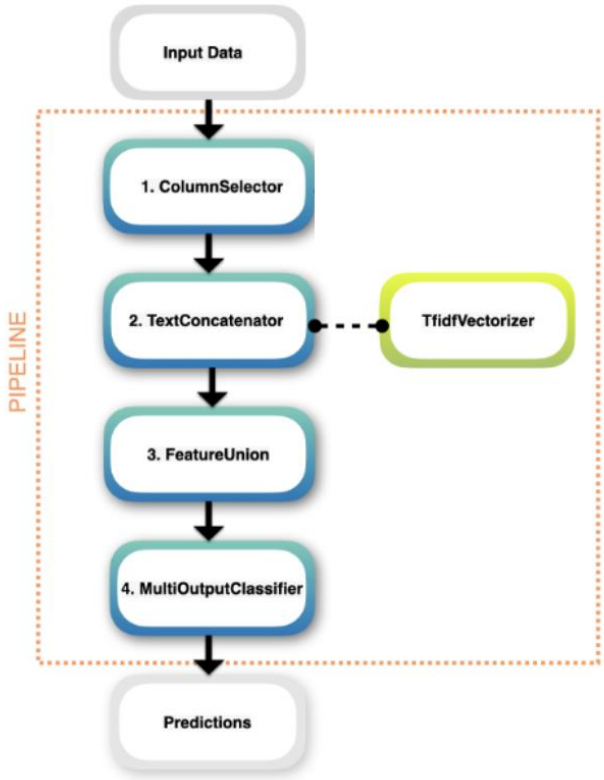

Figure 3.2.1: *The prediction pipeline for classifier*

3.3 Experiments

The following parameters were used in the model: our train data which used a 30/70 split; the number of trees in the forest; the maximum depth of each tree; the minimum number of trees; the minimum number of samples for each leaf node; the minimum number of sample splits for leaves; and the max number of features considered. A 'GridSearchCV' was applied to help optimize and fine-tune the hyperparameters of the model by cross-validating grid search over a parameter grid. The first initial grid used the following parameters as illustrated in Figure 3.3.1. The best recommend parameters after several attempted runs can be seen in Figure 3.3.2.

```
# Define the parameter grid for GridSearchCV
param_grid = {
    'clf__estimator__n_estimators': [100, 200],
    'clf__estimator__max_depth': [10, 20],
    'clf__estimator__min_samples_split': [2, 5],
    'clf__estimator__min_samples_leaf': [1, 3],
    'clf__estimator__max_features': ['sqrt', 'log2'],
}
```

Figure 3.3.1: *The initial parameters for running the model*

```
Best parameters found:
{
'clf__estimator__max_depth': 30,
'clf__estimator__max_features': 'sqrt',
'clf__estimator__min_samples_leaf': 1,
'clf__estimator__min_samples_split': 2,
'clf__estimator__n_estimators': 300
}
```

Figure 1.3.2: *Recommended parameters produced by GridSearchCV*

To further fine-tune the model, feature importance was assessed during the model training. Figure 3.3.3 illustrates the results obtains for feature importance. 'title' and 'description' are shown to be the most dominate important feature in the dataset.

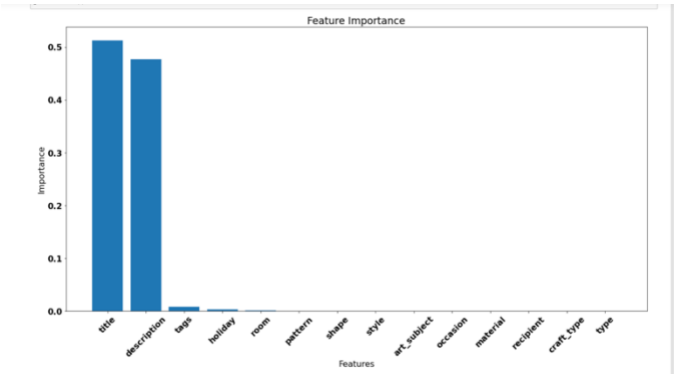

Figure 3.3.3: *Feature Importance of RFC*

Multiple runs of the training model were conducted to assess the different results acquired when features of lesser importance had been removed. Additionally, it was tested to see if different results were gained from using multiple text pipelines to treat each text feature separately, and a single text pipeline that concatenated the multiple text columns into a single text column.

IV. RESULTS AND EVALUATION

The first initial run of the model gained poor results with lower than 25% accuracy. The initial instance used a numeric pipeline, which called on the product_id, a unique id that identifies the individual rows. After removing this from the pipeline, the results improved to over 30%. It suggests that there are no meaningful patterns to be establish from this feature. The assumption could be made that product_id may have contributed to the model learning irrelevant patterns. The use of GridSearchCV helped us to obtain the best recommend parameters after several attempted runs as illustrated in Figure 3.3.2. After this, it was found that by concatenating the multiple text columns into a single text column, better results had been achieved. This suggests that the text concatenation may have allowed for the entire context of each row instance to be considered when learning the representation and patterns between words and phrases across different text columns. Feature importance was then assessed, with less important features being removed from the model to see if accuracy increased. It resulted in a decrease in accuracy each time an unimportant feature was removed. The highest results achieved can been seen in Table 4.1. This was obtained using the best recommended parameters recommend by GridSearchCV and using text concatenation.

Table 4.1: *Evaluation results of RFC*

Bottom Category ID	**PRECISION**	**RECALL**	**F1-SCORE**
Accuracy			0.36
Macro Average	0.41	0.37	0.35
Weighted Average	0.41	0.36	0.35
Top Category ID	**PRECISION**	**RECALL**	**F1-SCORE**
Accuracy			0.53
Macro Average	0.90	0.30	0.34
Weighted Average	0.78	0.53	0.47
Color ID	**PRECISION**	**RECALL**	**F1-SCORE**
Accuracy			0.33
Macro Average	0.70	0.16	0.16
Weighted Average	0.61	0.33	0.26

Table 4.1: *Evaluation results of RFC*

V. CONCLUSION

The results suggests that the use of irrelevant data such as product_id can lead to our RFC model learning irrelevant patterns, however focusing only on feature importance does not guarantee optimal results as found when unimportant

features had been removed. It is important to understand the features being fed into the model and how best utilize them. By combining all our text data, we were able to provide more context to our model for each listing instance which allowed for an efficient learning representation even.

Although much of the text was imbalanced which many columns such as room, craft_type, recipient, material, occasion, and holiday, having most empty rows. The data can be assumed to missing at random, in that not every listing will be related to each column. It is missing randomly related to the value of the observation but not randomly as related to other variables. The RFC was a suitable model choice for this dataset as its ability in handling imbalanced, noisy data, and large datasets.

Future work could explore other classifiers to further improve the prediction accuracy. Deep learning methods may explore additional features that could be extracting from image data available which could be incorporated into the model.

VI. REFERENCES

[1] A. Paul, D. P. Mukherjee, P. Das, A. Gangopadhyay, A. R. Chintha, and S. Kundu, 'Improved Random Forest for Classification', *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 4012–4024, Aug. 2018, doi: 10.1109/TIP.2018.2834830.

[2] L. Breiman, 'Random Forests', *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[3] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, 'How Many Trees in a Random Forest?', in *Machine Learning and Data Mining in Pattern Recognition*, Berlin, Heidelberg, 2012, pp. 154–168.

[4] N. Jalal, A. Mehmood, G. S. Choi, and I. Ashraf, 'A novel improved random forest for text classification using feature ranking and optimal number of trees', *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, Part A, pp. 2733–2742, Jun. 2022, doi: 10.1016/j.jksuci.2022.03.012.

[5] S. Maruf, K. Javed, and H. A. Babri, 'Improving Text Classification Performance with Random Forests-Based Feature Selection', *Arabian Journal for Science and Engineering*, vol. 41, no. 3, pp. 951–964, Mar. 2016, doi: 10.1007/s13369-015-1945-x.

[6] C. Strobl, A. Boulesteix, A. Zeileis, and T. Hothorn, 'Bias in random forest variable importance measures: illustrations, sources and a solution', *BMC Bioinformatics*, Jan. 2007, doi: doi:10.1186/1471-2105-8-25.