Product Categorization using Machine Learning

Jaslyn Samantha D'Souza
Student Number: 23262425
Dublin City University

***Abstract*—In this paper, we conduct a series of tests to determine which machine learning models are most effective in predicting product features, especially when predicting multi-class problems. The main objective of our study, which we conducted on a sizable dataset from Etsy with over 200,000 entries, is to predict four distinct categories: top category id, bottom category id, primary color id, and secondary color id.**
The core principle of this research is to determine which features—text, image, or both—are most effective in simultaneously predicting many categories. Initially, we use a text-based model to classify products based on textual attributes using Random Forest classification and TF-IDF vectorization. Then we experiment with an image-based model that uses Convolutional Neural Networks (CNN) to categorize product images by extracting hierarchical characteristics. Finally, we develop a multi-modal ResNet50 model that incorporates both text and image inputs to improve categorization accuracy. While other models were considered, Random Forest was ultimately chosen due to its ease of implementation, ability to predict all 4 classes at a time, and most importantly, computational efficiency.

Key Words: Machine Learning, Convolutional Neural Network (CNN), ResNet50, Random Forest

I. Introduction

In the rapidly growing world of e-commerce, effective product categorization serves as an important tool that can help enhance user experience, facilitate product discovery, and optimize business operations. The ability to accurately categorize products not only improves navigation and search functionality on online platforms but also enables targeted marketing campaigns, personalized recommendations, and efficient inventory management. However, manually assigning categories to a vast variety of products can be a labor-intensive, error-prone, and time-consuming task, especially as the volume of available products continues to grow exponentially by the minute.

The rise of big data and advancements in machine learning offer a more automated and potentially more accurate approach at classifying products on a large scale. By making use of the abundance of product data present in textual descriptions and images, machine learning models can learn patterns and associations to automatically assign relevant categories to products.

The main aim of this research is to compare the effectiveness of different approaches to classify and categorize products, considering both textual and visual properties of product data. We analyze a comprehensive dataset taken from Etsy.com that contains both textual descriptions as well as images of diverse products. With the help of this dataset, we develop and evaluate multiple machine learning models to predict product categories, including top-level categories, subcategories, and color attributes.

Our study comprises of three main machine learning based methodologies for automated product categorization. These are:

A. Text-Based Approach:

The text-based approach focuses solely on textual features extracted from product descriptions, titles, and tags. We preprocess the text data, apply TF-IDF vectorization to convert textual features into numerical representations, and utilize ensemble learning techniques such as Random Forest classification to predict product categories

B. Image-Based Approach:

The Image-based approach makes use of a simple Convolutional Neural Networks (CNN's) to extract hierarchical features from product images. After preprocessing and resizing product images, we feed them into a CNN architecture, which learns discriminative features that are subsequently used for product categorization.

C. Multi-Modal Approach:

A multi-modal approach combines both textual and visual attributes of product data. By integrating text and image inputs into a unified model architecture, we aim to harness the complementary nature of these features, potentially improving categorization accuracy and robustness.

Through extensive experimentation and evaluation on training and validation datasets, we assess the performance of each approach using metrics such as F1 score, loss, accuracy, and computational efficiency, along with analyzing the predictions they make on the test dataset. Additionally, we deploy the final model to make predictions on the whole dataset with the help of batch processing. This approach iterates through the data in smaller chunks in order to reduce memory usage.

II. Literature Review

In this section, we will analyze some well covered studies on various topics that are aspects of our research question:

A. Text-Based Approaches:

Nguyen et al. [1] conducted a review of machine learning techniques for product categorization, highlighting the significance of methods like decision trees and Support Vector Machines (SVMs). However, these traditional techniques often lack the ability to capture complex relationships in textual data. On the contrary, Wang et al. [2] proposed a deep learning-based approach, using architectures such as Convolutional

Neural Networks (CNN's) and Recurrent Neural Networks (RNN's) for hierarchical feature extraction. Despite their potential, training deep neural networks can be computationally intensive and may require extensive hyper parameter tuning.

B. Image-Based and Multi-Modal Approaches:

Simonyan and Zisserman [3] introduced the VGG architecture, a deep CNN widely used widely for image recognition tasks in today's world. However, deep neural networks like VGG may suffer from problems like overfitting especially on smaller datasets. Zhang et al. [4] proposed a multi-modal deep learning architecture for video classification, integrating textual and visual features. While effective, such models often require large amounts of labeled data and complex training procedures.

C. Scalability and Efficiency:

Li et al. [5] investigated batch processing and parallelization techniques for large-scale product categorization tasks. They observed that while batch processing can improve efficiency by processing multiple data samples simultaneously, it may also introduce challenges such as potential synchronization issues in distributed environments [5]. However, they can still be used for maximizing computational resources and reducing processing times for efficient model training and inference.

D. Random Forest for Text Features:

Yu et al. [6] proposed a Random Forest-based approach for product categorization. In addition to standard text features, their approach incorporated domain-specific features such as brand names, product types, and key terms, which they extracted using natural language processing (NLP) techniques. The results obtained show that the proposed approach outperforms traditional methods in terms of categorization accuracy and efficiency. However, domain-specific features may require careful preprocessing and extraction techniques.

Roberson [7] in their research, focused on investigating the use of machine learning algorithms, including Support Vector Machines (SVMs), Logistic Regression and Naive Bayes for automatic product categorization. These models were run on textual data. The results obtained show that traditional machine learning algorithms may struggle in capturing complex relationships in textual data and may require extensive manual feature engineering.

Oancea [8] explored the application of supervised machine learning algorithms, including Random Forests, for automatic product classification. They too implemented the model on text features. But this research was centered around price and economic analysis to achieve accurate classification in topics such as price index calculations.

Researchers have explored various avenues to enhance categorization accuracy. Additionally, scalability and efficiency considerations have led to investigations into batch processing and parallelization techniques, aiming to optimize computational resources for large-scale tasks.

In context to the above studies, our work aims to build upon these insights by using a Random Forest model for multi-level product categorization as a baseline, and also experimenting with models that will take image features into consideration as well to enhance classification accuracy.

III. Data Collection and Preprocessing

We used the Etsy products dataset to carry out our experiments. The dataset is divided into training and testing subsets having 229,624 and 25,514 records respectively. Each subset had various records stored in the .parquet format. Since we used the local jupyter environment to run our models, we could read all the parquet files directly into their respective dataframes.

On analyzing the data, we observed that:

- The data seemed to have no missing values.
- The categorical variables (such as type, occasion etc.) had a lot of empty '' values.
- The dataset showed signs of being imbalanced (for instance, more proportion of home-and-living products than pet-supplies; more of type 'physical' than 'download')

These factors could have an impact on the overall performance of our models.

After loading and understanding the data, we moved onto the preprocessing steps. We implemented different preprocessing steps for each model based on the input it was going to take. In general, the preprocessing methodologies looked as follows:

A. Text Preprocessing - steps to prepare textual data :

1) Text Cleaning: We considered the columns 'titles', 'descriptions', and 'tags' as our textual data. After integrating them together, it is subjected to cleaning processes to remove noise and standardize the text. This involves steps such as:

- Lowercasing: Converting all text to lowercase to ensure consistency.
- Removing Special Characters: Eliminating non-alphanumeric characters, punctuation, and symbols that do not contribute to the semantic meaning of the text.
- Handling Numbers: Removing or replacing numerical values, as they may not be relevant for text analysis tasks.

2) Tokenization: The cleaned text is tokenized into individual words or tokens to facilitate further analysis. This involves breaking down the text into smaller units, such as words or subwords. The word-tokenize function from the NLTK library was utilized for this purpose.

3) Removal of Stopwords and Punctuation: Stopwords, which are common words that often do not carry significant semantic meaning (e.g., "and", "the", "is"), are removed from the text to reduce noise and improve the efficiency of text analysis algorithms. This was achieved by using NLTK's built-in stopwords corpus.

4) Stemming: Stemming, which is a text normalization technique, is applied to reduce words to their base or root forms. This helps in standardizing the vocabulary and improving the effectiveness of text analysis tasks.

5) Vectorization: The pre-processed text data is then converted into numerical representations. We have used the TF-IDF (Term Frequency-Inverse Document Frequency) technique. This converts the text data into a format suitable for machine learning algorithms.

B. Image Preprocessing - steps to prepare image data:

We carried out preprocessing steps such as resizing and normalization to standardize the images and prepare them for analysis using computer vision techniques. The Image data was resized to a standardized dimension to ensure compatibility with the model architecture. This involved scaling images to a predefined size (i.e., 224x224 pixels in our case) with the help of image processing libraries. Resizing images to a uniform size is crucial for consistency in model input and computational efficiency during training and inference.

C. Data Integration - for the hybrid model:

The pre-processed text and image data are integrated into a unified dataset for subsequent analysis and modeling tasks. By implementing these preprocessing steps for both text and image data, the dataset was appropriately formatted and prepared for subsequent model development and training.

IV. Model Development

The development stages of each model are as follows:

A. Text-based Model using Random Forest:

The text-based model employs a combination of TF-IDF vectorization and Random Forest classification to categorize products based on textual attributes. The architecture of the model involves the following steps:

1) TF-IDF Vectorization: Text data, including product titles, descriptions, and tags, after preprocessing, undergo TF-IDF (Term Frequency-Inverse Document Frequency) vectorization. This process involves transforming text documents into a numerical matrix, where each row represents a document, and each column represents a unique word in the corpus [9]. TF-IDF weighting is applied to each term to emphasize its importance relative to the document and the entire corpus. TF-IDF accounts for both the frequency of a term in a document and its rarity across all documents, capturing its significance in categorization.

2) Random Forest Classifier: The TF-IDF vectors are then fed into a Random Forest classifier, a robust ensemble learning algorithm capable of handling multi-class classification tasks. Random Forest constructs multiple decision trees during training and aggregates their predictions to make final classifications. By leveraging Random Forest's ensemble of decision trees, the model can predict multiple category labels for each product instance, accommodating scenarios where products belong to multiple categories simultaneously.

B. Image-based Model using Convolutional Neural Networks (CNN):

The image-based model uses Convolutional Neural Networks (CNNs) which is a class of deep learning models well-suited for processing visual data. The architecture of the CNN model comprises of:

1) Convolutional Layers: Convolutional layers serve as feature extractors, applying learnable filters to input images to detect patterns and features at different spatial hierarchies. These layers capture low-level features such as edges and textures in initial layers and progressively learn higher-level representations in deeper layers.

2) Pooling Layers: Pooling layers help in reducing spatial dimensions while retaining important features [10]. Common pooling operations include max-pooling, which selects the maximum value within each pooling region, and average pooling, which computes the average value.

The CNN model learns hierarchical representations of image features through successive convolutional and pooling layers, enabling it to discern relevant patterns for product categorization.

C. Multi-Modal Model combining Text and Image Features:

The hybrid model integrates both text and image inputs to improve categorization accuracy by leveraging complementary information from multiple properties. The architecture of the multi-modal model involves:

1) Textual Feature Extraction: The textual data is preprocessed and transformed into numerical features using TF-IDF vectorization, similar to the text-based model. These features capture the semantic information present in product titles, descriptions, and tags.

2) Image Feature Extraction: Product images are processed using a pre-trained ResNet (Residual Neural Network) architecture. ResNet is chosen for its effectiveness in learning hierarchical representations of visual features, even in very deep networks [11]. By leveraging a pre-trained ResNet model, the multi-modal approach can extract meaningful image features without the need for extensive training on large image datasets.

3) Integration of Text and Image Features: The extracted text and image features are combined to form a unified feature representation. This combined feature vector captures both textual and visual information, providing a comprehensive representation of each product instance.

By integrating information from both text and image properties, the hybrid model enhances categorization accuracy by capturing complementary cues from different data sources.

4) Classification Layer: Following the integration of text and image features, the combined feature representation is fed into a classification layer, which predicts the product categories. This classification layer typically consists of one or more fully connected layers followed by a softmax activation function, which outputs probability distributions over the categories [12]. The model assigns each product instance to one or more categories based on the predicted probabilities.

V. Training and Evaluation

The training and evaluation steps followed for each model are as follows:

A. Text-based Model using Random Forest (the final model):

1) Training Process: The TF-IDF vectorizer is fitted on the training text data to learn the vocabulary and compute the IDF weights. The TF-IDF-transformed features are then used to train a Random Forest classifier. Batch processing is employed to handle the large volume of text data efficiently (batchsize is set to 20,000). The Random Forest model is trained using ensemble learning, where multiple decision trees are constructed and aggregated to make predictions.

2) Evaluation: The trained model is evaluated on both the training and validation sets. Evaluation metrics such as accuracy, precision, and recall are computed to assess classification performance.

TABLE I
Aggregate Classification Report

Class	Precision	Recall	F1-Score	Support
0	0.66	0.34	0.45	110
1	0.57	0.16	0.24	77
2	0.54	0.33	0.41	42
3	0.75	0.51	0.60	97
4	0.75	0.48	0.59	62
5	0.68	0.89	0.77	275
6	0.52	0.60	0.56	292
7	0.87	0.34	0.49	59
8	0.52	0.85	0.65	433
9	0.82	0.45	0.58	80
10	0.66	0.56	0.61	103
11	0.81	0.63	0.71	54
12	0.84	0.88	0.86	81
13	0.71	0.27	0.39	144
14	0.58	0.33	0.42	91
Accuracy			0.61	2000
Macro Avg	0.69	0.51	0.56	2000
Weighted Avg	0.64	0.61	0.59	2000

Table 1 represents the precision, recall, F1- score, and Support for the first 15 items in the dataset. The overall accuracy of the model is also computed.

The F1-Score's of the Random Forest model are:

- Average F1 Score: 0.246 (24%)
- F1 Score - Top Category: 0.47 (47%)
- F1 Score - Bottom Category: 0.097 (9%)
- F1 Score - Primary Color: 0.268 (26%)
- F1 Score - Secondary Color: 0.15 (15%)

B. Image-based Model using Convolutional Neural Networks (CNN):

1) Training Process: The model is trained on only a sample of the entire dataset (train:test = 100:10). Hyperparameters such as learning rate, batch size, and optimizer settings are tuned to optimize model performance. Due to computational constraints, the image-based model could only be properly trained to predict the top category id for each product image.

2) Evaluation: The loss and accuracy on running 10 epochs of the model are -

loss: 0.3933 ; accuracy: 095

The loss value of 0.3933 indicates the average loss incurred during the training process. A lower loss value suggests better performance in minimizing prediction errors. With an accuracy of 0.9500 (or 95%), the model correctly predicts the top category id for approximately 95% of the product images in the dataset.

Overall, achieving a loss value of 0.3933 and an accuracy of 0.9500 after 10 epochs of training indicates that the image-based model performs exceptionally well. However, considering the computational constraints and potential memory utilization, it may not be feasible to continue experimenting with this model. Also, the small sample size could be one explanation as to why the models accuracy is so high.

C. Hybrid Model combining Text and Image Features:

1) Training Process: The hybrid model combines textual and image features extracted from TF-IDF vectorization and ResNet, respectively. Fusion layers integrate the representations from both features before passing them through fully connected layers for classification. Hyperparameters for both text and image processing components are tuned to achieve optimal performance. The model is trained on only a sample of the entire dataset (train:test = 500:50).

2) Evaluation: In order to evaluate the model, we can examine the training and validation loss and accuracy values. The training loss values indicate the average loss incurred during the training process for each output category. Similarly, the training accuracy values represent the percentage of correctly predicted categories during training.

- For the top category output: The training loss is 784.1983, and the training accuracy is 0.0600.
- For the primary color output: The training loss is 1359.6978, and the training accuracy is 0.0100.
- For the secondary color output: The training loss is 1437.8594, and the training accuracy is 0.0200.

Validation Loss and Accuracy provide insight into the model's performance on unseen data. They indicate how well the model generalizes to new data. They are as follows:

- Validation Loss: 3581.755
- Validation Top Category Loss: 784.198
- Validation Primary Color Loss: 1359.697
- Validation Secondary Color Loss: 1437.859
- Validation Top Category Accuracy: 0.059
- Validation Primary Color Accuracy: 0.009
- Validation Secondary Color Accuracy: 0.019

The high values of training and validation loss suggest that the model is struggling to minimize the error between the

predicted and actual values. This could indicate that the model is not effectively capturing the underlying patterns in the data or that it is overfitting to the training data. The low accuracy values for all output categories suggests that the model may be making incorrect predictions.

VI. Predictions on Test Data

By applying the trained model to unseen data, we can evaluate its effectiveness in accurately categorizing products and assess its potential impact on practical applications in the real world, particularly in the field of e-commerce.

A. Text-Based Model using TF-IDF and Random Forest:

The predictions generated by the text-based model exhibit a satisfactory level of accuracy in categorizing products based on textual attributes extracted from product descriptions. Upon analysis, it is observed that the model effectively captures the semantic meaning of product descriptions and accurately assigns them to appropriate categories. However, there are instances where the model struggles with ambiguous descriptions, leading to misclassifications. Despite these challenges, the model demonstrates robust performance overall, showcasing its effectiveness in leveraging textual features for product categorization tasks.

TABLE II
Predictions on Test Data - Random Forest Model

Product ID	Predicted Top Category ID	Predicted Bottom Category ID	Predicted Primary Color ID	Predicted Secondary Color ID
661373440	3	2809	11	1
1501009290	6	249	14	14
1105447030	8	6	18	18
1140081090	8	1633	9	17
793448890	8	76	17	17

Table 3 showcases the predicted values against the ground truth. The model was tested out against part of the training data to compare the predicted values with the actual values. The results show that while there are some discrepancies, the model is also able to predict the right value at certain points

B. Image-Based Model using Convolutional Neural Networks (CNN):

The predictions made by the image-based model showcase its capability to categorize products based on visual features extracted from product images. Upon examination, it becomes apparent that the model does fairly well in recognizing visual patterns and structures, enabling accurate categorization of products. However, there are instances where the model may misclassify products with subtle visual differences or variations in lighting conditions. There may also be inherent bias present in the model leading to such high accuracy. Despite these limitations, the image-based model demonstrates good performance, highlighting its potential in being used to derive visual information for product categorization.

C. Hybrid Model combining Text and Image Features:

The predictions generated by the hybrid model underline its attempt to integrate textual and visual features for product categorization. While the concept of combining multiple modalities holds promise, the actual performance of the hybrid model falls short compared to individual text-based or image-based models. Despite efforts to leverage complementary information from both properties, the hybrid model demonstrates inferior performance, suggesting challenges in effectively integrating text and image features. Nevertheless, these findings shed light on the complexities involved in multimodal approaches and highlight the need for further refinement and optimization to unlock their full potential in product categorization tasks

VII. Findings and Discussions

A. Insights and Implications:

The analysis of model performance reveals several insights into the effectiveness of different approaches for product categorization:

1) Text-based models demonstrate good interpretability but may struggle with capturing nuanced semantic relationships in product descriptions.
2) Image-based models offer robust feature extraction from visual data but require substantial computational resources and may be prone to overfitting.
3) Hybrid models integrating both text and image features hold significant potential for enhancing categorization accuracy by harnessing the complementary strengths of each modality. However, this integration may introduce complexities and computational overhead, potentially impacting the efficiency and scalability of the model.

B. Recommendations and Future Work:

Based on the findings and limitations observed in the analysis, several recommendations and avenues for future research can be proposed:

1) Look into using alternative text representation techniques, such as word embeddings or contextualized embeddings (e.g., BERT), to capture more nuanced semantic relationships in product descriptions.
2) Explore transfer learning approaches for image-based models, leveraging pretrained CNN architectures to improve categorization performance and mitigate overfitting.
3) Experiment with different fusion strategies for hybrid models, including attention mechanisms or graph-based fusion, to better integrate information from text and image modalities.
4) Collect and annotate larger and more diverse datasets to train and evaluate the models, allowing for better generalization and robustness to variations in product categories and attributes.

Actual Top ID	Predicted Top ID	Actual Bottom ID	Predicted Bottom ID	Actual Primary ID	Predicted Primary ID	Actual Secondary ID	Predicted Secondary ID
13	6	1583	1957	13	1	19	16
7	8	12193	1619	1	1	1	1
6	6	7041	161	3	7	7	7
6	6	6240	6240	2	16	4	17
5	5	1827	462	1	9	17	17

TABLE III
PREDICTED VALUES VS ACTUAL VALUES - USING VALIDATION DATA

TABLE IV
PREDICTIONS ON TEST DATA - IMAGE ONLY MODEL (CNN)

Product ID	Predicted Top Category ID
726273220	10
546069580	5
1639862700	7
1089529420	5
1126213770	6

TABLE V
PREDICTIONS ON TEST DATA - HYBRID MODEL (RESNET50)

Product ID	Top Category	Primary Color	Secondary Color
726273220	12	13	18
546069580	12	13	18
1639862700	12	13	18
1089529420	12	13	18
1126213770	12	13	7

VIII. CONCLUSION

In conclusion, this study explored the problem of automated product categorization in e-commerce, employing diverse machine learning models to address this critical task. Through thorough analysis and evaluation, we have obtained a better understanding about the effectiveness of different approaches in categorizing products based on textual and visual attributes.

The text-based model, utilizing Random Forest, demonstrated solid performance in extracting semantic information from product descriptions, albeit with occasional misclassifications. The image-based model, using Convolutional Neural Networks (CNNs), exhibited proficiency in recognizing visual patterns associated with one variable. The hybrid model, integrating both text and image features, emerged as a promising contender by connecting two different features to achieve superior categorization accuracy. Despite being the less accurate out of the three, the hybrid approach still holds potential for practical implementation in real-world e-commerce scenarios.

Accurate product categorization is essential for enhancing user experience, facilitating product search and recommendation systems, and optimizing inventory management and marketing strategies in the e-commerce world. The choice of model architecture to accomplish automated product categorization should be informed by factors such as the available computational resources, the nature of the input data (textual, visual, or a combination of both), and the desired trade-offs between accuracy, interpretability, and scalability.

REFERENCES

[1] N H Nguyen et al. “Product categorization: a review of machine learning techniques”. In: *Expert Systems with Applications* 109 (2018), pp. 31–49.

[2] Y Wang et al. “Product categorization based on deep learning”. In: *International Conference on Machine Learning*. 2019.

[3] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).

[4] H Zhang et al. “Attentional multimodal fusion for video classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019.

[5] M Li et al. “Efficient batch processing for large-scale product categorization”. In: *Journal of Parallel and Distributed Computing* 146 (2020), pp. 201–210.

[6] Xiaotian Yu et al. “Product Categorization Approach Based on Random Forest and Domain-Specific Features””. In: *IEEE Access* 8 (2020), pp. 121081–121090.

[7] Andrea Roberson. “Applying Machine Learning for Automatic Product Categorization”. In: *Journal of Official Statistics* 37.2 (2021), pp. 395–410.

[8] Bogdan Oancea. “Automatic Product Classification Using Supervised Machine Learning Algorithms in Price Statistics”. In: *Mathematics* 11.7 (2023), p. 1588.

[9] Gerard Salton, Anita Wong, and Chung-Shu Yang. “A vector space model for automatic indexing”. In: *Communications of the ACM* 18.11 (1975), pp. 613–620.

[10] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[11] Bougareche Samia, Zehani Soraya, and Mimi Malika. “Fashion images classification using machine learning, deep learning and transfer learning models”. In: *2022 7th International Conference on Image and Signal Processing and their Applications (ISPA)*. IEEE. 2022, pp. 1–5.

[12] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006.