Enhancing E-commerce Proficiency: A Machine Learning Approach to Product Characteristic Prediction

Pranav Lodh
MSc in Computing
pranav.lodh2@mail.dcu.ie
Dublin City University
Student Number: 23263834

***Abstract*—This research explores the significance of anticipating particular product characteristics in e-commerce and online platforms. The proposal suggests constructing a multiclass classifier that precisely forecasts attributes like top_category_id, bottom_category_id, primary_color_id, and secondary_color_id.The ultimate goal of the study is to develop a machine-learning model capable of swiftly and accurately predicting these attributes. The investigation scrutinises the methods of text preprocessing, feature extraction, and feature transformation utilising techniques such as count vectorization and Term Frequency-Inverse Document Frequency (TF-IDF). Multiple models like Multinomial NB , Support Vector Machines and ADABoost were used to predict the target variables and their performance was evaluated using F1 scores.**

I. Introduction

In ecommerce and online marketplaces, properly categorising products is a critical task. This involves carefully analysing a product's features and descriptions in order to determine the appropriate category. The analysis takes into account various factors including the product's characteristics, materials, intended use and function. Precise product category prediction can greatly enhance the online shopping experience by simplifying the process for shoppers to locate the products they desire.

Supervised learning models like Support Vector Machines (SVMs), probabilistic classifiers such as Multinomial Naive Bayes, and ensemble learning methods like AdaBoost are all powerful tools for classification and regression tasks. SVMs, for example, focus on finding the optimal hyperplane that separates data into distinct classes, while Multinomial Naive Bayes models feature distributions as multinomial distributions and assumes independence between features. AdaBoost iteratively combines multiple weak learners to create a strong learner, giving more weight to misclassified instances to improve generalization performance. These algorithms are widely used in various domains due to their effectiveness in handling different types of classification problems.In this paper, we will use these models with parameter tuning to predict the top_category_id, bottom_category_id , primary_color_id and secondary_color_id from the ecommerce dataset provided.

The objective of this task is to meticulously examine the training dataset to identify specific patterns and predict the top category ID, bottom category ID, primary and secondary color ID for an unseen product in the test dataset. The ultimate aim is to improve the F1 score for each class associated with every attribute.

II. Related Work

Machine learning algorithms have gained attention for automating product categorization in e-commerce.

This paper by [1] proposes an automatic system to categorize products on e-commerce sites. It uses deep learning models (deep belief nets and deep autoencoders) trained on product titles and descriptions to predict the most fitting category within a hierarchical structure. The system, designed to handle large datasets efficiently, achieved an impressive 81% accuracy in matching human-assigned categories (excluding "others"), demonstrating its potential to significantly reduce manual workload for e-commerce merchants.

[3] present a new method for e-commerce product category recognition in their paper titled "Category Recognition in E-commerce Using Sequence-to-Sequence Hierarchical Classification". This approach might differ from previous methods by using a sequence-to-sequence hierarchical classification model. This model likely analyzes product information (potentially including titles, descriptions, or other attributes) as a sequence and classifies it within a hierarchical category structure, potentially improving accuracy or handling complex category relationships compared to prior work.

The paper "Object detection using YOLO: challenges, architectural successors, datasets and applications" by [2] focuses on YOLO (You Only Look Once), a popular deep learning model for object detection. It discusses the challenges associated with YOLO, such as the trade-off between accuracy and speed compared to two-stage detectors. The paper also explores advancements in object detection by mentioning YOLO's architectural successors, which might address these limitations. Additionally, the paper highlights various datasets used for training object detection models and explores the diverse applications of YOLO in real-world scenarios.

code : https://gitlab.computing.dcu.ie/lodhp2/ca684$_{a}ssignment$

The paper by [4] investigates color attributes for object detection, focusing on their effectiveness in computer vision applications. The authors propose a novel method that utilizes color information to improve object detection accuracy. By analyzing various color attributes, such as hue, saturation, and brightness, they demonstrate the importance of considering color cues in object detection tasks. Through experiments conducted on benchmark datasets, the paper showcases the efficacy of their approach in accurately detecting objects in images. Overall, the study highlights the significance of leveraging color attributes for enhancing object detection performance in computer vision systems.

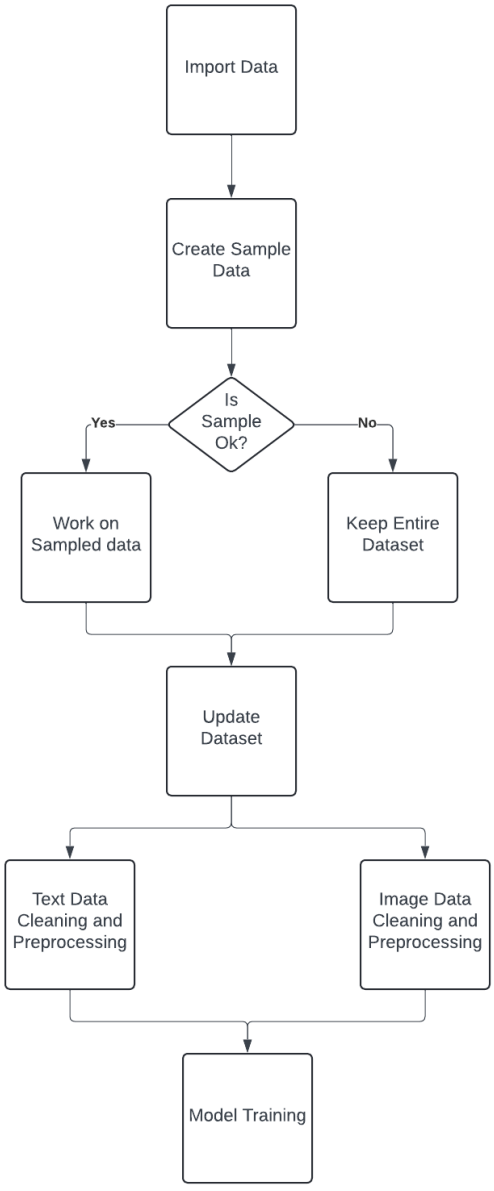

Fig. 1: Workflow Diagram

III. Methodology

A. Dataset Description

To validate our model and predictions, we used the dataset that was provided by Etsy. Etsy is a global online marketplace that enables individuals to buy and sell handmade, vintage and uniquely crafted items. Both training and test sets were provided by Etsy. The training datasets contain a total of 229,624 distinct products. Each product in the training set is linked to 26 features, such as product_id, title, description, tags, type, room, craft_type, recipient, material, occasion, holiday, art_subject, style, shape, pattern and the target variables like top_category_id, bottom_category_id, primary_color_id, secondary_color_id. We will test and compare our approach and models with a test dataset that has not been revealed yet.

B. Dataset Exploration and Analysis

We thoroughly explored and analyzed the provided dataset to identify patterns. The dataset included raw text data with special characters and emojis in columns such as titles, descriptions, and tags. Some columns had a missing percentage of more than 50%. The image data provided was in bytes format within the data frame, and there were class imbalances in the dataset regarding the target variables. The dataset contained 15 unique top categories, 2609 unique bottom categories, and 19 unique color IDs. Due to the dataset's vastness, we worked on a sample set to reduce computation costs. To do this, we utilized stratified sampling [5] and statistics to calculate a sample size of 15517 with a 99% confidence interval, 1% margin of error, and 50% population proportion. We verified whether the sampled data distribution represented the original dataset by conducting chi-square and Kolmogorov-Smirnov tests. However, when training for bottom category IDs, we used the full dataset as the sampled data would sometimes have bottom category IDs with only 1 or 2 rows, leading to an underfit model.

C. Data Preprocessing

Various preprocessing steps were performed on each target variable. To train top_category_id and bottom_category_id, we utilised the title, description, type, and tags columns from the dataset. These text columns were cleaned by removing special characters and numbers, eliminating non-English words and stopwords, and converting the text to lowercase. Additionally, some records in the 'type' column were missing. After thorough analysis of the 'type' column, we decided to impute using forward fill. We also removed columns which contained more than 50% missing value. To process images, we found that images contained a lot of noise in addition to the main object. Therefore, it was necessary to extract the object from the image. We used YOLOv8 to detect objects in the image and crop the image based on the bounding box. If more than one object was detected, we considered the object that occupied the most space in the image. If YOLO was unable to detect the object in some cases, we considered the entire image.We also considered text data columns like title,tags and description to extract colours from the text.

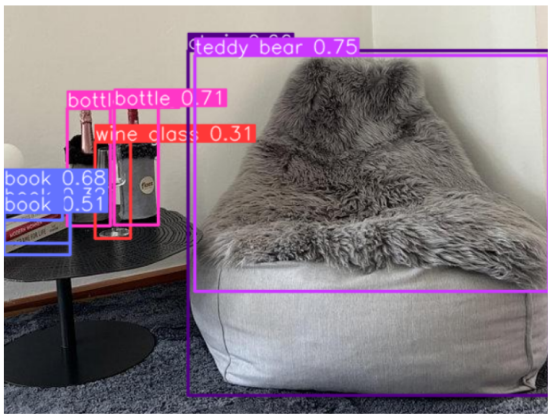

Fig. 2: YOLOv8 Objects detection

D. Feature Engineering

In order to train our model, we used a combination of different data features. Specifically, we integrated the title, description, type, and tags columns for both the top and bottom category IDs. To do this, we employed two different techniques: CountVectorizer and TFIDF transformer. When it came to analyzing images, we took a slightly different approach. We decided to use KMeans clustering, a popular unsupervised machine learning algorithm, in order to identify the primary and secondary color IDs for each image. We used n=2 for the clustering process. This allowed us to develop an RGB color palette for each image. Once we had the color palette for each image, we calculated the weighted average of the RGB color codes. This gave us a singular value that we could use to represent the main color of each image. By using this approach, we were able to extract valuable data from the images and incorporate it into our model training process.

E. Model Selection

In order to predict the target variables, we utilized a variety of models. For the top category ID, we experimented with Stochastic Gradient Descent, a Support Vector Machine with a learning rate of C=1, and a Multinomial Naive Bayes algorithm that utilized K-fold cross-validation , as well as GridSearch CV for hyperparameter tuning. When predicting the bottom category ID, we found that Multinomial Naive Bayes and Stochastic Gradient Descent were more efficient classification algorithms than traditional models like Logistic regression and Support Vector Machines as these algorithms were utilising high computation and training time and required powerful processing. For our Color ID Prediction model, we began with ADABoost as the base model, using rgb value as the feature. As we progressed in our analysis, we implemented an ensemble model. This approach involved training the model with ADABoost and rgb value, and then utilizing Multinomial NB to train the model with the colors in the text columns. This iterative technique allowed us to improve our predictions over time. Along with this we also utilised kaggle's TPU which has more computation power in order to train our model on large dataset.

F. Results

We considered F1 score as our evaluation metrics to get the performance of our model. Our model performed well to predict Top category ID and bottom Category ID while it struggled a lot while predicting the primary and secondary colour ID, even though we performed 15% better than our baseline model as we noticed that primary_color_id and secondary_color_id were not annotated properly.Table shows the results of our model in predicting each target variable.

Model	F1 Score
Multinomial Naive Bayes	0.66
Stochastic Gradient Descent	0.725
Support Vector Machine	**0.744**

(a) Evaluation Results for Top Category.

Model	F1 Score
Multinomial Naive Bayes	**0.46**
Stochastic Gradient Descent	0.44

(b) Evaluation Results for Bottom Category.

Model	Primary Colour ID	Secondary Colour ID
ADABoost	0.09	0.07
ADABoost with Multinomial NB	**0.24**	**0.20**

(c) Evaluation Results for primary and secondary color ID.

IV. Conclusion and Future Scope

In the research paper, various classification algorithms were employed to predict the target variables based on the dataset. Going ahead, it may be feasible to carry out further image processing techniques like removing the background and gathering RGB values. Additionally, the study might also utilize the YOLO Object detection findings to forecast top and bottom category id through word2vec or other word embedding techniques. By implementing these methods, the accuracy of the predictions can be improved, and the overall results can be enhanced, providing better insights into the data analysis.

References

[1] Ali Cevahir and Koji Murakami. Large-scale multi-class and hierarchical product categorization for an e-commerce giant. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 525–535, Osaka, Japan, 2016. COLING 2016 Organizing Committee.

[2] T. Diwan, G. Anirudh, and J.V. Tembhurne. Object detection using yolo: challenges, architectural successors, datasets and applications. *Multimed Tools Appl*, 82:9243–9275, 2023.

[3] I. Hasson, S. Novgorodov, G. Fuchs, and Y. Acriche. Category recognition in e-commerce using sequence-to-sequence hierarchical classification. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 902–905, March 2021.

[4] F.S. Khan, R.M. Anwer, J. Van De Weijer, A.D. Bagdanov, M. Vanrell, and A.M. Lopez. Color attributes for object detection. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3306–3313. IEEE, June 2012.

[5] X. Meng. Scalable simple random sampling and stratified sampling. In *International conference on machine learning*, pages 531–539. PMLR, May 2013.