Product Classification using Text and Multimodal Embedding extraction

Ishrat Fatima Syed, School Of Computing, Dublin City University

***Abstract*—This research project aims to predict product categories, specifically top category, bottom category, and color. To achieve this goal, we are exploring both unimodal and multimodal approaches. In the unimodal approach, we consider text and images separately, while in the multimodal approach, we utilize both text and image embeddings. Our focus is on developing accurate and efficient models that can effectively predict product categories using these approaches. We believe that our research will contribute to the development of effective methods for predicting product categories, which has significant applications in various industries.**

I. Introduction

In the realm of e-commerce, it is crucial for brands to deliver a seamless user experience to thrive. Accurate product classification and categorization are key components in achieving this goal. Each product is associated with metadata, including its title, image, description, top and bottom categories, color, and other attributes provided by the seller. Precise categorization of products enables customers to easily discover similar items, search for products using common keywords, and enjoy an organized and efficient shopping experience. This leads to higher customer satisfaction and loyalty to the brand. However, manually categorizing products for large-scale deployment is impractical, highlighting the need for automatic categorization systems [1]. Automatic product categorization reduces manual work, saves time, and lowers costs. It is essential because manual labeling can be unreliable and potentially impact sales opportunities.

Developing an automatic system for product categorization is challenging, and extensive research has been conducted in this field. Past studies have mainly treated this issue as a text classification problem [2] [3]. However, some researchers have also utilized a multi-modal approach combining text and image to categorize products [1] [4]. The unimodal approach basically consisted of either textual information or images of the product to build a classifier to categorize products, whereas the multimodal approach attempts to build a classifier that combines multiple modalities.

Transfer learning is a popular technique in computer vision and natural language processing. Its ability to develop highly accurate and time-efficient models has contributed to its increasing popularity. This is achieved through the use of pre-trained models. Pre-trained models are already trained on large benchmark datasets to solve similar tasks. Using these pre-trained models can prove to be efficient, as it saves time and reduces the cost of training a model from scratch.

Our paper presents two techniques namely unimodal and multi-modal, for categorizing products based on their top category, bottom category, and color. To achieve this, we have used text and images of the products. We have also extracted text and image embeddings using pre-trained models and combined them into a single vector. This concatenated embedding was then used as input for our classification model.

II. Literature Review

In a study [5] the authors explore lightweight methods for large-scale product categorization, focusing on their effectiveness and applicability in the context of e-commerce platforms. The authors comprehensively analyze various methods, including text-based, image-based, and hybrid approaches. They evaluate the advantages and limitations of each method and consider their suitability for different e-commerce scenarios.

One popular approach they mentioned for product categorization was using text-based methods. Text-based methods involve analyzing product titles, descriptions, and other textual attributes to determine the appropriate category. One example of a text-based method is the use of keyword-based classifiers. These classifiers rely on predefined lists of keywords and phrases that are associated with specific product categories.

Another approach they suggested was a hybrid approach that combines text-based and image-based methods can also be used for product categorization. This approach involves using a product's textual and visual features to determine its category. One example of a hybrid approach is the use of a text-based classifier trained on textual attributes, such as product titles and descriptions, and an image-based classifier trained on product images.

Their research has studied various lightweight methods for large-scale product categorization, ranging from simple keyword-based classifiers to more complex machine learning and deep learning algorithms. The choice of method depends on the specific requirements of the e-commerce platform, such as the size of the inventory and the available computational resources. Text-based methods are simple and easy to implement but can be limited in their accuracy. Image-based methods can be powerful but require significant computational resources. Hybrid approaches that combine textual and visual features can provide a good balance between accuracy and efficiency.

As part of the SIGIR 2020 e-commerce workshop, Rakuten France provided a dataset of multi-modal product data that included product images, French titles, and detailed descriptions. The task was to predict 27 category labels across four major genres, namely books, children, households, and entertainment. Many teams and researchers participated in the

competition including [1], [6] and [7], and most proposed fine-tuning pre-trained text and image models as feature extractors, followed by bimodal fusion mechanisms to combine predictions.

The most commonly used pre-trained models were BERT for text feature extraction and ResNet for image feature extraction. BERT models are open-source language models that are fine-tuned for specific goals and have been used in various ways to improve language processing tasks.

In a study [8], the pre-trained French language models were fine-tuned separately for each textual modality, while the SE-ResNeXt-50 model was used to extract features for image products. The extracted features were then hierarchically fused using different fusion strategies to make the final prediction. Other pre-trained models used in this field include FlauBERT and CamemBERT.

III. Methodology

In this section, we outline the techniques employed in constructing our solution, along with the reasoning behind the choices made. The fundamental principles guiding our model design were portability and scalability. We approached the task of classification not as a mere competition to achieve the highest accuracy or F1 score, but rather as a pursuit of a robust solution that could be deployed in a production environment to provide value to Etsy. We recognized that a model that performs well in a research or experimental setting may not necessarily translate seamlessly into a production environment. Therefore, our methodology was aimed at building a solution that not only achieves high accuracy but also can be efficiently deployed in a real-world production system. Therefore, we placed significant emphasis on the portability of our approach and the potential for future improvements. In the following sections, we provide a detailed description of our methodology and the approach we adopted, highlighting its strengths and potential for scalability.

Understanding the data

Understanding the data is a crucial step in our methodology, as it forms the foundation for building an effective classification solution. We employed a comprehensive data analysis process to gain insights into the characteristics of the dataset and make informed decisions about our approach.

In our classification task, we were tasked with categorizing products into three distinct categories: top category, bottom category, and color. The dataset comprised of text features that described the products, as well as actual product images. The top category consisted of 15 unique categories, with the distribution of products as depicted in Figure 1.

The bottom category, on the other hand, exhibited a hierarchical structure, with over 2700 categories to choose from. To gain further insights into the bottom category and its hierarchical structure, we utilized a treemap visualization, as depicted in Figure 3. The treemap provided a graphical representation of the taxonomy and structure of the bottom category in the dataset and the hierarchical relationships between

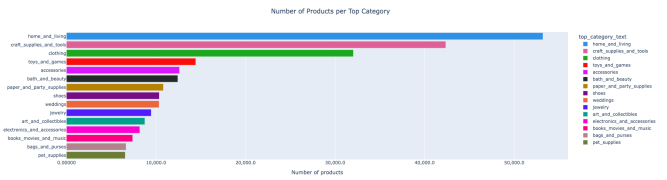

Figure 1: Distribution of number of products in top category

the categories, with larger rectangles representing higher-level categories and smaller rectangles representing subcategories. The size of the rectangles was proportional to the number of products within each category, providing a visual indication of the distribution of products across different categories as shown in Figure 2.

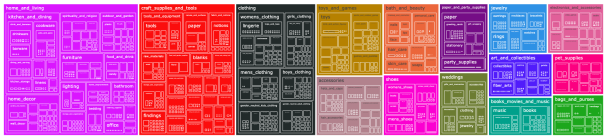

Figure 2: Treemap showing number of categories in each bottom category level

Lastly, the color category encompassed 20 different values, with their distribution illustrated in Figure 3.

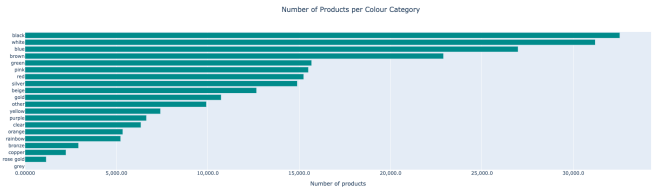

Figure 3: Distribution of number of products for available colors

This multi-class classification problem required careful analysis and modeling to accurately categorize products into the appropriate categories. The complexity of the task was amplified by the large number of categories in the bottom category hierarchy, which necessitated a robust and scalable approach. Additionally, the textual features and product images provided diverse and rich information that needed to be effectively utilized to achieve accurate classification results.

Feature Selection and Data Preparation:

Before proceeding with the development of our model, we conducted an exploration of the available product features. We observed that although there were several textual fields, many of them contained sparse data with a significant number of null values. Therefore, we made a decision to focus on the three text features that were most informative, namely product title, product description, and product tags, along with the visual feature of product images.

Table I: Text Features and Visual Features in the Data

Text Features	Visual Features
Product Title	Product Images
Product Description	
Product tags	

By carefully considering the available data, we aimed to make the best use of the available information while accounting for the limitations in terms of sparse and null data. We recognized that the product title, description, and tags could provide valuable information for classification tasks, while the product images could potentially provide additional visual cues for accurate classification. This approach allowed us to leverage the available textual and visual features effectively in the development of our model

To begin our approach, we needed to address the issue of vectorizing our features, as both text and image data cannot be used in their raw form by most machine learning models. We divided our modeling and data preparation strategy into two groups: 1) working with text data using a unimodal approach, and 2) working with both text and image data using a multimodal approach.

Although a multimodal approach logically seemed like the best option, we were aware that working with large vocabularies of text and images could be computationally intensive and may require GPU-based processing. Therefore, we decided to start with a quick and interpretable prototype in the first iteration, focusing on the rich text features in the data. By applying clever natural language processing (NLP) techniques, we aimed to extract meaningful insights from the text data, which would serve as the foundation for building a robust multimodal solution. Thus, we designed data preparation techniques that fit our specific needs, taking into consideration the richness of the text features in the data.

In the data preparation pipeline for the text data, we implemented a series of standard text pre-processing steps to ensure the integrity and relevance of the features. These steps included removing stop words (such as ”the”, ”a”, etc.), special characters, HTML, URLs, ASCII codes, emojis, and other non-essential elements to reduce the size of the vocabulary. We also performed tokenization, which involved splitting a string into an array of individual words or ”tokens”. Furthermore, we applied lemmatization to the tokens, which involves reducing words to their base or root form to capture their core meaning. These comprehensive pre-processing steps were vital in preparing the text data for subsequent numeric feature extraction and modeling, ensuring the quality and accuracy of our analysis. Note that we also use these preprocessed text features as input to our multimodal model data pipeline.

In our multi-modal track of data preparation, we aimed to leverage the combined cleaned text from the title, tags, and description fields. After extensive research and experimentation, we opted for transfer learning using neural networks as our approach. We utilized pre-trained models for both image and text embeddings, which allowed us to convert our raw features into meaningful representations for further classification.

To determine the most suitable models for our task, we conducted a thorough comparison of various pre-trained models, including CLIP, ResNet, and EfficientNet. Our selection criteria considered both model performance and computational cost, striking a balance between the two. Ultimately, we decided to use distilBERT [9] for text embeddings and MobileNet [10] for image embeddings. These choices were based on their demonstrated performance and computational efficiency, making them well-suited for our multi-modal classification task.

We saved the generated embeddings (see figure 4) in three forms for flexibility in our modeling techniques:

Text Embeddings: We saved the embeddings generated from the distilBERT model for text separately. The size of these embeddings would depend on the specific configuration and architecture of the distilBERT model, but it is typically 768-dimensional for each token in the input text.

Image Embeddings: We saved the embeddings generated from the MobileNet model for images separately. The size of these embeddings would depend on the specific version of MobileNet used, but it could be 1024-dimensional for the original MobileNet.

Concatenated Embeddings: We also concatenated the text and image embeddings using np.concatenate(axis=0) to obtain combined embeddings. The size of the concatenated embeddings would be the sum of the sizes of the text and image embeddings. For example, if the text embeddings are 768-dimensional and the image embeddings are 1024-dimensional, the concatenated embeddings would be of size 1792-dimensional (i.e., 768 + 1024).

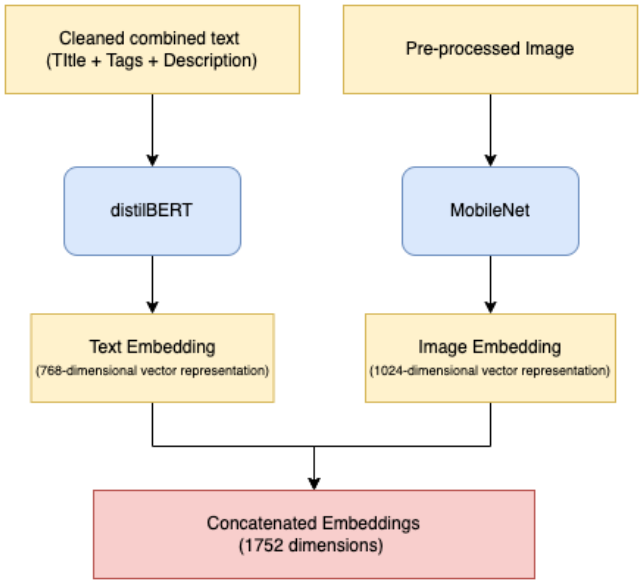

Figure 4: Strategy for building Joint Embedding

By saving the embeddings in these different forms, we have the flexibility to use them separately or combined in our

modeling techniques, depending on the specific requirements of our research or application

Modeling Technique and Experiments:

a) ***Top Category Classification****:* Our initial task was to predict the top category for each product, and we started with a unimodal approach by utilizing preprocessed text data as described in the previous section. To extract meaningful information from the text data, we performed feature engineering using the RAKE algorithm from the Natural Language Toolkit. This algorithm identifies important phrases from the product title and description by analyzing word co-occurrence patterns.

To create a comprehensive representation of the product's features, we combined the extracted keywords with the product's tags, which contained valuable information about the product's category and subcategory. This combined approach of tags and keywords helped us in our analysis. We then experimented with various algorithms, including tree-based models like LightGBM, XGBoost, etc., to predict the top category for each product.

In our multi-model approach, we also attempted to improve the model's performance by using concatenated embeddings of text and image, as depicted in Figure 4. However, we faced challenges due to the size and complexity of our dataset, and many of these approaches required significant computing resources and were not easily scalable for large datasets. The time-consuming nature of model training and limited resources led us to suboptimal F1 scores in the multi-modal approach.

After conducting extensive research, we decided to use LightGBM as our classification model due to its significantly faster computational speed and lower memory consumption compared to other popular algorithms like XGBoost and SGB [11]. Classifying the top category is a relatively simpler task, and opting for a more efficient and less complex model like LightGBM was a practical choice, given our limited resources and dataset size.

b) ***Bottom Category Classification****:* The task of predicting the bottom category for each product presented several challenges due to the vast number of categories (over 2700) combined with a limited number of samples available for each class (11-20 samples). To address this issue, we experimented with various approaches.

In our first two experiments, we attempted a multi-modal model by using concatenated embeddings generated from pre-trained models such as DistilBERT and MobileNet, as well as the Hugging Face implementation of Open AI's CLIP (Contrastive Language-Image Pre-Training). However, despite the potential of these complex models, we encountered challenges such as the suboptimal results and the high time complexity of building these models, which made it difficult to run multiple iterations and tune the models efficiently.

As a result, we opted for a simpler approach for our third experiment. Given the high number of classes, we decided to use the K-Nearest Neighbors (KNN) algorithm in combination with the TF-IDF vectorizer to extract text features from the tags and keywords associated with each product. Using KNN with N=5, we achieved an accuracy of approximately 55pc. While this accuracy may not be ideal, it was the best we could achieve with the available computational resources.

Furthermore, we explored the possibility of using hierarchical classification [12] to break down the problem into smaller subproblems. Our exploratory data analysis (EDA) revealed the presence of a definite hierarchy in the bottom category, with multiple levels of depth, as shown in Figure 2. We organized the categories into a hierarchy with up to 7 levels (see Figure 5). Notably, levels 3 and 4 had over 900 classes or subcategories, making it a challenging task to handle the higher number of classes in these levels.

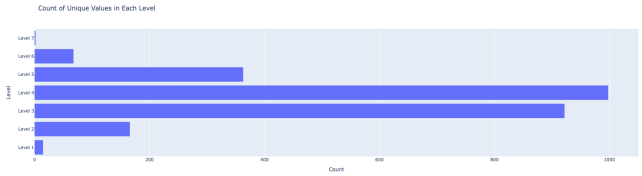

Figure 5: Number of classes in each level

Although we were unable to fully explore the potential of hierarchical classification due to time constraints in our project, it presented a promising approach that could be further investigated in future work. In summary, predicting the bottom category proved to be a challenging task, but our experiments highlight the importance of exploring different techniques and the potential benefits of using hierarchical classification to simplify complex problems.

c) ***Colour Classification****:* Predicting the color of products proved to be a challenging task, primarily due to the highly imbalanced dataset and a limited number of samples for certain colors, such as gray (see Figure 3). Our initial approach involved using a model combining text and image embedding. However, the results were unsatisfactory, so we built another model using raw images and a ResNet50 pre-trained model to extract features and predict color. This approach resulted in an accuracy of 52%, which was better than our initial model, but still not ideal.

We performed feature engineering to improve accuracy by extracting color names from the product's title, description, and tags. We extracted all standard color names from these fields and combined them with tags to train a LightGBM model. This approach achieved an accuracy of 55%, further improving the results.

Despite our efforts, we realized that none of the individual models we used could produce satisfactory results on their own. Therefore, we decided to build an ensemble of weak and diverse learners, which helped us to increase the accuracy to a reasonable level. Our ensemble approach consisted of two models, namely LightGBM and Neural Network built on concatenated embeddings. We utilized the predictions from LightGBM when the color was explicitly stated in the title, tags, or description of the product. However, when color was not mentioned in any of these fields, we employed the

model based on concatenated embeddings to predict the color category.

Unfortunately, we could not explore other approaches due to time and resource constraints, but we believe there is room for further improvement in future work. Overall, predicting a product's color is a challenging task, but by combining feature engineering and ensemble learning, we achieved satisfactory results.

IV. Results and Discussion

Dataset:

The data used in our research project was sourced from Etsy - an American e-commerce company that specializes in handmade or vintage items and craft supplies. The items on this platform span a wide range of categories, including jewelry, bags, clothing, home decor, furniture, toys, art, craft supplies, and tools. Both training and test sets were provided, containing product information such as product id, image, image height, image width description, tags, top category text, bottom category text, color text, top category id, bottom category id, color id, type, craft type, room, pattern, shape, occasion, material, recipient, and holiday.

In the training set, labels for the top category, bottom category, and color were provided, while in the test set, labels were not provided. Therefore, all the evaluation metrics used in this research project were based on the training set.

Results:

The table below presents a comparison of different models used for a classification task. The table includes the models tried for each category, their corresponding F1 scores, and a brief explanation of the approach taken.

Table II: Model Performance for Classification Tasks

Category	Model	F1 Score
Top Category	LightGBM	0.84
	XGBoost	0.78
	Joint Embeddings	0.65
Bottom Category	KNN (k=5)	0.55
	Joint Embeddings	0.48
	CLIP (Join Embeddings)	0.32
Color Category	Ensemble (LightGBM, Embeddings)	0.56
	Joint Embeddings	0.43
	CLIP (Join Embeddings)	0.31

For the "Top Category", three models were tried, including "LightGBM" with the highest F1 score of 0.84, followed by "XGBoost" with an F1 score of 0.78, and "Joint Embeddings" with an F1 score of 0.65.

For the "Bottom Category", three models were also tried, including "KNN (k=5)" with an F1 score of 0.55, "Joint Embeddings" with an F1 score of 0.48, and "CLIP (Join Embeddings)" with the lowest F1 score of 0.32.

For the "Color Category", two models were tried, including an ensemble of "LightGBM", "NN (Neural Network)" with an F1 score of 0.56, and "Joint Embeddings" with an F1 score of 0.43. Additionally, "CLIP (Join Embeddings)" was also tried with an F1 score of 0.31.

We believe that we obtained better results with simpler models such as "LightGBM" and "KNN" because they were easier to train and optimize. "Joint Embeddings" required training and optimizing neural networks with features represented in a joint vector space, which was more complex and time-consuming.

However, we acknowledge that there is potential for further improvements in performance. By spending more time on error analysis, we can identify categories or product types where the neural network made mistakes and address those specific areas. Additionally, conducting multiple iterations of training with different architectures can help us extract the best performance from the features.

One limitation we faced was resource constraints, including time and computational resources. The process of training and optimizing neural network models can be time-consuming and resource-intensive, which has impacted our ability to fully optimize the models in this study. Despite these limitations, we believe that with additional resources and more in-depth analysis, we could potentially unlock better performance from the "Joint Embeddings" approach or other more complex models.

V. Future Scope

Optimization is an essential aspect of every machine learning model, and model improvement is a continual process. This project focuses on large-scale product categorization and explores various modeling techniques that can be applied in this domain. Our work demonstrates that several approaches can be taken to solve this problem. We have made use of the available dataset to the best of our abilities, incorporating both image and text features. However, due to resource and time limitations, we were unable to optimize the models we constructed. This project presents an opportunity for future research to refine and improve these models, further enhancing their efficacy in large-scale product categorization.

Moving forward, there are several suggestions for future improvements in our project. Firstly, an error analysis should be performed to identify specific classes in which the classification models performed poorly. This analysis can be used to augment the dataset with examples where the model is making frequent mistakes or to build ensembles with diverse classification strategies to correct the mistakes made by a single classifier.

Secondly, hierarchical classification could be implemented to address the challenge of the 2700+ classes in the bottom category. One option is to build seven different classifiers, one for each level in the bottom category. Alternatively, a neural network could be trained with seven output layers, with the output of each layer fed into the next layer for prediction.

Thirdly, clustering on embeddings could be performed using algorithms like k-means to group similar products together. The assigned cluster could then be used as a feature in the classification model, enabling it to perform better by leveraging similarities between products.

Finally, individual models could be built on text embeddings, color embeddings, and text+color embeddings, and an ensemble could be created by combining the outputs of these models. This would enable the model to take advantage of the strengths of each individual model to improve overall performance

VI. Conclusion

In this study, we investigated the effectiveness of both unimodal and multimodal approaches in predicting product categories. Through our experimentation with various models for each category, we found that simpler models like "LightGBM" and "KNN" performed better than more complex models like "Joint Embeddings" and "CLIP." However, it's important to note that we did not optimize the complex models extensively.

Our findings suggest that simpler models can be more effective in certain contexts, particularly where computational resources and time are limited. However, we also acknowledge that more complex models may have the potential for better performance with more in-depth analysis and greater resource allocation.

Overall, our research contributes to the development of methods for predicting product categories using both text and image embeddings. By identifying the strengths and limitations of different models and approaches, we can continue to refine and optimize these methods for use in various industries. Future research can build on these findings by exploring additional approaches and models and further investigating the potential for improving performance.

References

[1] Y. Bi, S. Wang, and Z. Fan, "A multimodal late fusion model for e-commerce product classification," *arXiv preprint arXiv:2008.06179*, 2020.

[2] Z. Kozareva, "Everyone likes shopping! multi-class product categorization for e-commerce," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1329–1333, 2015.

[3] D. Vandic, F. Frasincar, and U. Kaymak, "A framework for product description classification in e-commerce," *Journal of Web Engineering*, pp. 001–027, 2018.

[4] P. Wirojwatanakul and A. Wangperawong, "Multi-label product categorization using multi-modal fusion models," *arXiv preprint arXiv:1907.00420*, 2019.

[5] E. Cortez, M. Rojas Herrera, A. S. da Silva, E. S. de Moura, and M. Neubert, "Lightweight methods for large-scale product categorization," *Journal of the American Society for Information Science and Technology*, vol. 62, no. 9, pp. 1839–1848, 2011.

[6] V. Chordia and V. K. BG, "Large scale multimodal classification using an ensemble of transformer models and co-attention," *arXiv preprint arXiv:2011.11735*, 2020.

[7] D. Basaj, B. Rychalska, J. Dabrowski, and K. Gołuchowski, "Synerise at sigir rakuten data challenge 2020: Efficient manifold density estimator for multimodal classification," Sigir E-commerce Workshop. Retrieved from https://sigir-ecom. github. io ..., 2020.

[8] T. M. Tashu, S. Fattouh, P. Kiss, and T. Horváth, "Multimodal e-commerce product classification using hierarchical fusion," in *2022 IEEE 2nd Conference on Information Technology and Data Science (CITDS)*, pp. 279–284, IEEE, 2022.

[9] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[10] B. Khasoggi, E. Ermatita, and S. Sahmin, "Efficient mobilenet architecture as image recognition on mobile and embedded devices," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 16, no. 10, p. 2019, 2019.

[11] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.

[12] D. Gao, W. Yang, H. Zhou, Y. Wei, Y. Hu, and H. Wang, "Deep hierarchical classification for category prediction in e-commerce system," *arXiv preprint arXiv:2005.06692*, 2020.