Selling Flow Enhancement by Early Product Categorization

Gregory Goren[1], Ido Guy[2] and Slava Novgorodov[3]

[1]eBay Research, Netanya, Israel
[2]Ben-Gurion University of the Negev
[3]Tel Aviv University

Abstract

Product categorization in e-commerce platforms is an action of placement and organization of products into their respective classes. It has attracted a lot of research interest and attention as one of the most fundamental tasks in e-commerce. Categorization has high importance for both buyers and sellers, and plays a significant role for various downstream tasks such as product search, price comparison, and complementary product recommendation. In this work, we study product categorization based solely on the product's title and, specifically, prefixes of the title. This aims at identifying the category at an early stage of the selling flow, the process in which a seller uploads an item offered for sale to the e-commerce platform. Once the item's category is identified, the rest of the selling process can be adapted accordingly and expedited towards a smooth conclusion. We perform an extensive analysis of title prefix categorization, inspecting to what degree the product categorization task could be effectively accomplished while only using the beginning of the title. To this end, we propose BERT-Attrs, an extension of BERT that considers, in addition to the prefix's representation, also the association of its tokens with attributes, such as brand, color, or material. Evaluation, conducted over datasets from two of the world's largest e-commerce platforms, with hundreds of categories, considers the prefix-based categorization task both from a classification and recommendation points of view. To the best of our knowledge, we are the first to introduce and study the task of product categorization based on title prefixes.

1. Introduction

E-commerce platforms have demonstrated tremendous growth over the past decade, with the number of products available for online shopping rapidly increasing year over year. Maintaining these products in an organized manner requires much effort both from the sellers who upload their inventory for sale and from the e-commerce platforms hosting these products. Sellers are expected to provide as accurate product data as possible, which in turn has a major effect on buyers' purchase likelihood and the marketplace's success as a whole [1]. E-commerce platforms need to match products offered for sales across multiple sellers, present them to buyers in an engaging manner, and, ultimately, facilitate the transaction between sellers and buyers.

When sellers upload a new product for sale on an e-commerce platform, they need to provide much information, so that it can be correctly and attractively presented to potential buyers. Often, the upload process involves coming up with a product title, selecting the most relevant category in the platform's product taxonomy, deciding which product attributes to provide, uploading images, writing a product description, and providing information pertaining to price, shipping, and available quantity in stock. We refer to this process, supported by a dedicated user interface in leading e-commerce platforms, as the selling flow (also sometimes referred to as the listing flow) [2, 3]. This process usually consist of many steps and is known to often be cumbersome, requiring a substantial time investment from sellers and sometimes forming an entry barrier for the selling process as a whole [4, 5]. In particular, sellers often struggle to adapt

ECOM'25: SIGIR Workshop on eCommerce, Jul 17, 2025, Padua, Italy
ggoren@ebay.com (G. Goren); idoguy@acm.org (I. Guy); slavanov@post.tau.ac.il (S. Novgorodov)
https://slavanov.com (S. Novgorodov)
0009-0003-1822-5275 (G. Goren); 0000-0002-5525-1064 (I. Guy); 0000-0003-4082-7128 (S. Novgorodov)


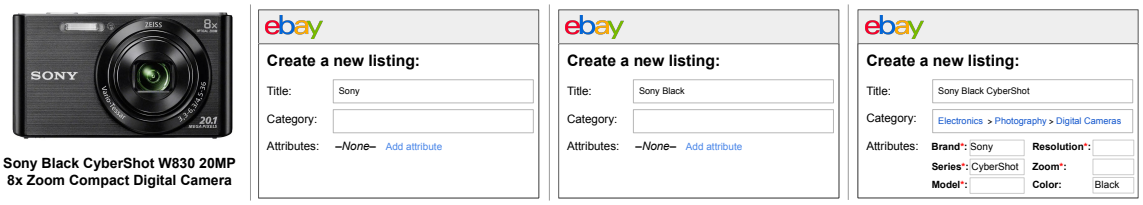

Figure 1: Example of a selling flow (eBay). In the last step, a category is identified and relevant attributes can be added or edited.

to the specific taxonomy and terminology of the e-commerce platform, come up with the best title for their product, and verify that the most important attributes are provided, to ensure that the product is easy to find, highly ranked by recommendation algorithms, attracts the buyer's attention, and is perceived as a quality purchase [1, 6]. Research has therefore been devoted to optimizing various phases of the selling flow, in forms such as price guidance (e.g., [7, 8]) and title optimization (e.g., [9, 10]).

A first step that can play a key role in making the selling flow smoother is the identification of the correct category of the product offered for sale. Once the offered product can be associated with a category or type (e.g., a wristwatch, a juicer, or a backpack), the selling flow can be adapted to facilitate the remainder of the process for the seller and enable a swift and productive conclusion. Specifically, once the category is known, the seller can be directed to provide the most relevant information for that category. For instance, the platform can explicitly ask the seller to provide the name of the network carrier and the capacity of the internal storage if it categorizes the product as a smartphone. In other cases, the platform can redirect the seller to a separate flow if it detects the specific product category. For example, a platform for selling consumer electronics may not support personal computers and therefore redirect sellers to an affiliated website when it detects that they try to sell a laptop. Furthermore, domain-specific tools can be applied on top of seller-provided information once the product's category is known. For instance, computer vision models for identifying product attributes that specialize in specific business verticals (e.g., Jewelry, Toys) can be applied to seller-provided images [11].

Automatic product categorization is one of the most fundamental tasks in e-commerce and received attention in different studies over the last two decades. Many works addressed this task under varying assumptions and product facets as the sources of data. Several works use product descriptions and reviews (e.g., [12, 13, 14]), while others use a combination of titles and images (e.g., [15]). Since title is the most essential and typically the first facet of the product provided by sellers, multiple works have focused on the task of product categorization based on the title only [16, 17, 18]. Yet, coming up with a good title can be a demanding task [9]. In this work, we therefore suggest that categorization can be effectively performed before a complete title is typed in. Such early categorization, based on the first few typed tokens of the first draft of the title, can help enhance the selling flow. Devising a method that can address this task based on as few tokens as possible is the focus of our study.

It should be noted that manual product categorization (i.e., when sellers explicitly assign a category to the product at the beginning of the selling flow) is quite difficult and sometimes even unfeasible. First, the seller should be familiar with the platform's set of categories (which may consists of hundreds and sometimes thousands of various categories). Moreover, the category structure and hierarchy may differ from one platform to another, making the process even more cumbersome and confusing. It is therefore not surprising that all large e-commerce platforms apply an automatic approach for product categorization [17, 19].

Figure 1 shows an example for the potential operation of early categorization as part of the selling flow on eBay, one of the world's largest e-commerce platforms. This example considers a seller who intends to upload a camera and comes up with the title Sony Black CyberShot W830 20MP 8x Zoom Compact Digital Camera. This title contains 10 tokens, however the product category (digital camera) is mentioned only at its end. As the seller starts to type the

title, it is difficult to identify an accurate category based on the first two tokens, since Sony is a manufacturer of many types of electronic goods (e.g., TVs, Cameras, Smartphones) and Black is a common color across a wide variety of categories within the Electronics business vertical. However, the third token (CyberShot) is a well-known series of digital cameras, therefore already disclosing the product's type without an explicit mention. This early category recognition allows for applying an attribute extraction model [20, 21, 22] specialized in the cameras domain, which can identify the existing attributes from the title's prefix (brand, color, and series, in our example) and suggest the seller to fill in the remaining key attributes for this type of product (model, zoom, and resolution), after they have typed only 3 tokens rather than 10. Once the seller has provided these, they may never need to type the whole title. The platform can automatically suggest a title based on the already-provided attributes (potentially reusing titles of existing products uploaded by other sellers) and the seller's efforts may focus on uploading some images or providing price and stock information.

To our knowledge, all prior efforts on product categorization considered the product title as a whole [16, 17, 18], based on a user-provided indication that the title has been completed (e.g., clicking on the next requested field). Since product titles in e-commerce are typically long – our analysis indicates that the median title length on eBay is 12 tokens – early categorization can save substantial efforts. Specifically, we consider early categorization based on at most of half of the title's tokens, aiming to save at least half of the title typing effort. No less important, however, is the ability to suggest key attributes to the seller before having to complete a full title (see last step in Figure 1). This spares the need to apply attribute extraction techniques over the rest of the title [20, 23], saves the seller from having to familiarize with the taxonomy and terminology of the e-commerce platform, and allows to automatically suggest a full title that the seller can use.

The task of early categorization based on only very few tokens is challenging, since the input is short and partial. Our solution is based on state-of-the-art natural language processing methods, which involve contextual embeddings of the prefix tokens produced by language models. To allow more efficient category inference, we extend BERT [24] in a novel manner, by encoding attribute-value information identified in the prefix. While we examine the traditional classification approach, aiming to identify the one specific category matching the prefix, we also consider a recommendation approach, which allows involving the seller in the process, asking them to select one out of few (3 or 5) options as part of the selling flow, after typing the first few title tokens. Our experiments over large datasets from two of the world's largest e-commerce platforms, eBay and Amazon, show the superiority of our novel approach relative to baselines.

Our work's main contributions can be summarized as follows:

- To the best of our knowledge, we are the first to introduce and study the task of product categorization based on title prefixes, motivated by the need to enhance the selling flow experience on e-commerce platforms.
- We provide an extensive analysis of the effect of title prefix length on product categorization quality, also tying to the length of the original title.
- We propose a novel method for product categorization that extends BERT to include product attributes, which we empirically show to consistently improve performance.
- We address the product categorization task as a recommendation problem, which can be an integral part of the selling flow, and show that this approach yields high performance gains, making it applicable for short prefixes, with as few as 3 to 4 tokens.

2. Related Work

E-commerce platforms provide a space for buyers and sellers to connect and engage in online transactions. While much research has been conducted to improve the experience of buyers,

for instance in product search [25], personalized recommendations [26, 27], and product review summarization [28], less attention has been dedicated to improving the end-to-end selling experience. Nevertheless, in addition to studies on product categorization (e.g., [12, 29, 17], discussed in detail below), several works have focused on different phases of the selling process. Specifically, recent work explored title optimization to help sellers come up with the most effective and attractive title [9, 10]; attribute enhancement to ensure that the product information most important to consumers is provided [30, 31, 22]; and description generation to equip sellers with automatically-produced descriptions of their products [32, 33]. Additionally, research has been dedicated to the task of price guidance (e.g., [7, 8]), which aims to provide sellers with price suggestions, typically based on the rate of similar products recently purchased.

Our work examines the task of product categorization, motivated by the desire to enhance the selling flow as early as possible, by adapting it to the type of the product offered for sale. To this end, we take a novel approach by using only the first few tokens of a title to predict the product's category, rather than relying on the whole title or additional information such as description, attributes or images, as has been done in previous work (e.g., [14, 34, 15]). To the best of our knowledge, we are the first to examine such early categorization based on title prefixes.

In general, the task of product categorization in e-commerce has received a lot of attention in the last two decades. Several works used product descriptions to identify a product's category [29, 12, 14, 13, 35]. For instance, Liu and Wangperawong [36] fine-tuned a BERT model for product categorization via descriptions. Their work highlights the effectiveness of BERT with respect to XLNet [35] over that task. Other works performed categorization based on additional facets of the product, such as reviews [34] or images [37, 38]. Using images for categorization in e-commerce can be challenging, as they are often of poor quality or missing altogether [39, 40].

A few works used solely the title for product categorization, since it is typically available for all products. Shen et al. [16] proposed a hierarchical approach that decomposed the categorization problem into a coarse-level task and a fine-level task for categorizing the title. Paulucio et al. [41] used a fine-tuned BERT model's title embedding as an input to additional machine learning models for product categorization. Their work showed that BERT was able to produce effective representations of product titles for the categorization task. Other works used the title for hierarchical categorization and highlighted the differences between title and text classification [17, 18]. For instance, stemming and stopword removal can both be beneficial for text classification, but are not suggested for title classification. As mentioned above, our work differs from prior efforts by its focus on incomplete information, i.e., studying the categorization task over title prefixes.

A related body of research to our work focuses on short text classification. The core challenge of this task is to address the lack of available information in short texts. A number of works proposed methods that employ external information, such as latent topical representation and taxonomy-based semantic features, to enhance the representation of the short texts [42, 43]. Moreover, query classification can be viewed as a short text classification problem. The average length of a typed query in web search ranges between 1.9 and 3.2 tokens [44, 45]. Several studies tackled query classification by exploiting different types of external information to overcome the ambiguity manifested by the shortness of queries [46, 47, 48]. Other works used deep neural models for the task of query classification for event categorization and extreme classification [49, 50]. Our work has two primary differences from short text classification. First, short text classification typically addresses short textual snippets that are complete, while in our case the prefix only covers the beginning of the title and may miss key information that appears later. Second, to our knowledge, none of the existing works on short text classification specifically focuses on product titles, which pose unique characteristics and loose language structure, with a high number of nouns (attribute values) and low grammatically [21].

3. Prefix-Based Categorization

In this section, we describe the product categorization task we examine, as well as the models and methods we use to address it.

3.1. Categorization approaches

The task of product categorization aims to identify the category of the product given its characteristics. In our work, the source of information taken into account for conducting the task is only the product's title, and specifically a prefix of the title. We examine categorization in its classic approach, as a classification problem. To further improve performance in a way that would be applicable early in the selling flow, we also suggest and evaluate a recommendation approach, which entails the suggestion of the top-k most probable categories. Both approaches require a learned model and the recommendation approach also requires the cooperation of the seller in selecting the correct category out of a short list of suggested categories during the selling flow. The benefit of the recommendation approach is that it allows more room for error, since identifying the correct category among the top-k is sufficient. On the other hand, classification can facilitate an automatic flow without any seller involvement (e.g., via an API that allows batch listing of multiple items). It is therefore interesting to examine the trade-off between these approaches.

3.2. Models

Our experimental setting focuses on analyzing the performance of models based on the pre-trained BERT architecture [24], which has been previously shown to represent product titles effectively for categorization and outperformed other pre-trained language models [36, 41]. The pre-training corpora are the BookCorpus dataset [51] and English Wikipedia. We fine-tune our pre-trained models with an additional classification layer $V \in \mathbb{R}^{C \times d}$ by optimizing the cross-entropy loss:

$$\mathbf{L}_{CE} = -\log \frac{\exp((H^{cls}V^T)_k)}{\sum_{j=1}^{C} \exp((H^{cls}V^T)_j)}$$

where C is the number of categories, d is the dimension of BERT's hidden state representation, k is the index of the correct category, and $H^{cls} \in \mathbb{R}^d$ is the final hidden state representation of the special [CLS] token. We estimate the category association probabilities via $\frac{\exp((H^{cls}V^T)_k)}{\sum_{j=1}^{C} \exp((H^{cls}V^T)_j)}$ and we use the ordering induced from these probabilities for both the classification and recommendation approaches. We examined multiple training techniques over title prefixes, along with different input enhancement methods, and compared them to learning based on complete titles.

In addition to different methods utilizing BERT, we also inspect the results of LSTM, a recurrent neural network based on long short-term memory architecture [52], with pre-trained word2vec embeddings using continuous bag of words (CBOW) with negative sampling [53].

3.3. Title prefix training methods

Throughout this work, we considered prefixes of length in the range of $\in [1, \lfloor \frac{l_t}{2} \rfloor]$, where l_t is the title length in tokens. For instance, for titles of length 12, we examined prefixes of length 1 to 6 tokens. We inspected two different learning methods for prefix categorization. The first trains only on complete product titles and the second trains on prefixes. Specifically for the latter, for each title in the training set, in each iteration of the training process (epoch), a prefix length is drawn uniformly at random out of the $[1, \lfloor \frac{l_t}{2} \rfloor]$ range and the corresponding prefix is used for

training.[1]

As we will later show, prefix-based training substantially outperformed training based on complete titles. We therefore proceeded to build our models based on title prefix training. Since title prefixes consists of very few tokens, we sought to extend their representation with additional information to allow more effective learning, as described in the next section.

3.4. Attribute-enhanced BERT

In addition to the standard BERT-based approach (training using randomized prefixes), we devised an extension of BERT, termed BERT-Attrs, which makes use of novel input enhancement methods. As will be shown in Section 5, BERT consistently outperformed LSTM when trained on title prefixes, and we therefore opted to focus on a BERT-based extension. Concretely, BERT-Attrs leverages additional information regarding the attributes of a product, extracted using a state-of-the-art named entity recognition (NER) method specialized and trained for attribute extraction from product titles [54]. This NER method aims at associating each token in the title prefix with an attribute name such as `brand`, `color`, or `size`[2]. Overall, 63.68% of the tokens in the training set could be associated with an attribute name. The extracted attribute information is then injected into the BERT input as an additional sentence in the form: "[$AttributeName_1$] $value_1$ [$AttributeName_2$] $value_2$ [$AttributeName_N$] $value_N$", where each [*AttributeName*] is defined as a special token. For example, for the prefix:

"green adidas cotton"

the derived "attribute tokens" sentence would be:

"[Color] green [Brand] adidas [Material] cotton".

In cases where prefix tokens could not be associated with an attribute name, BERT-Attrs injects them along with the special corresponding token [UNKNOWN] as their attribute name. Considering the example above, suppose the last token in "green adidas cotton" could not be classified into an attribute name, BERT-Attrs would inject the sentence:

"[Color] green [Brand] adidas [UNKNOWN] cotton"

Based on this injection, the whole input to BERT is of the form:

[*CLS*] <prefix tokens> [*SEP*] <attribute tokens> [*SEP*]

where [CLS] and [SEP] are standard tokens, used as in the original BERT [24], to denote the start of the input and a separation between two input sentences, respectively. To examine whether the first part of this input, i.e., the prefix in its original form, is necessary to include in addition to the attribute tokens, we also experimented with a variant that only considers the attribute tokens as the prefix's representation input to BERT. We refer to this as BERT-AttrsOnly.

In addition to the methods mentioned above, we examined several other representations that consider attribute information as part of the input to BERT. First, we considered a variant of BERT-Attrs that excludes tokens that cannot be associated with an attribute from the attribute injection altogether. In other words, only tokens that can be associated with an attribute name are included in the attribute-based representation that follows the original prefix. Second, we experimented with a variation that aims to exploit BERT's pre-trained language understanding by adding the attribute information as natural language to the prefix. For instance, the attribute

[1]We also experimented with a variation of this approach that trains over all prefixes in the range $[1, \lfloor \frac{l_t}{2} \rfloor]$ for each title of length l_t in each training iteration. This variation, however, performed similarly and often slightly more poorly compared to drawing a prefix at random, while bearing substantially higher computational costs.

[2]We used an in-house NER model trained on eBay's product titles. The quality of the model is depicted at [54].

Table 1
Title examples from the eBay and Amazon datasets.

Vertical	Category	Dataset	Title
Fashion	Shirts	eBay	Calvin Klein Womens Performance Plus Size Twist Front Cap Sleeve Top Stone
	Shoes	eBay	Calvin Klein Womens Flats in Blue Color, Size 10 HHA
	Sweaters	eBay	CALVIN KLEIN Womens Black Long Sleeve V Neck Wrap Party Sweater Size: XL
	Women's Sunglasses	Amazon	The North Face Profit Pack Blue Sunglasses NF034S 0084
	Women's Sneakers	Amazon	The North Face Base Camp Casual Sneaker - Women's - B Width Size 7.5-B Color DeepBlue
	Women's Flats	Amazon	The North Face Base Camp Mary-Jane (8, TNF Black)
Electronics	Video Games	eBay	PS5 Marvel's Spiderman Remastered Edition for Sony Playstation 5 Digital Code
	Video Games Console	eBay	Sony PlayStation 5 Console Disc Brand New SHIPS IMMEDIATELY! FAST!
	Video Games Console Accessories	eBay	Sony Playstation 5 Dualsense Wirless Controller White Factory Sealed
Home & Garden	Garden Bridge	eBay	Miniature Japanese Wood Garden Bridge, 25"L x 11-1/2"H x11-1/8"W, New
	Garden Building	eBay	10'X20' Garage Tent Carport Car Shelter Sidewall Canopy Caravan Cover Enclosure
	Garden/Beach Umbrella	eBay	9' X 11' Double Top Deluxe Rectangle Patio Umbrella Offset Hanging Umbrella Out

injection of our example prefix would be in the form: "The prefix green adidas cotton contains green as [Color], adidas as [Brand] and cotton as [Material]". Finally, we considered an approach that operates directly on BERT's architecture. This is done by adding the word embedding of the corresponding attribute to each token's representation. Each token's embedding is therefore derived by:

$$WordEmb \oplus TokenTypeIDEmb \oplus PositionEmb \oplus AttributeEmb$$

where $\oplus$ marks the tensor addition operation; WordEmb, TokenTypeIDEmb, and PositionEmb are the standard components in the BERT architecture [24]; and AttributeEmb is the added attribute representation. For tokens without any attribute association, we experimented both with a variant that adds the representation of the special token [UNKNOWN] and a variant that does not add any attribute representation.

All methods described in the previous paragraph consistently underperformed both BERT-Attrs and BERT-AttrsOnly. We therefore exclude them from our reported results, for clarity of presentation, and focus on comparing LSTM, BERT, BERT-Attrs, and BERT-AttrsOnly, trained over title prefixes as described in Section 3.3.

Table 2
Dataset characteristics.

	eBay	Amazon
# titles	17,195,225	464,745
# classes	704	147
Max # titles per class	1,896,923	39,055
Min # titles per class	3	3
Avg. # titles per class	24,425.04	3,161.53
Median # titles per class	3154	990

4. Experimental Setup

In this section, we provide details about our setup for experimentation, including the two datasets and their basic characteristics and evaluation metrics. We publicly release our implementation for reproducibility purposes.[3]

4.1. Datasets

We experiment with two different datasets. Our main dataset is from eBay, one of the world's largest e-commerce platforms.[4] For reproducibility purposes, we also include results over a

[3] https://github.com/titleprefixes/prefixes_code
[4] A small sample of eBay's data can be found at https://github.com/titleprefixes/prefixes_code/blob/main/data/ebay_example_data.tsv

Table 3
Distribution of the number of titles per category.

# titles per category	Portion of all categories		Portion of all titles	
	eBay	Amazon	eBay	Amazon
<=10	2.13%	11.56%%	0.00%	0.02%
11-100	8.66%	17.69%	0.02%	0.21%
101-1,000	21.59%	21.09%	0.40%	3.20%
1001-10,000	39.49%	43.54%	6.36%	53.87%
10,001 - 100,000	23.58%	6.12%	30.65%	42.71%
100,001 - 1M	4.26%	–	44.51%	–
>1M	0.28%	–	18.06%	–

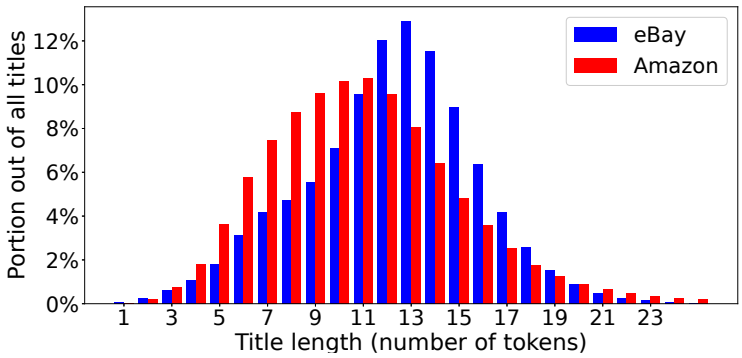

Figure 2: Distribution of title length across our two datasets.

public dataset – the Amazon products dataset. In both datasets, the categories are defined by a taxonomy , where verticals are at the highest level and the categories we consider for the task are at the lowest level.

4.1.1. eBay dataset

This dataset includes a sample of 17 million listing titles and their corresponding categories from eBay's logs. The sample contains listed items (listings) that were offered for sale on the United States site during December 2020. In this dataset, the listings stem from 6 different verticals and 704 corresponding categories. We randomly split the data into training, validation, and test sets with a ratio of 60%, 20%, and 20%, respectively. Table 1 presents several examples of titles and their corresponding verticals and categories in the dataset. In some cases, the differences between categories are subtle. For example, Watches Parts and Watches Accessories or Golf Equipment and Golf Parts Repair. These minor differences and overall large number of categories entail difficulty in the categorization task.

4.1.2. Amazon dataset

We consider the Clothing, Shoes and Jewelry vertical (also referred to as "Fashion") of the publicly available product dataset from Amazon [55], another one of the world's largest e-commerce platforms. Overall, this portion of the dataset contains 464,745 product titles spanning 147 categories. As in the eBay dataset, some categories are rather similar to one another, for instance Men's Shirts and Women's Shirts or Men's Wrist Watches and Men's Pocket Watches. We use the same splitting scheme of 60%-20%-20% for training, validation, and test sets, respectively, as for the eBay's dataset. The products in this dataset span the years 1996-2014. Table 1 presents several examples of titles and their corresponding categories in the dataset. Basic characteristics of both the eBay and Amazon datasets are summarized in Table 2.

4.1.3. Dataset characteristics

Table 3 demonstrates the distribution of titles across the category size (the category size is the number of titles in the dataset associated with that category). These statistics convey the extreme imbalance with respect to class size in both datasets. In the eBay dataset, 10.79% of the categories (up to 100 titles per category) cover only 0.02% of all titles, while in the Amazon dataset, 29.25% of the categories (also up to 100 titles per category) cover 0.2% of all titles. On the other hand, on eBay, 4.54% of the categories (more than $100K$ titles per category) account for 62.6% of all titles, while in Amazon 6.12% of the categories (bin "10K-100K") cover 42.71% of the titles. Naturally, this imbalance stems from the skewed distribution of e-commerce items across categories, reflecting high differences in their popularity.

Figure 2 presents the distribution of title lengths (number of tokens) in both datasets. In the eBay dataset, the average length is 12.24, with a standard deviation of 3.59 and a median of 12. In the Amazon dataset, the average length is 11.06 tokens, with a standard deviation of 4.24, and a median of 11. The titles on the eBay's platform are strictly restricted to 80 characters, whereas on Amazon there is a restriction of 200 characters. Thus, in rare cases, Amazon's titles can become substantially longer when compared to eBay. This phenomenon at the tail of the distribution can be observed in Figure 2.

4.2. Evaluation

4.2.1. Classification

For the category classification task, we report accuracy, macro precision, and macro recall [56]. We focus on these metrics to understand the model's performance in the scenario of extreme imbalance. The macro metrics consider the average precision and recall across all classes, while assigning each class with an equal weight, regardless of the number of its associated instances. We omit micro precision and recall as they are both equal to accuracy when each data point is assigned to exactly one class. Formally, the metrics are defined as:

$$Accuracy = \frac{\sum_{i=1}^{n} \mathbb{1}_{\hat{y}_i = y_i}}{n} \qquad MacroX = \frac{1}{C} \sum_{j=1}^{C} X_j$$

where n is number of instances, $\hat{y}_i$ is the predicted label of instance i, y_i is its true label, C is the number of classes and $X \in \{Recall, Precision\}$.

4.2.2. Recommendation

For the category recommendation task, we focus on the Hits@k metric [57, 58], defined as the fraction of examples where the correct class is among the top-k ranks. It is worth noting that Hits@1 is equivalent to the accuracy metric. Formally, it is defined as:

$$\frac{\sum_{i=1}^{n} \mathbb{1}_{y_i \in TopK(\{\hat{y}_j\}_{j=1}^{C})}}{n}$$

where $TopK$ is the set of K highest-ranked classes according to model's predictions.

5. Results

In this section, we report results for classification and recommendation, using the methods described in Section 3. We start by showing that the results produced by training using complete titles are not satisfactory, and thus motivate our approach of training over prefixes. We then compare different approaches for the prefix-based product categorization task using prefixes for training, for the problem of classification, followed by analogous results for recommendation.

Table 4
Accuracy for prefix categorization of 12-token titles using models trained on complete titles.

Prefix length	eBay		Amazon	
	LSTM	BERT	LSTM	BERT
1	25.62	20.20	26.20	12.62
2	38.46	33.78	31.74	22.42
3	49.18	45.48	37.70	30.41
4	58.90	56.18	43.80	35.81
5	66.90	65.32	50.88	43.27
6	73.31	72.66	59.14	53.54

Finally, we examine the contribution of different attributes to the categorization performance, using ablation tests. The majority of the results are reported over a test set of titles with an original length of 12 tokens, which is the most common across our two datasets combined, as described in Section 4.1. We report results for all prefix lengths in the range of $[1, 6]$. Evaluation using other title lengths yielded very similar results, hence excluded here.

5.1. Title-trained classification

We first examine the use of complete titles as training examples for prefix-based categorization. To this end, we apply the BERT and LSTM models trained over complete titles on our varying-length prefixes. The classification results for titles of length 12 are summarized in Table 4. Overall, we observe low values with respect to all reported metrics and datasets. Even for prefixes of length 6 (half of the title), the accuracy did not exceed 74% and 60% over the eBay and Amazon datasets, respectively. However, when applied on complete titles, the accuracy of both models exceeds 90% on both datasets. This indicates that the learned patterns over the complete titles do not generalize well when used to categorize prefixes. The setup of our proposed methods is intended to address this issue and our observations provide empirical support.

Table 5
Accuracy, macro precision, and macro recall for prefix categorization of 12-token titles over the eBay dataset.

Prefix length	LSTM			BERT			BERT-Attrs			BERT-AttrsOnly		
	Acc	Macro-p	Macro-r	Acc	Macro-p	Macro-r	Acc	Macro-p	Macro-r	Acc	Macro-p	Macro-r
1	35.06	**35.42**	14.92	35.09	32.26	15.52	**37.08**	34.18	17.02	37.00	34.74	**17.18**
2	50.31	47.56	29.28	52.84	47.52	33.48	**54.34**	48.70	**35.43**	54.01	**48.87**	35.34
3	60.60	54.99	39.19	63.85	57.06	44.53	**65.16**	**58.33**	**47.04**	64.88	58.18	46.68
4	68.59	59.22	46.44	71.84	62.72	52.13	**72.87**	**65.66**	**54.81**	72.63	64.06	54.24
5	74.52	62.47	51.58	77.56	66.62	57.48	**78.38**	**68.29**	**59.87**	78.21	67.34	59.49
6	78.85	64.72	54.89	81.51	68.25	60.47	**82.19**	**70.33**	**62.75**	82.06	68.89	62.25

Table 6
Accuracy for prefix categorization of 12-token titles over the Amazon dataset.

Prefix length	LSTM	BERT	BERT-Attrs	BERT-AttrsOnly
1	42.71	44.32	44.96	**45.39**
2	52.63	54.91	**55.59**	55.25
3	60.14	60.85	**61.79**	61.60
4	65.47	66.52	**67.76**	67.08
5	70.81	72.52	**73.10**	72.33
6	75.44	76.76	**77.35**	77.04

5.2. Prefix-trained classification

In this section, we evaluate our ability to classify title prefixes using our proposed methods, discussed in Section 3, and explore the trade-offs between prefix length and categorization performance. Tables 5 and 6 present the classification performance results when training on prefixes using the LSTM, BERT, BERT-Attrs and BERT-AttrsOnly methods over the eBay and Amazon datasets, respectively.

As expected, classification performance degrades as prefix length decreases. For prefixes up to 3 tokens, we observe poor classification performance across both the eBay and Amazon datasets. This reflects the difficulty in identifying the correct category based on very few tokens. Consider, for example, the titles presented in Table 1 in the Fashion vertical. In both datasets, we observe titles that share the same prefix when considering the first 3 tokens (i.e., Calvin Klein Womens or The North Face), but are from different categories. Therefore, even a perfect classifier would not distinguish between them based on three-token prefixes. On the other hand, for prefixes of length 6, i.e., 50% of the title's tokens, we can observe much improved performance. In particular, over the eBay dataset, all BERT-based methods (BERT, BERT-Attrs and BERT-AttrsOnly) reach accuracy of over 80%.

Examining Table 5, it can be observed that the BERT-Attrs method outperforms all other methods in terms of accuracy across all different prefix lengths. It also outperforms the other methods in the vast majority of the cases in terms of macro precision and macro recall. The consistent gap in performance between BERT-Attrs and BERT indicates that modeling the attribute information as part of the input prefixes helps the categorization process. The uplift of BERT-Attrs on top of BERT is more substantial for short prefixes (e.g. +5.7% for prefix length 1 compared to 0.8% for prefix length 6), which are the most difficult to categorize and therefore can benefit the most from the additional information used by the BERT-Attrs model. The BERT-AttrsOnly model yields consistently lower performance than the BERT-Attrs model (in accuracy), indicating that the inclusion of the title in its original form as part of the input is not redundant. The results over the Amazon dataset in Table 6 show similar trends, with BERT-Attrs outperforming the other methods across all prefix lengths except for length 1 where BERT-AttrsOnly performs best. The gap between BERT-Attrs and BERT is consistent, while largest for prefix lengths of 3 and 4, and substantially diminishing for the longer prefixes of 5 and 6 tokens.

Figure 3 presents the classification accuracy as a function of the original titlle length (for original length $\in[9, 18]$), for prefixes of length 4, 5, and 6, using the BERT-Attrs method. The results indicate that the performance is rather invariant to the original length of the title. That is, the accuracy remains stable across all the original title lengths, while the prefix length is the main factor that affects the performance. We note that the results when using the BERT and BERT-AttrsOnly methods show a similar trend.

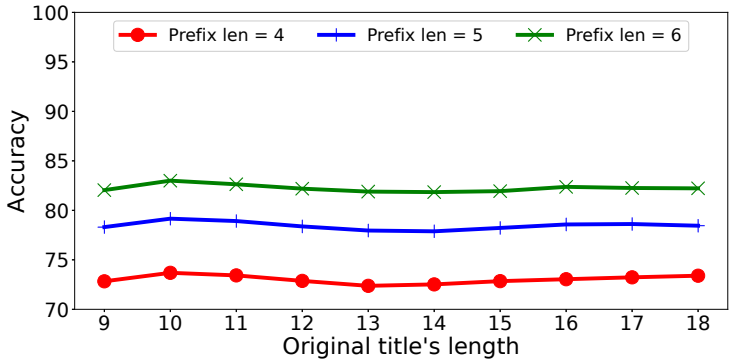

Figure 3: Prefix classification accuracy by length of the original title over the eBay dataset using the BERT-Attrs model.

Table 7
Hits@k for title prefix recommendation, over titles of original length of 12 tokens.

	eBay								Amazon							
	LSTM		BERT		BERT-Attrs		BERT-AttrsOnly		LSTM		BERT		BERT-Attrs		BERT-AttrsOnly	
Prefix length	Hits@3	Hits@5	Hits@3	Hits@5	Hits@3	Hits@5	Hits@3	Hits@5	Hits@3	Hits@5	Hits@3	Hits@5	Hits@3	Hits@5	Hits@3	Hits@5
1	53.86	63.13	54.85	64.39	**56.93**	**66.34**	56.81	66.25	67.51	76.41	68.30	77.84	**69.87**	**79.21**	69.57	79.16
2	69.19	77.19	72.40	80.18	**73.57**	**81.16**	73.43	81.04	76.38	84.10	78.09	85.43	78.43	**85.80**	**78.44**	85.63
3	78.39	84.95	81.45	87.54	**82.42**	**88.22**	82.25	88.16	82.32	88.74	83.25	89.48	**83.97**	89.89	83.55	**90.06**
4	84.54	89.72	87.20	91.83	**87.87**	**92.24**	87.76	92.23	85.32	91.19	86.60	91.80	**86.92**	92.04	86.82	**92.19**
5	88.62	92.64	90.90	94.43	**91.39**	**94.68**	91.31	94.67	88.88	93.62	89.75	93.87	**90.03**	**94.00**	89.85	93.95
6	91.31	94.47	93.26	95.95	**93.59**	**96.14**	93.58	96.11	91.72	95.21	92.03	95.28	**92.30**	**95.53**	92.17	95.31

5.3. Prefix ranking

In this section, we examine the performance of the learned models in the scenario of category recommendation for title prefixes. That is, we use our models to recommend the top-k most probable categories based on the input. This is achieved by considering the scores of our classifier as ranking scores over categories and using the induced ranking for the recommendation.

Table 7 presents the Hits@k results of all four methods over both the eBay and Amazon datasets for titles of length 12. We focus on $k \in \{3, 5\}$, as we assume these values allow sellers to traverse the candidate list and select the suitable category without substantial cognitive load. We also focus on low values of k to allow the seller to view the entire list of categories on a single screen.

Observing the table, we can see that as in the classification results, performance improves with the length of the prefix. Moreover, as in the classification case, the BERT-Attrs model outperforms all other models across all prefix lengths for the eBay dataset, and in the vast majority of the cases for the Amazon dataset (in a few cases BERT-AttrsOnly slightly outperforms BERT-Attrs). The gap between BERT-Attrs and BERT is consistent for both Hits@3 and Hits@5 across all prefix lengths over both datasets, and is particularly large for the shorter prefixes, where attribute information is most essential to allow narrowing down the list of potentially matching categories. Overall, these results reinforce the added value of modeling attribute information. The small but rather consistent gap between BERT-Attrs and BERT-AttrsOnly attests again to the benefit of including the original title as part of the input representation. Also consistent with previous findings is the lower performance of the LSTM model compared to the BERT-based models.

In contrast to the classification case, however, the performance using all four models reaches high values starting already from very short prefixes. For example, considering the best performing BERT-Attrs model, while its accuracy over the eBay dataset only exceeds 82% for a prefix length of 6 (Table 5), it exceeds 82% with Hits@3 when the prefix length is only 3 tokens (25% of the title) and exceeds 81% with Hits@5 when the prefix length is only 2 tokens. This indicates that while short prefixes tend to be ambiguous and may fit multiple categories, the cardinality of the overall set of potential matching categories is already low after 2-3 tokens, in many cases. This enables the recommendation approach, which loops in sellers and allows them to disambiguate at an early stage, to be highly productive. In Table 1, the provided examples in the Electronics vertical demonstrate the above point. For example, Sony Playstation 5 (or PS5) is a prefix that can be shared across different categories, however, the list of relevant categories is limited to the video games domain, with only few alternatives.

Table 7 indicates that by the time a seller has typed 5 of the title's tokens, the Hits@k values using BERT-Attrs over both datasets are above 90%. For prefix length of 6 tokens, Hits@3 on both datasets is higher than 92% and Hits@5 exceeds 95.5%, reaching as high as 96.14% over the eBay dataset.

To conclude this section, we explore in more depth the potential gains in performance applying the recommendation method with different values of k. We focus on short prefixes of length 2, 3, and 4, as these demonstrated rather low accuracy using the classification approach (Tables 5

and 6). Figure 4 plots the Hits@k as a function of $k \in [1, 10]$ for these prefix lengths over the eBay dataset, for titles of original length of 12 tokens. It can be observed that a substantial performance gain is achieved when moving from Hits@1 (accuracy) to Hits@2, for all three prefix lengths. For instance, when the prefix length is 2, performance rises from 54.34 to 66.87, when moving from Hits@1 to Hits@2. As the value of k increases, the performance gain naturally becomes smaller, but the overall performance continues to increase. For k=10, which entails a rather cognitive-heavy selection from the seller, a prefix length of 2 tokens is sufficient to yield a hit in 88.73% of the cases.

Table 8
Attribute ablation results over 6-token prefixes on the eBay dataset. "Diff" shows the categorization accuracy difference between the original prefix and its version that excludes the attribute's tokens. "Average length" shows the average number of tokens of the corresponding attribute within 6-token prefixes in the category.

Shirts			Video Games			Home Decor		
Attribute	Diff	Average length	Attribute	Diff	Average length	Attribute	Diff	Average length
type	15.41	1.38	game name	47.95	4.05	type	25.63	1.93
style	15.16	1.59	type	5.38	1.37	model	23.34	2.17
sleeve length	9.98	1.68	platform	5.15	1.76	width	15.06	2.15
brand	6.92	1.82	region code	3.45	1.00	brand	13.81	1.59
size	4.63	1.10	country	3.21	1.01	country	7.84	1.41
department	3.15	1.49	publisher	1.28	1.03	material	5.82	1.16
color	1.22	1.28	media format	1.03	1.01	color	1.25	1.27

5.4. Attribute contribution

Our BERT-Attrs model relies on attribute information to learn different patterns within the titles to perform categorization. We therefore set out to explore the influence of key attributes on the categorization task using ablation tests. In this experiment, we focus on three different verticals within the eBay dataset – Fashion, Electronics, and Home & Garden. For each vertical, we consider the most popular category within the eBay's test set (see Section 4.1), as measured by the number of associated listings. The categories explored are Shirts, Video games, and Home Decor from the Fashion, Electronics, and Home & Garden verticals, respectively. For each category, we consider all its associated titles in our test set of at least 12 tokens and their corresponding 6-token prefixes. The total number of such prefixes in our test set from each category is 236,223, 39,757 and 47,623, respectively. For each category, we use the NER model [54] to extract attribute values from the prefixes. We report results over the top 7 most frequent attributes in each category. For each such attribute, we consider all prefixes in the corresponding category that include it. We measure the categorization accuracy across these prefixes, and compare it to the accuracy over the same set of prefixes while removing all tokens that correspond to values extracted for that attribute. This allows us to measure the relative impact of removing different attribute values from the prefix on the categorization performance.

Table 8 reports the accuracy difference between prefixes that include and exclude each of the top 7 attributes in each of the categories. Alongside the accuracy difference, which attests to its importance to categorization performance, the average length (in tokens) of each such attribute across all category prefixes that include it, is presented. Observing the table, it can be seen that the type attribute is the most important for Shirts and Home Decor categories. In the Shirts category, the average length of the type values is only 1.38 tokens, ranking only 5th out of the seven attributes in terms of length, indicating that the contribution to performance does not necessarily coincide with length. In the Home Decor category, the average length of type is 1.93 tokens, which is the third highest. The importance of type to the categorization task is intuitive, as it is typically closely associated with the category. For example, possible values for the attribute type in Shirts and Home Decor categories are T-shirt and Wall Picture,

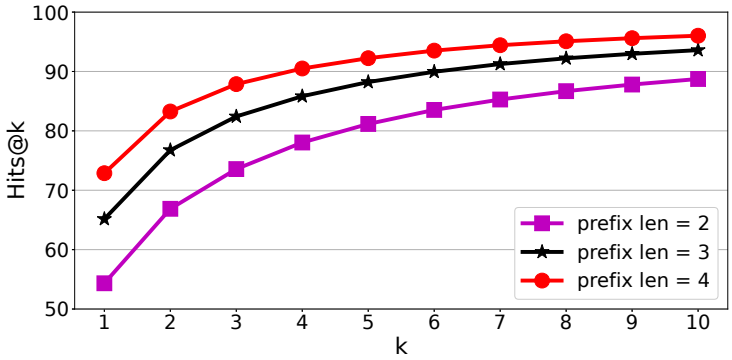

Figure 4: Hits@k as a function of k for prefixes of 12-token titles over the eBay dataset using the BERT-Attrs method.

respectively. These values are associated strongly with their respective categories. We note that `type` attribute tends to appear towards the end of the title. For instance, in the Fashion vertical of the eBay dataset, it appears on the second half of the title in 65.5% of its occurrences.

Other than `type`, the `style` attribute (e.g., retro, loungewear) is found to be the most important attribute for categorization in the Shirts category. In Home Decor, the `model` attribute is as nearly important for categorization as `type`, with values such as Cierra by Uttermost and ST1216B by FAIRFIELD.

For the Video Games category, the most important attribute is `game name`. This is an example of an attribute that is mostly unique to a specific category (e.g., Tony Hawk's Pro Skater) and therefore its occurrence is highly revealing. It also tends to be exceptionally long at over 4 tokens. Therefore, its removal leaves very few tokens on the prefix, leading to the very large performance gap in its ablation test. The `type` attribute, which is most important in the other two categories, tends to be more generic in Video Games: its most common values are Disc or Game. Accordingly, its importance as reflected in the ablation test is lower.

The commonly-used `brand` attribute is not found to be among the most important attributes for the categorization task, ranked fourth for both the Shirts and Home Decor categories. The relatively low importance of the `brand` attribute in Shirts is intuitive, as the same brand can span different categories. An example of such phenomenon is the known brand of Michael Kors, which has products in various categories, including dresses, shoes, shirts, handbags, and even watches. However, for the Home Decor category, the `brand` attribute can be more distinctive. For example, it can be associated with the designer's name (or an artist's name), which is less likely to be shared across different categories. The `brand` attribute is particularly common at the beginning of the title: in the eBay dataset, it appears on the first half of the title in 88.6% and 83.2% of its occurrences in Fashion and Home & Garden, respectively.

6. Conclusions and Implications

We studied the task of product categorization based on title prefixes. To our knowledge, we are the first to introduce and explore prefix-based categorization, which can play an important role in e-commerce platforms' selling flow, allowing to adapt early to the unique characteristics and attributes of a specific category, out of hundreds of available options. We first demonstrated that using both BERT and LSTM models, trained on complete titles, for the prefix categorization task, results in poor performance, and hence suggested a simple approach for training a model based on title prefixes, where for each title a prefix of random length is selected in each iteration of the training process. This method accomplished substantial performance improvements, with an accuracy of roughly 80% for prefixes of half the length of the title (e.g., 6 out of 12 tokens), and BERT consistently outperforming LSTM. Following, we suggested BERT-Attrs, a novel yet simple extension of BERT for learning over title prefixes, by extending the representation of

the prefix with attribute information. This extended representation showed to yield consistent performance enhancement, especially over prefixes of short length, where the original information provided in the prefix is more limited. Interestingly, our results indicated that classification performance using our prefix-trained approach is highly dependent on the prefix length, naturally increasing as the prefix is longer, but is almost completely oblivious to the length of the original title, for a given prefix length.

We demonstrated the difficulty of prefixed-based classification, which often introduces ambiguity, as the first few tokens are not uniquely identifying the specific category. Our attribute ablation tests showed that many attributes that commonly occur on titles, such as `color`, `size`, `material`, and even `brand`, which often appear at the beginning of the title, do not contribute much to accurate categorization. On the other hand, a revealing attribute such as `type` commonly appears toward the end of the title. This ambiguity, however, typically narrows down to only a handful of categories after as few as 2 or 3 tokens have been provided. We therefore suggest addressing the task using a recommendation approach, which loops sellers in the process and allows them to select the category out of a shortlist of candidates. We showed that this approach can dramatically improve the performance, reaching a hit rate of over 95% over both datasets for prefixes of 6 out of 12 tokens, when shortlisting to 5 candidates. Moreover, for prefixes of 3 tokens only, the hit rate within the top 5 already exceeds 88%. The category selection out of the recommended shortlist is, thus, a key step in our envisioned selling flow. It enables quick identification of the type of product offered for sale, and adaptation of the remainder of the process towards a simplified and smooth conclusion.

Our results quantitatively draw the trade-offs between both the prefix length and the size of the candidate shortlist, and categorization performance. These suggest a variety of alternatives for e-commerce platforms to implement prefix-based categorization, considering how early they want to provide a suggestion, whether they want to allow sellers to select the category out of a certain-sized list, and how accurate they desire the results to be. All of our results generalize similarly across two of the world's largest e-commerce platforms, eBay and Amazon, and can be reproduced over the public Amazon dataset.

As the task of prefix-based categorization has not been previously studied, it offers many opportunities for future research. Validating the results on additional e-commerce platforms, with different types of category sets and granularity, can help to further generalize. Even more importantly, in-vivo experimentation with our proposed solution is necessary to quantify its impact on the simplification and improvement of the selling flow. This experimentation would also provide an opportunity to leverage otherwise unattainable user feedback, such as whether and which suggested category was selected, at what stage, and how the selection was reflected in the remainder of the selling flow. Using prefix-based categorization in an online setting may also influence the way titles are formulated. Sellers may become increasingly aware of this feature and adjust their titles to contain category-distinctive tokens, such as `type` and `model`, in the beginning, to facilitate faster categorization. Finally, experimentation with additional models, including the potential use of LLMs within the flow, is an interesting future direction. Both in terms of the potential to increase performance and in terms of the challenge of applying large models in an online scenario.

References

[1] F. Moraes, J. Yang, R. Zhang, V. Murdock, The role of attributes in product quality comparisons, in: Proc. of CHIIR, 2020, pp. 253–262.

[2] How to create a listing: the step-by-step guide, 2022. URL: https://export.ebay.com/en/first-steps/how-create-listing/how-create-listing/.

[3] How to start selling on Amazon, 2022. URL: https://sell.amazon.com/sell.

[4] G. Fuchs, H. Roitman, M. Mandelbrod, Automatic form filling with form-bert, in: Proc. of SIGIR, 2021, pp. 1850–1854.

[5] H. Aragonda, K. Shaik, H. Jain, S. Shah, Accurate and real time assisted cataloging in e-commerce using dual images, in: Proc. of CODS-COMAD, 2022, pp. 265–269.

[6] M. Niemir, B. Mrugalska, Product data quality in e-commerce: Key success factors and challenges, Production Management and Process Control 36 (2022) 1–12.

[7] Y. Zheng, C. Gao, X. He, D. Jin, Y. Li, Incorporating price into recommendation with graph convolutional networks, TKDE (2021).

[8] Z. Pang, W. Xiao, X. Zhao, Preorder price guarantee in e-commerce, M&SOM 23 (2021) 123–138.

[9] J. Wang, J. Tian, L. Qiu, S. Li, J. Lang, L. Si, M. Lan, A multi-task learning approach for improving product title compression with user search log data, in: Proc. of AAAI, volume 32, 2018.

[10] L. Wang, J. Zhang, W. Yan, Toor: A novel product title optimization method based on online reviews in e-commerce, Frontiers of Business Research in China 9 (2015) 536.

[11] A. Dagan, I. Guy, S. Novgorodov, An image is worth a thousand terms? analysis of visual e-commerce search, in: Proc. of SIGIR, 2021, pp. 102–112.

[12] A. Cevahir, K. Murakami, Large-scale multi-class and hierarchical product categorization for an e-commerce giant, in: Proc. of COLING, 2016, pp. 525–535.

[13] A. Krishnan, A. Amarthaluri, Large scale product categorization using structured and unstructured attributes, arXiv:1903.04254 (2019).

[14] M. Y. Li, S. Kok, L. Tan, Don't classify, translate: Multi-level e-commerce product categorization via machine translation, arXiv:1812.05774 (2018).

[15] S. Eskesen, Improving product categorization by combining image and title, 2017.

[16] D. Shen, J.-D. Ruvini, B. Sarwar, Large-scale item categorization for e-commerce, in: Proc. of CIKM, 2012, pp. 595–604.

[17] I. Hasson, S. Novgorodov, G. Fuchs, Y. Acriche, Category recognition in e-commerce using sequence-to-sequence hierarchical classification, in: Proc. of WSDM, 2021, pp. 902–905.

[18] H.-F. Yu, C.-H. Ho, P. Arunachalam, M. Somaiya, C.-J. Lin, Product title classification versus text classification, Csie. Ntu. Edu. Tw (2012) 1–25.

[19] Z. Kozareva, Everyone likes shopping! multi-class product categorization for e-commerce, in: Proc. of the NAACL-HLT, 2015, pp. 1329–1333.

[20] A. More, Attribute extraction from product titles in ecommerce, arXiv:1608.04670 (2016).

[21] D. Putthividhya, J. Hu, Bootstrapped named entity recognition for product attribute extraction, in: Proc. of EMNLP, 2011, pp. 1557–1567.

[22] L. Bing, T.-L. Wong, W. Lam, Unsupervised extraction of popular product attributes from e-commerce web sites by considering customer reviews, TOIT 16 (2016) 1–17.

[23] G. Zheng, S. Mukherjee, X. L. Dong, F. Li, Opentag: Open attribute value extraction from product profiles, in: Proc. of KDD, 2018, pp. 1049–1058.

[24] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv:1810.04805 (2018).

[25] A. Ahuja, N. Rao, S. Katariya, K. Subbian, C. K. Reddy, Language-agnostic representation learning for product search on e-commerce platforms, in: Proc. of WSDM, 2020, pp. 7–15.

[26] H. Hwangbo, Y. S. Kim, K. J. Cha, Recommendation system development for fashion retail e-commerce, ECRA 28 (2018) 94–101.

[27] L. Jiang, Y. Cheng, L. Yang, J. Li, H. Yan, X. Wang, A trust-based collaborative filtering algorithm for e-commerce recommendation system, JAIHC 10 (2019) 3023–3034.
[28] A. Mabrouk, R. P. D. Redondo, M. Kayed, Seopinion: Summarization and exploration of opinion from e-commerce websites, Sensors 21 (2021) 636.
[29] J. Chen, D. Warren, Cost-sensitive learning for large-scale hierarchical classification, in: Proc. of CIKM, 2013, pp. 1351–1360.
[30] T. Zhu, Y. Wang, H. Li, Y. Wu, X. He, B. Zhou, Multimodal joint attribute prediction and value extraction for e-commerce product, arXiv:2009.07162 (2020).
[31] I. Guy, T. Milo, S. Novgorodov, B. Youngmann, Improving constrained search results by data melioration, in: Proc. of ICDE, 2021, pp. 1667–1678.
[32] S. Novgorodov, I. Guy, G. Elad, K. Radinsky, Generating product descriptions from user reviews, in: Proc. of WWW, 2019, pp. 1354–1364.
[33] M.-T. Nguyen, P.-T. Nguyen, V.-V. Nguyen, Q.-M. Nguyen, Generating product description with generative pre-trained transformer 2, in: Proc. of CITISIA, 2021, pp. 1–7.
[34] S. Huang, X. Liu, X. Peng, Z. Niu, Fine-grained product features extraction and categorization in reviews opinion mining, in: Proc. of ICDM Workshops, 2012, pp. 680–686.
[35] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, Advances in neural information processing systems 32 (2019).
[36] X. Liu, A. Wangperawong, Transfer learning robustness in multi-class categorization by fine-tuning pre-trained contextualized language models, arXiv:1909.03564 (2019).
[37] P. Ristoski, P. Petrovski, P. Mika, H. Paulheim, A machine learning approach for product matching and categorization, Semantic web 9 (2018) 707–728.
[38] P. Wirojwatanakul, A. Wangperawong, Multi-label product categorization using multi-modal fusion models, arXiv:1907.00420 (2019).
[39] F. Yang, A. Kale, Y. Bubnov, L. Stein, Q. Wang, H. Kiapour, R. Piramuthu, Visual search at ebay, in: Proc. of SIGKDD, 2017, pp. 2101–2110.
[40] A. Goswami, N. Chittar, C. H. Sung, A study on the impact of product images on user clicks for online shopping, in: Proc. of WWW (Companion Volume), 2011, pp. 45–46.
[41] L. S. Paulucio, T. M. Paixão, R. F. Berriel, A. F. De Souza, C. Badue, T. Oliveira-Santos, Product categorization by title using deep neural networks as feature extractor, in: Proc. of IJCNN, 2020, pp. 1–7.
[42] H. Linmei, T. Yang, C. Shi, H. Ji, X. Li, Heterogeneous graph attention networks for semi-supervised short text classification, in: Proc. of EMNLP-IJCNLP, 2019, pp. 4821–4830.
[43] B. Škrlj, M. Martinc, J. Kralj, N. Lavrač, S. Pollak, tax2vec: Constructing interpretable features from taxonomies for short text classification, Computer Speech & Language 65 (2021) 101104.
[44] A. Spink, B. J. Jansen, A study of web search trends, Webology 1 (2004) 4.
[45] I. Guy, Searching by talking: Analysis of voice queries on mobile web search, in: Proc. of SIGIR, 2016, pp. 35–44.
[46] D. Shen, R. Pan, J.-T. Sun, J. J. Pan, K. Wu, J. Yin, Q. Yang, Query enrichment for web-query classification, TOIS 24 (2006) 320–352.
[47] D. Shen, J.-T. Sun, Q. Yang, Z. Chen, Building bridges for web query classification, in: Proc. of SIGIR, 2006, pp. 131–138.
[48] H. Cao, D. H. Hu, D. Shen, D. Jiang, J.-T. Sun, E. Chen, Q. Yang, Context-aware query classification, in: Proc. of SIGIR, 2009, pp. 3–10.
[49] S. Gandhi, B. Mansouri, R. Campos, A. Jatowt, Event-related query classification with deep neural networks, in: Proc. of WWW (Companion Volume), 2020, pp. 324–330.
[50] S. Kharbanda, A. Banerjee, A. Palrecha, R. Babbar, Embedding convolutions for short text extreme classification with millions of labels, arXiv:2109.07319 (2021).
[51] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, S. Fidler, Aligning books and movies: Towards story-like visual explanations by watching movies and reading

books, in: Proc. of ICCV, 2015, pp. 19–27.
[52] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (1997) 1735–1780.
[53] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv:1301.37810 (2013).
[54] Y. Xin, E. Hart, V. Mahajan, J.-D. Ruvini, Learning better internal structure of words for sequence labeling, arXiv:1810.12443 (2018).
[55] R. He, J. McAuley, Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering, in: Proc. of WWW, 2016, pp. 507–517.
[56] M. Grandini, E. Bagli, G. Visani, Metrics for multi-class classification: an overview, arXiv:2008.05756 (2020).
[57] L. Yao, C. Mao, Y. Luo, Kg-bert: Bert for knowledge graph completion, arXiv:1909.03193 (2019).
[58] Y. Tay, V. Q. Tran, M. Dehghani, J. Ni, D. Bahri, H. Mehta, Z. Qin, K. Hui, Z. Zhao, J. Gupta, et al., Transformer memory as a differentiable search index, arXiv:2202.06991 (2022).