Extending CLIP for Category-to-image Retrieval in E-commerce

Mariya Hendriksen[1], Maurits Bleeker[2], Svitlana Vakulenko[2]
Nanne van Noord[2], Ernst Kuiper[3], and Maarten de Rijke[2]

[1] AIRLab, University of Amsterdam
[2] University of Amsterdam
[3] Bol.com
m.hendriksen@uva.nl m.j.r.bleeker@uva.nl s.vakulenko@uva.nl
n.j.e.vannoord@uva.nl ekuiper@bol.com m.derijke@uva.nl

Abstract. E-commerce provides rich multimodal data that is barely leveraged in practice. One aspect of this data is a category tree that is being used in search and recommendation. However, in practice, during a user's session there is often a mismatch between a textual and a visual representation of a given category. Motivated by the problem, we introduce the task of category-to-image retrieval in e-commerce and propose a model for the task, CLIP-ITA. The model leverages information from multiple modalities (textual, visual, and attribute modality) to create product representations. We explore how adding information from multiple modalities (textual, visual, and attribute modality) impacts the model's performance. In particular, we observe that CLIP-ITA significantly outperforms a comparable model that leverages only the visual modality and a comparable model that leverages the visual and attribute modality.

Keywords: Multimodal retrieval · Category-to-image retrieval · E-commerce

1 Introduction

Multimodal retrieval is a major but understudied problem in e-commerce [33]. Even though e-commerce products are associated with rich multi-modal information, research currently focuses mainly on textual and behavioral signals to support product search and recommendation. The majority of prior work in multimodal retrieval for e-commerce focuses on applications in the fashion domain, such as recommendation of fashion items [21] and cross-modal fashion retrieval [6, 14]. In the more general e-commerce domain, multimodal retrieval has not been explored that well yet [10, 18]. The multimodal problem on which we focus is motivated by the importance of category information in e-commerce. Product category trees are a key component of modern e-commerce as they assist customers when navigating across large and dynamic product catalogues [13, 30, 36]. Yet, the ability to retrieve an image for a given product category remains a challenging task mainly due to noisy category and product

data, and the size and dynamic character of product catalogues [17, 33].

The category-to-image retrieval task. We introduce the problem of retrieving a ranked list of relevant images of products that belong to a given category, which we call the *category-to-image* (CtI) retrieval task. Unlike image classification tasks that operate on a predefined set of classes, in the CtI retrieval task we want to be able not only to understand which images belong to a given category but also to generalize towards unseen categories. Consider the category "Home decor." A CtI retrieval should output a ranked list of k images retrieved from the collection of images that are relevant to the category, which could be anything from images of carpets to an image of a clock or an arrangement of decorative vases. Use cases that motivate the CtI retrieval task include (1) the need to showcase different categories in search and recommendation results [13, 30, 33]; (2) the task can be used to infer product categories in the cases when product categorical data is unavailable, noisy, or incomplete [39]; and (3) the design of cross-categorical promotions and product category landing pages [24].

The CtI retrieval task has several key characteristics: (1) we operate with categories from non-fixed e-commerce category trees, which range from very general (such as "Automative" or "Home & Kitchen") to very specific ones (such as "Helmet Liners" or "Dehumidifiers"). The category tree is not fixed, therefore, we should be able to generalize towards unseen categories; and (2) product information is highly multimodal in nature; apart from category data, products may come with textual, visual, and attribute information.

A model for CtI retrieval. To address the CtI retrieval task, we propose a model that leverages image, text, and attribute information, CLIP-ITA. CLIP-ITA extends upon Contrastive Language-Image Pre-Training (CLIP) [26]. CLIP-ITA extends CLIP with the ability to represent attribute information. Hence, CLIP-ITA is able to use textual, visual, and attribute information for product representation. We compare the performance of CLIP-ITA with several baselines such as unimodal BM25, bimodal zero-shot CLIP, and MPNet [29]. For our experiments, we use the XMarket dataset that contains textual, visual, and attribute information of e-commerce products [2].

Research questions and contributions. We address the following research questions: (RQ1) How do baseline models perform on the CtI retrieval task? Specifically, how do unimodal and bi-modal baseline models perform? How does the performance differ w.r.t. category granularity? (RQ2) How does a model, named CLIP-I, that uses product image information for building product representations impact the performance on the CtI retrieval task? (RQ3) How does CLIP-IA, which extends CLIP-I with product attribute information, perform on the CtI retrieval task? (RQ4) And finally, how does CLIP-ITA, which extends CLIP-IA with product text information, perform on the CtI task?

Our main contributions are: (1) We introduce the novel task of CtI retrieval and motivate it in terms of e-commerce applications. (2) We propose CLIP-ITA, the first model specifically designed for this task. CLIP-ITA leverages multimodal product data such as textual, visual, and attribute data. On average, CLIP-ITA outperforms CLIP-I on all categories by 217% and CLIP-IA by 269%. We share

our code and experimental settings to facilitate reproducibility of our results.[4]

2 Related Work

Learning multimodal embeddings. Contrastive pre-training has been shown to be highly effective in learning joined embeddings across modalities [26]. By predicting the correct pairing of image-text tuples in a batch, the CLIP model can learn strong text and image encoders that project to joint space. This approach to learning multimodal embeddings offers key advantages over approaches that use manually assigned labels as supervision: (1) the training data can be collected without manual annotation; real-world data in which image-text pairs occur can be used; (2) models trained in this manner learn more general representations that allow for zero-shot prediction. These advantages are appealing for e-commerce, as most public multimodal e-commerce datasets primarily focus on fashion only [2]; being able to train from real-world data avoids the need for costly data annotation.

We build on CLIP by extending it to category-product pairs, taking advantage of its ability to perform zero-shot retrieval for a variety semantic concepts.

Multimodal image retrieval. Early work in image retrieval grouped images into a restricted set of semantic categories and allowed users to retrieve images by using category labels as queries [28]. Later work allowed for a wider variety of queries ranging from natural language [11, 34], to attributes [23], to combinations of multiple modalities (e.g., title, description, and tags) [32]. Across these multimodal image retrieval approaches we find three common components: (1) an image encoder, (2) a query encoder, and (3) a similarity function to match the query to images [7, 26]. Depending on the focus of the work some components might be pre-trained, whereas the others are optimized for a specific task.

In our work, we rely on pre-trained image and text encoders but learn a new multimodal composite of the query to perform CtI retrieval.

Multimodal retrieval in e-commerce. Prior work on multimodal retrieval in e-commerce has been mainly focused on cross-modal retrieval for fashion [6, 16, 42]. Other related examples include outfit recommendation [15, 19, 21] Some prior work on interpretability for fashion product retrieval proposes to leverage multimodal signals to improve explainability of latent features [20, 38]. Tautkute et al. [31] propose a multimodal search engine for fashion items and furniture. When it comes to combining signals for improving product retrieval, Yim et al. [40] propose to combine product images, titles, categories, and descriptions to improve product search, Yamaura et al. [37] propose an algorithm that leverages multimodal product information for predicting a resale price of a second-hand product.

Unlike prior work on multimodal retrieval in e-commerce that mainly focuses on fashion data, we focus on creating multimodal product representations for the general e-commerce domain.

[4] `https://github.com/mariyahendriksen/ecir2022_category_to_image_retrieval`

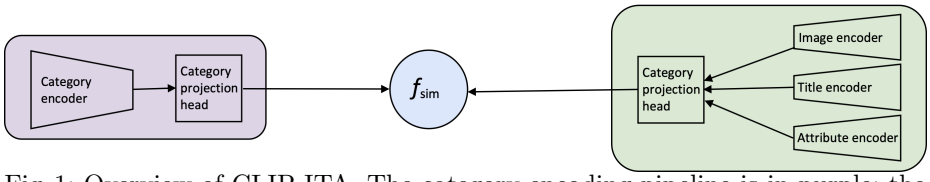


Fig. 1: Overview of CLIP-ITA. The category encoding pipeline is in purple; the category information pipeline in green; f_{sim} is a cosine similarity function.

3 Approach

Task definition. We follow the same notation as in [41]. The input dataset can be presented as category-product pairs $(\mathbf{x}_c, \mathbf{x}_p)$, where $\mathbf{x}_c$ represents a product category, and $\mathbf{x}_p$ represents information about product that belong to the category $\mathbf{x}_c$. The product category $\mathbf{x}_c$ is taken from the category tree T and is represented as a category name. The product information comprises titles $\mathbf{x}_t$, images $\mathbf{x}_i$, and attributes $\mathbf{x}_i$, i.e., $\mathbf{x}_p = \{\mathbf{x}_i, \mathbf{x}_t, \mathbf{x}_a\}$.

For the CtI retrieval task, we use the target category name $\mathbf{x}_c$ as a query and we aim to refturn a ranked list of top-k images that belong to the category $\mathbf{x}_c$.

CLIP-ITA. Fig. 1 provides a high-level view of CLIP-ITA. CLIP-ITA projects category $\mathbf{x}_c$ and product information $\mathbf{x}_p$ into a d-dimensional multimodal space where the resulting vectors are respectively $\mathbf{c}$ and $\mathbf{p}$. The category and product information is processed by a category encoding pipeline and product information encoding pipeline. The core components of CLIP-ITA are the encoding and projection modules. The model consists out of four encoders: a category encoder, an image encoder, a title encoder, and an attribute encoder. Besides, CLIP-ITA comprises two non-linear projection heads: the category projection head and the multimodal projection head.

While several components of CLIP-ITA are based on CLIP [26], CLIP-ITA differs from CLIP in three important ways: (1) unlike CLIP, which operates on two encoders (textual and visual), CLIP-ITA extends CLIP towards a category encoder, image encoder, textual encoder, and attribute encoder; (2) CLIP-ITA features two projection heads, one for the category encoding pipeline, and one for the product information encoding pipeline; and (3) while CLIP is trained on text-image pairs, CLIP-ITA is trained on category-product pairs, where product representation is multimodal.

Category encoding pipeline. The *category encoder* (f_c) takes as input category name $\mathbf{x}_c$ and returns its representation $\mathbf{h}_c$. More specifically, we pass the category name $\mathbf{x}_c$ through the category encoder f_c:

$$\mathbf{h}_c = f_c(\mathbf{x}_c). \tag{1}$$

To obtain this representation, we use pre-trained MPNet model [29]. After passing category information through the category encoder, we feed it to the category projection head. The *category projection head* (g_c) takes as input a query representation $\mathbf{h}_c$ and projects it into d-dimensional multi-modal space:

$$\mathbf{c} = g_c(\mathbf{h}_c), \tag{2}$$

where $\mathbf{c} \in \mathbb{R}^d$.

Product encoding pipeline. The product information encoding pipeline represents three encoders, one for every modality, and a product projection head. The *image encoder* (f_i) takes as input a product image $\mathbf{x}_i$ aligned with the category $\mathbf{x}_c$. Similarly to the category processing pipeline, we pass the product image $\mathbf{x}_i$ through the image encoder:

$$\mathbf{h}_i = f_i(\mathbf{x}_i). \tag{3}$$

To obtain the image representation $\mathbf{h}_i$, we use pre-trained Vision Transformer from CLIP model. The *title encoder* (f_t) takes a product title $\mathbf{x}_t$ as input and returns a title representation $\mathbf{h}_t$:

$$\mathbf{h}_t = f_t(\mathbf{x}_t). \tag{4}$$

Similarly to the category encoder f_c, we use pre-trained MPNet to obtain the title representation $\mathbf{h}_t$. The *attribute encoder* (f_a) is a network that takes as input a set of attributes $\mathbf{x}_a = \{a_1, a_2, \ldots, a_n\}$ and returns their joint representation:

$$\mathbf{h}_a = f_a(\mathbf{x}_a) = \frac{1}{n}\sum_{i=1}^{n} f_a(\mathbf{x}_{ai}). \tag{5}$$

Similarly to the category encoder f_c and title encoder f_t, we obtain representation of each attribute with the pre-trained MPNet model. After obtaining title, image and attribute representations, we pass the representations into the product projection head. The *product projection head* (g_p) takes as input a concatenation of the image representation $\mathbf{h}_i$, title representation $\mathbf{h}_t$, and attribute representation $\mathbf{h}_a$ and projects the resulting vector $\mathbf{h}_p = concat(\mathbf{h}_i, \mathbf{h}_t, \mathbf{h}_a)$ into multimodal space:

$$\mathbf{p} = g_p(\mathbf{h}_p) = g_p(concat(\mathbf{h}_i, \mathbf{h}_t, \mathbf{h}_a)), \tag{6}$$

where $\mathbf{p} \in \mathbb{R}^d$.

Loss function. We train CLIP-ITA using bidirectional contrastive loss [41]. The loss is a weighted combination of two losses: a category-to-product contrastive loss and a product-to-category contrastive loss. In both cases the loss is the InfoNCE loss [25]. Unlike prior work that focuses on a contrastive loss between inputs of the same modality [3, 8] and on corresponding inputs of two modalities [41], we use the loss to work with inputs from textual modality (category representation) vs. a combination of multiple modalities (product representation). We train CLIP-ITA on batches of category-product pairs $(\mathbf{x}_c, \mathbf{x}_p)$ with batch size β. For the j-th pair in the batch, the category-to-product contrastive loss is computed as follows:

$$\ell_j^{(c \to p)} = -\log \frac{\exp(f_{sim}(\mathbf{c}_j, \mathbf{p}_j)/\tau)}{\sum_{k=1}^{\beta} \exp(f_{sim}(\mathbf{c}_j, \mathbf{p}_k)/\tau)}, \tag{7}$$

where $f_{sim}(\mathbf{c}_i, \mathbf{p}_i)$ is the cosine similarity, and $\tau \in \mathbb{R}^+$ is a temperature parameter. Similarly, the product-to-category loss is computed as follows:

$$\ell_j^{(p \to c)} = -\log \frac{\exp(f_{sim}(\mathbf{p}_j, \mathbf{c}_j)/\tau)}{\sum_{k=1}^{\beta} \exp(f_{sim}(\mathbf{p}_j, \mathbf{c}_k)/\tau)}. \tag{8}$$

The resulting contrastive loss is a combination of the two above-mentioned losses:

$$\mathcal{L} = \frac{1}{\beta} \sum_{j=1}^{\beta} \left(\lambda \ell_j^{(p \to c)} + (1 - \lambda) \ell_j^{(c \to p)} \right), \tag{9}$$

where β represents the batch size and $\lambda \in [0, 1]$ is a scalar weight.

4 Experimental Setup

Dataset. We use the XMarket dataset recently introduced by Bonab et al. [2] that contains textual, visual, and attribute information of e-commerce products as well as a category tree. For our experiments, we select 38,921 products from the US market. Category information is represented as a category tree and comprises 5,471 unique categories across nine levels. Level one is the most general category level, level nine is the most specific level. Every product belongs to a subtree of categories $t \in T$. In every subtree t, each parent category has only one associated child category. The average subtree depth is 4.63 (minimum: 2, maximum: 9). Because every product belongs to a subtree of categories, the dataset contains 180,094 product-category pairs in total. We use product titles as textual information and one image per product as visual information. The attribute information comprises 228,368 attributes, with 157,049 unique. On average, every product has 5.87 attributes (minimum: 1, maximum: 24).

Evaluation method. To investigate how model performance changes w.r.t. category granularity, for every product in the dataset, $\mathbf{x}_p$, and the corresponding subtree of categories to which the product belongs, t, we train and evaluate the model performance in three settings: (1) *all categories*, where we randomly select one category from the subtree t; (2) *most general category*, where we use only the most general category of the subtree t, i.e., the root; and (3) *most specific category*, where we use the most specific category of the subtree t. In total, there are 5,471 categories in all categories setup, 34 categories in the most general category, and 4,100 in the most specific category setup. We evaluate every model on category-product pairs $(\mathbf{x}_c, \mathbf{x}_p)$ from the test set. We encode each category and a candidate product data by passing them through category encoding and product information encoding pipelines. For every category $\mathbf{x}_c$ we retrieve the top-k candidates ranked by cosine similarity w.r.t. the target category $\mathbf{x}_c$.

Metrics. To evaluate model performance, we use Precision@K where $K = \{1, 5, 10\}$, mAP@K where $K = \{5, 10\}$, and R-precision.

Baselines. Following [4, 27, 35] we use BM25, MPNet, CLIP as our baselines.

Four experiments. We run four experiments, corresponding to our research

questions as listed at the end of Section 1. In *Experiment 1* we evaluate the baselines on the CtI retrieval task (RQ1). We feed BM25 corpora that contain textual product information, i.e., product titles. We use MPNet in a zero-shot manner. For all the products in the dataset, we pass the product title $\mathbf{x}_t$ through the model. During the evaluation, we pass a category $\mathbf{x}_c$ expressed as textual query through MPNet and retrieve top-k candidates ranked by cosine similarity w.r.t. the target category $\mathbf{x}_c$. We compare categories of the top-k retrieved candidates with the target category $\mathbf{x}_c$. Besides, we use pre-trained CLIP in a zero-shot manner with a Text Transformer and a Vision Transformer (ViT) [5] an configuration. We pass the product images $\mathbf{x}_i$ through the image encoder. For evaluation, we pass a category $\mathbf{x}_c$ through the text encoder and retrieve top-k image candidates ranked by cosine similarity w.r.t. the target category $\mathbf{x}_c$. We compare categories of the top-k retrieved images with the target category $\mathbf{x}_c$.

In *Experiment 2* we evaluate image-based product representations (RQ2). After obtaining results with CLIP in a zero-shot setting, we build product representations by training on e-commerce data. First, we investigate how using product image data for building product representations impacts performance on the CtI retrieval task. To introduce visual information, we extend CLIP in two ways: (1) We use ViT from CLIP as image encoder f_i. We add product projection head g_p that takes as an input product visual information $\mathbf{x}_i \in \mathbf{x}_p$. (2) We use the text encoder from MPNet as category encoder f_c; we add a category projection head g_c on top of category encoder f_c thereby completing category encoding pipeline (see Fig. 1). We name the resulting model CLIP-I. We train CLIP-I on category-product pairs $(\mathbf{x}_c, \mathbf{x}_p)$ from the training set. Note that $\mathbf{x}_p = \{\mathbf{x}_i\}$, i.e., we only use visual information for building product representations.

In *Experiment 3*, we evaluate image- and attribute-based product representations (RQ3). We extend CLIP-I by introducing attribute information to the product information encoding pipeline. We add an attribute encoder f_a through which we obtain a representation of product attributes, $\mathbf{h}_a$. We concatenate the resulting attribute representation with image representation $\mathbf{h}_p = concat(\mathbf{h}_i, \mathbf{h}_a)$ and pass the resulting vector to the product projection head g_p. Thus, the resulting product representation $\mathbf{p}$ is based on both visual and attribute product information. We name the resulting model CLIP-IA. We train CLIP-IA on category-product pairs $(\mathbf{x}_c, \mathbf{x}_p)$ where $\mathbf{x}_p = \{\mathbf{x}_i, \mathbf{x}_a\}$, i.e., we use visual and attribute information for building product representation.

In *Experiment 4*, we evaluate image- attribute-, and title-based product representations (RQ4). We investigate how extending the product information processing pipeline with the textual modality impacts performance on the CtI retrieval task. We add title encoder f_t to the product information processing pipeline and use it to obtain title representation $\mathbf{h}_t$. We concatenate the resulting representation with product image and attribute representations $\mathbf{h}_p = concat(\mathbf{h}_i, \mathbf{h}_t, \mathbf{h}_a)$. We pass the resulting vector to the product projection head g_p. The resulting model is CLIP-ITA. We train and test CLIP-ITA on category-product pairs $(\mathbf{x}_c, \mathbf{x}_p)$ where $\mathbf{x}_p = \{\mathbf{x}_i, \mathbf{x}_a, \mathbf{x}_t\}$, i.e., we use visual, attribute, and textual information for building product representations.

Table 1: Results of Experiments 1–4. The best performance is highligthed in bold.

Model	P@1	P@5	P@10	MAP@5	MAP@10	R-precision
All categories (5,471)						
BM25 [12]	0.01	0.01	0.01	0.01	0.01	0.01
CLIP [26]	0.01	0.02	0.02	0.03	0.04	0.02
MPNet [29]	0.01	0.06	0.06	0.07	0.09	0.05
CLIP-I (Ours)	3.3	3.8	3.79	6.81	7.25	3.67
CLIP-IA (Ours)	2.5	3.34	3.29	5.95	6.24	3.27
CLIP-ITA (Ours)	**9.9**	**13.27**	**13.43**	**20.3**	**20.53**	**13.42**
Most general category (34)						
BM25 [12]	2.94	4.71	4.71	8.33	8.28	4.48
CLIP [26]	11.76	12.35	11.76	16.12	15.18	9.47
MPNet [29]	14.70	15.8	15.01	18.44	18.78	9.35
CLIP-I (Ours)	17.85	17.14	16.78	19.88	20.14	13.02
CLIP-IA (Ours)	21.42	21.91	22.78	25.59	26.29	20.74
CLIP-ITA (Ours)	**35.71**	**30.95**	**30.95**	**35.51**	**34.28**	**25.79**
Most specific category (4,100)						
BM25 [12]	0.02	0.02	0.01	0.01	0.01	0.01
CLIP [26]	11.92	9.81	9.23	15.12	14.95	8.14
MPNet [29]	33.36	28.56	26.93	37.43	36.77	25.29
CLIP-I (Ours)	14.06	12.11	11.53	18.24	17.9	11.22
CLIP-IA (Ours)	35.3	30.21	29.32	39.93	39.27	28.86
CLIP-ITA (Ours)	**45.85**	**41.04**	**40.02**	**50.04**	**49.87**	**39.69**

Implementation details. We train every model for 30 epochs, with a batch size $\beta = 8$ for most general categories, $\beta = 128$ — for most specific categories and all categories. For loss function, we set $\tau = 1$, $\lambda = 0.5$. We implement every projection head as non-linear MLPs with two hidden layers, GELU non-linearities [9] and layer normalization [1]. We optimize both heads with the AdamW optimizer [22].

5 Experimental results

Experiment 1: Baselines. Following RQ1, we start by investigating how do baselines perform on CtI retrieval task. Besides, we investigate how does the performance on the task differs between the unimodal and the bimodal approach.

The results are shown in Table 1. When evaluating on all categories, all the baselines perform poorly. For the most general category setting, MPNet outperforms CLIP on all metrics except R-precision. The most prominent gain is for Precision@10 where MPNet outperforms CLIP by 28%. CLIP outperforms BM25 on all metrics. For the most specific category setting, MPNet performance is the highest, BM25 — the lowest. In particular, MPNet outperforms CLIP by 211% in Precision@10. Overall, MPNet outperforms CLIP and both models sig-

nificantly outperforms BM25 for both most general and most specific categories. However, when evaluation is done on all categories, the performance of all models is comparable. As an answer to RQ1, the results suggest that using information from multiple modalities is beneficial for performance on the task.

Experiment 2: Image-based product representations. To address RQ2, we compare the performance of CLIP-I with CLIP and MPNet, the best-performing baseline. Table 1, shows the experimental results for Experiment 2. The biggest performance gains are obtained in "all categories" setting. However, there, the performance of the baselines was very poor. For the most general categories, CLIP-I outperforms both CLIP and MPNet. For CLIP-I vs. CLIP, we observe the biggest increase of 51% for Precision@1, for CLIP-I vs. MPNet — 39% in R-precision. In the case of the most specific categories, CLIP-I outperforms CLIP but loses to MPNet. Overall, CLIP-I outperforms CLIP in all three settings and outperforms MPNet except the most specific categories. Therefore, we answer RQ2 as follows: the results suggest that extension of CLIP by the introduction of product image data for building product representations has a positive impact on performance on CtI retrieval task.

Experment 3: Image- and attribute-based product representations. To answer RQ3, we compare the performance of CLIP-IA with CLIP-I and the baselines. The results are shown in Table 1. When evaluated on all categories, CLIP-IA performs worse than CLIP-I but outperforms MPNet. In particular, CLIP-I obtains the biggest gain relative of 32% on Precision@1 and the lowest gain of 12% on R-precision. For the most general category, CLIP-IA outperforms CLIP-I and MPNet on all metrics. More specifically, we observe the biggest gain of 122% on R-precision over MPNet and the biggest gain of 59% on R-precision for CLIP-I. Similarly, for the most specific category, CLIP-IA outperforms both CLIP-I and MPNet. We observe the biggest relative gain of 138% over CLIP-I. The results suggest that further extension of CLIP by the introduction of the product image and attribute data for building product representations has a positive impact on performance on CtI retrieval task, especially when evaluated on most specific categories. Therefore, we answer RQ4 positively.

Experiment 4: Image-, attribute-, and title-based product representations. We compare CLIP-ITA with both CLIP-IA, CLIP-I, and the baselines. The results are shown in Table 1. In general, CLIP-ITA outperforms CLIP-I and CLIP-IA and the baselines in all settings. When evaluated on all categories, the maximum relative increase of CLIP-ITA over CLIP-I is 265% in R-precision, the minimum relative increase is 183% in mAP@10. The biggest relative increase of CLIP-ITA performance over CLIP-IA is 310% in Precision@1, the smallest relative increase is 229% in mAP@10. For the most general categories, CLIP-ITA outperforms CLIP-I by 82% and CLIP-IA by 38%. For most specific categories, we observe the biggest increase of CLIP-ITA over CLIP-I of 254% in R-precision and the smallest relative increase of 172% on mAP@5. At the same time, the biggest relative increase of CLIP-ITA over CLIP-IA is a 38% increase in R-precision and the smallest relative increase is a 27% increase in mAP@5. Overall, CLIP-ITA wins in all three settings. Hence, we answer RQ4 positively.

Table 2: Erroneous CLIP-ITA prediction counts for "same tree" vs. " different tree" predictions per evaluation type.

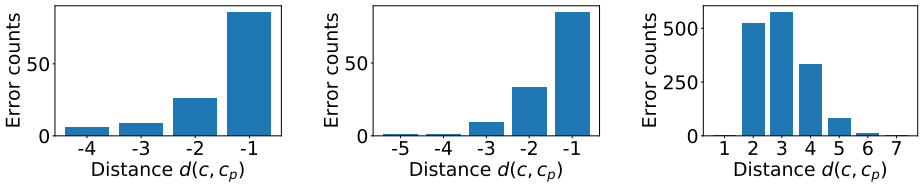
	Same tree	**Different tree**
All categories	1,655	639
The most general category	2	21
The most specific category	127	1,011
Total	1,786	1,671

6 Error Analysis

Distance between predicted and target categories. We examine the performance of CLIP-ITA by looking at the pairs of the ground-truth and predicted categories (c, c_p) in cases when the model failed to predict the correct category, i.e., $c \neq c_p$. This allows us to quantify how far off the incorrect predictions lie w.r.t. the category tree hierarchy. First, we examine in how many cases target category c and predicted category c_p belong to the same most general category, i.e., belong to the same category tree; see Table 2. In the case of most general categories, the majority of incorrectly predicted categories belong to a tree different from the target category tree. For the most specific categories, about 11% of predicted categories belong to the category tree of the target category. However, when evaluation is done on all categories, 72% of incorrectly predicted cases belong to the same tree as a target category.

Next, we turn to the category-predicted category pairs (c, c_p) where the incorrectly predicted category c_p belongs to the same tree as target category c. We compute the distance d between a category used as a query c and a predicted category c_p. We compute the distance between target category c and a top-1 predicted category c_p as the difference between their respective depths $d(c, c_p) = \mathit{depth}(c_p) - \mathit{depth}(c)$. The distance d is positive if the depth of the predicted category is bigger than the depth of the target category, $\mathit{depth}(c_p) > \mathit{depth}(c)$, i.e., the predicted category is more specific than the target category. The setup is mirrored for negative distances. See Fig. 2. We do not plot the results for the most general category because for this setting there are only two cases when target category c and a predicted category c_p were in the same tree. In both cases, predicted category c_p was more general than target category c with distance $d(c, p_c) = 2$. In cases when target category c was sampled from the most specific categories, the wrongly predicted category c_p belonging to the same tree was always more specific than the target category c with the maximum absolute distance between c and c_p, $|d(c, c_p)| = 4$. In 68% of the cases the predicted category was one level above the target category, for 21% $d(c, c_p) = -2$, for 7% $d(c, c_p) = -3$, and for 5% $d(c, c_p) = -4$. For the setting with all categories, in 92% of the cases, the predicted category c_p was more specific than the target category c; for 8% the predicted category was more general.

Overall, for the most general category and the most specific category, the majority of incorrectly predicted categories are located in a category tree different from the one where the target category was located. For the "all categories"



(a) Most specific categories (b) All categories, $d < 0$ (c) All categories, $d > 0$

Fig. 2: Error analysis for CLIP-ITA. Distance between target category c and a predicted category c_p when c and c_p are in the same tree.

setting, it is the other way around. When it comes to the cases when incorrectly predicted categories are in the same tree as a target category, the majority of incorrect predictions are 1 level more general when the target category is sampled from the most specific categories. For the "all categories" setting, the majority of incorrect predictions belonging to the same tree as the target category were more specific than the target category. Our analysis suggests that efforts to improve the performance of CLIP-ITA should focus on minimizing the (tree-based) distance between the target and predicted category in a category tree. This could be incorporated as a suitable extension of the loss function.

Performance on seen vs. unseen categories. Next, we investigate how well CLIP-ITA generalizes to unseen categories. We split the evaluation results into two groups based on whether the category used as a query was seen during training or not; see Table 3. For the most general categories, CLIP-ITA is unable to correctly retrieve an image of the product of the category that was not seen during training at all. For the most specific categories, CLIP-ITA performs better on seen categories than on unseen categories. We observe the biggest relative performance increase of 85% in mAP@10 and the smallest relative increase of 57% in R-precision. When evaluating on all categories, CLIP-ITA performs on unseen categories better when evaluated on Precision@k (27% higher in Precision@1, 33% higher in Precision@5, 10% increase in Precision@10) and R-precision (relative increase of 32%). Performance on seen categories is better in terms of mAP@k (10% increase for both mAP@5 and mAP@10).

Overall, for the most general and most specific categories, the model performs much better on categories seen during training. For "all categories" setting, however, CLIP-ITA's performance on unseen categories is better.

7 Conclusion

We introduced the task of category-to-image retrieval and motivated its importance in the e-commerce scenario. In the CtI retrieval task, we aim to retrieve an image of a product that belongs to the target category. We proposed a model specifically designed for this task, CLIP-ITA. CLIP-ITA extends CLIP, one of the best performing text-image retrieval models. CLIP-ITA leverages multimodal product data such as textual, visual, and attribute data to build product representations. In our experiments, we contrasted and evaluated different combinations of signals from modalities, using three settings: on all categories, the most

Table 3: CLIP-ITA performance on seen vs. unseen categories.

Model	P@1	P@5	P@10	mAP@5	mAP@10	R-precision
			All categories (5,471)			
CLIP-ITA (unseen cat.)	13.3	18.56	15.55	19.7	19.65	18.52
CLIP-ITA (seen cat.)	10.48	13.95	14.08	21.65	21.65	14.07
			Most general category (34)			
CLIP-ITA (unseen cat.)	0.0	0.0	0.0	0.0	0.0	0.0
CLIP-ITA (seen cat.)	19.23	20.01	17.31	20.41	20.01	15.73
			Most specific category (4,100)			
CLIP-ITA (unseen cat.)	27.27	26.44	26.44	27.92	27.92	26.45
CLIP-ITA (seen cat.)	47.83	43.09	42.14	52.41	51.89	41.58

general, and the most specific categories.

We found that combining information from multiple modalities to build product representation produces the best results on the CtI retrieval task. CLIP-ITA gives the best performance both on all categories and on the most specific categories. On the most general categories, CLIP-I, a model where product representation is based on image only, works slightly better. CLIP-I performs worse on the most specific categories and across all categories. For identification of the most general categories, visual information is more relevant. Besides, CLIP-ITA is able to generalize to unseen categories except in the case of most general categories. However, the performance on unseen categories is lower than the performance on seen categories. Even though our work is focused on the e-commerce domain, the findings can be useful for other areas, e.g., digital humanities.

Limitations of our work are due to type of data in the e-commerce domain. In e-commerce, there is typically one object per image and the background is homogeneous, textual information is lengthly and noisy; in the general domain, there is typically more than one object per image, image captions are more informative and shorter. Future work directions can focus on improving the model architecture. It would be interesting to incorporate attention mechanisms into the attribute encoder and explore how it influences performance. Another interesting direction for future work is to evaluate CLIP-ITA on other datasets outside of the e-commerce domain. Future work can also focus on minimizing the distance between the target and predicted category in the category tree.

Acknowledgements. This research was supported by Ahold Delhaize, the Nationale Politie, and the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, `https://hybrid-intelligence-centre.nl`.

All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

Bibliography

[1] Ba JL, Kiros JR, Hinton GE (2016) Layer normalization. arXiv preprint arXiv:160706450

[2] Bonab H, Aliannejadi M, Vardasbi A, Kanoulas E, Allan J (2021) XMarket: Cross-market training for product recommendation. In: CIKM, ACM

[3] Chen T, Kornblith S, Norouzi M, Hinton G (2020) A simple framework for contrastive learning of visual representations. In: International conference on machine learning, PMLR, pp 1597–1607

[4] Dai Z, Lai G, Yang Y, Le QV (2020) Funnel-transformer: Filtering out sequential redundancy for efficient language processing. arXiv preprint arXiv:200603236

[5] Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N (2021) An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations

[6] Goei K, Hendriksen M, de Rijke M (2021) Tackling attribute fine-grainedness in cross-modal fashion search with multi-level features. In: SIGIR 2021 Workshop on eCommerce, ACM

[7] Gupta T, Vahdat A, Chechik G, Yang X, Kautz J, Hoiem D (2020) Contrastive learning for weakly supervised phrase grounding. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16, Springer, pp 752–768

[8] He K, Fan H, Wu Y, Xie S, Girshick R (2020) Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 9729–9738

[9] Hendrycks D, Gimpel K (2016) Gaussian error linear units (GELUs). arXiv preprint arXiv:160608415

[10] Hewawalpita S, Perera I (2019) Multimodal user interaction framework for e-commerce. In: 2019 International Research Conference on Smart Computing and Systems Engineering (SCSE), IEEE, pp 9–16

[11] Hu R, Xu H, Rohrbach M, Feng J, Saenko K, Darrell T (2016) Natural language object retrieval. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 4555–4564

[12] Jones KS, Walker S, Robertson SE (2000) A probabilistic model of information retrieval: development and comparative experiments: Part 2. Information processing & management 36(6):809–840

[13] Kondylidis N, Zou J, Kanoulas E (2021) Category aware explainable conversational recommendation. arXiv preprint arXiv:210308733

[14] Laenen K, Moens MF (2019) Multimodal neural machine translation of fashion e-commerce descriptions. In: International Conference on Fashion communication: between tradition and future digital developments, Springer, pp 46–57

[15] Laenen K, Moens MF (2020) A comparative study of outfit recommendation methods with a focus on attention-based fusion. Information Processing & Management 57(6):102316

[16] Laenen K, Zoghbi S, Moens MF (2017) Cross-modal search for fashion attributes. In: Proceedings of the KDD 2017 Workshop on Machine Learning Meets Fashion, ACM, vol 2017, pp 1–10

[17] Laenen K, Zoghbi S, Moens MF (2018) Web search of fashion items with multimodal querying. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp 342–350

[18] Li H, Yuan P, Xu S, Wu Y, He X, Zhou B (2020) Aspect-aware multimodal summarization for chinese e-commerce products. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 34, pp 8188–8195

[19] Li X, Wang X, He X, Chen L, Xiao J, Chua TS (2020) Hierarchical fashion graph network for personalized outfit recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 159–168

[20] Liao L, He X, Zhao B, Ngo CW, Chua TS (2018) Interpretable multimodal retrieval for fashion products. In: Proceedings of the 26th ACM international conference on Multimedia, pp 1571–1579

[21] Lin Y, Ren P, Chen Z, Ren Z, Ma J, de Rijke M (2019) Improving outfit recommendation with co-supervision of fashion generation. In: The World Wide Web Conference, pp 1095–1105

[22] Loshchilov I, Hutter F (2017) Decoupled weight decay regularization. arXiv preprint arXiv:171105101

[23] Nagarajan T, Grauman K (2018) Attributes as operators: factorizing unseen attribute-object compositions. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 169–185

[24] Nielsen J, Molich R, Snyder C, Farrell S (2000) E-commerce user experience. Nielsen Norman Group

[25] Oord Avd, Li Y, Vinyals O (2018) Representation learning with contrastive predictive coding. arXiv preprint arXiv:180703748

[26] Radford A, Kim JW, Hallacy C, Ramesh A, Goh G, Agarwal S, Sastry G, Askell A, Mishkin P, Clark J, et al. (2021) Learning transferable visual models from natural language supervision. arXiv preprint arXiv:210300020

[27] Shen S, Li LH, Tan H, Bansal M, Rohrbach A, Chang KW, Yao Z, Keutzer K (2021) How much can CLIP benefit vision-and-language tasks? arXiv preprint arXiv:210706383

[28] Smeulders A, Worring M, Santini S, Gupta A, Jain R (2000) Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(12):1349–1380

[29] Song K, Tan X, Qin T, Lu J, Liu TY (2020) MPNet: Masked and permuted pre-training for language understanding. arXiv preprint arXiv:200409297

[30] Tagliabue J, Yu B, Beaulieu M (2020) How to grow a (product) tree: personalized category suggestions for ecommerce type-ahead. arXiv preprint arXiv:200512781

[31] Tautkute I, Trzciński T, Skorupa AP, Brocki Ł, Marasek K (2019) Deepstyle: Multimodal search engine for fashion and interior design. IEEE Access 7:84613–84628

[32] Thomee B, Shamma DA, Friedland G, Elizalde B, Ni K, Poland D, Borth D, Li LJ (2016) YFCC100M: The new data in multimedia research. Communications of the ACM 59(2):64–73

[33] Tsagkias M, King TH, Kallumadi S, Murdock V, de Rijke M (2020) Challenges and research opportunities in ecommerce search and recommendations. SIGIR Forum 54(1)

[34] Vo N, Jiang L, Sun C, Murphy K, Li LJ, Fei-Fei L, Hays J (2019) Composing text and image for image retrieval-an empirical odyssey. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6439–6448

[35] Wang S, Zhuang S, Zuccon G (2021) Bert-based dense retrievers require interpolation with bm25 for effective passage retrieval. In: Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval, pp 317–324

[36] Wirojwatanakul P, Wangperawong A (2019) Multi-label product categorization using multi-modal fusion models. arXiv preprint arXiv:190700420

[37] Yamaura Y, Kanemaki N, Tsuboshita Y (2019) The resale price prediction of secondhand jewelry items using a multi-modal deep model with iterative co-attention. arXiv preprint arXiv:190700661

[38] Yang X, He X, Wang X, Ma Y, Feng F, Wang M, Chua TS (2019) Interpretable fashion matching with rich attributes. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp 775–784

[39] Yashima T, Okazaki N, Inui K, Yamaguchi K, Okatani T (2016) Learning to describe e-commerce images from noisy online data. In: Asian Conference on Computer Vision, Springer, pp 85–100

[40] Yim J, Kim JJ, Shin D (2018) One-shot item search with multimodal data. arXiv preprint arXiv:181110969

[41] Zhang Y, Jiang H, Miura Y, Manning CD, Langlotz CP (2020) Contrastive learning of medical visual representations from paired images and text. arXiv preprint arXiv:201000747

[42] Zoghbi S, Heyman G, Gomez JC, Moens MF (2016) Cross-modal fashion search. In: International Conference on Multimedia Modeling, Springer, pp 367–373