MULTI-MODAL JOINT EMBEDDING FOR FASHION PRODUCT RETRIEVAL

A. Rubio[1, 2]   LongLong Yu[2]   E. Simo-Serra[3]   F. Moreno-Noguer[1]

[1]Institut de Robòtica i Informàtica Industrial (CSIC-UPC)
[2]Wide Eyes Technologies
[3]Waseda University
Email: arubio@iri.upc.edu, longyu@wide-eyes.it, esimo@aoni.waseda.jp, fmoreno@iri.upc.edu

ABSTRACT

Finding a product in the fashion world can be a daunting task. Everyday, e-commerce sites are updating with thousands of images and their associated metadata (textual information), deepening the problem, akin to finding a needle in a haystack. In this paper, we leverage both the images and textual metadata and propose a joint multi-modal embedding that maps both the text and images into a common latent space. Distances in the latent space correspond to similarity between products, allowing us to effectively perform retrieval in this latent space, which is both efficient and accurate. We train this embedding using large-scale real world e-commerce data by both minimizing the similarity between related products and using auxiliary classification networks to that encourage the embedding to have semantic meaning. We compare against existing approaches and show significant improvements in retrieval tasks on a large-scale e-commerce dataset. We also provide an analysis of the different metadata.

Index Terms— Multi-modal embedding, neural networks, retrieval

1. INTRODUCTION

The level of traffic of modern e-commerce is growing fast. U.S. retail e-commerce, for instance, was expected to grow 16.6% on 2016 Christmas holidays (after a 15.3% increase in 2014), with 92% of the holiday shoppers going online to search or buy gifts [1]. In order to adapt to these trend, modern retail sellers have to provide an easy-to-use experience to their customers, where products are easy to find and well classified. In this work, we consider the problem of multi-modal retrieval, in which a user searches for either text or images given a text or image query, and propose a joint embedding for this task.

E-commerce products usually consist of pictures and associated metadata, generally in the form of textual information such as brief descriptions, titles, series of tags,

Text query:

ELEVENTY, piquet, solid color, polo collar, long sleeves, no appliqués, no pockets, small sized. 100% Cotton.

Closest images:

Fig. 1. Example of a text and nearest images from the test set. Our embedding produces low distances between texts and images referring to similar objects.

colors, sizes, etc. Existing approaches for retrieval focus image-only and require hard to obtain datasets for training [2]. Instead, we opt to leverage easily obtained metadata for training our model, and learning a mapping from text and images to a common latent space, in which distances correspond to similarity.

Our approach consists of exploiting a Convolutional Neural network (CNN) for processing images, as well as word2vec-based embedding with a Neural Network for processing the textual information. Both networks are trained such that the distance between the output of related image-text pairs is minimized, while the distance between unrelated image-text pairs is maximized. Additionally, two auxiliary classification networks are used in combination with classification losses to retain semantic information in the common embedding.

We evaluate our approach in the retrieval task and our proposed approach outperforms KCCA [3] and Bag-of-word features on a large e-commerce dataset. We additionally provide an analysis of the different textual metadata.
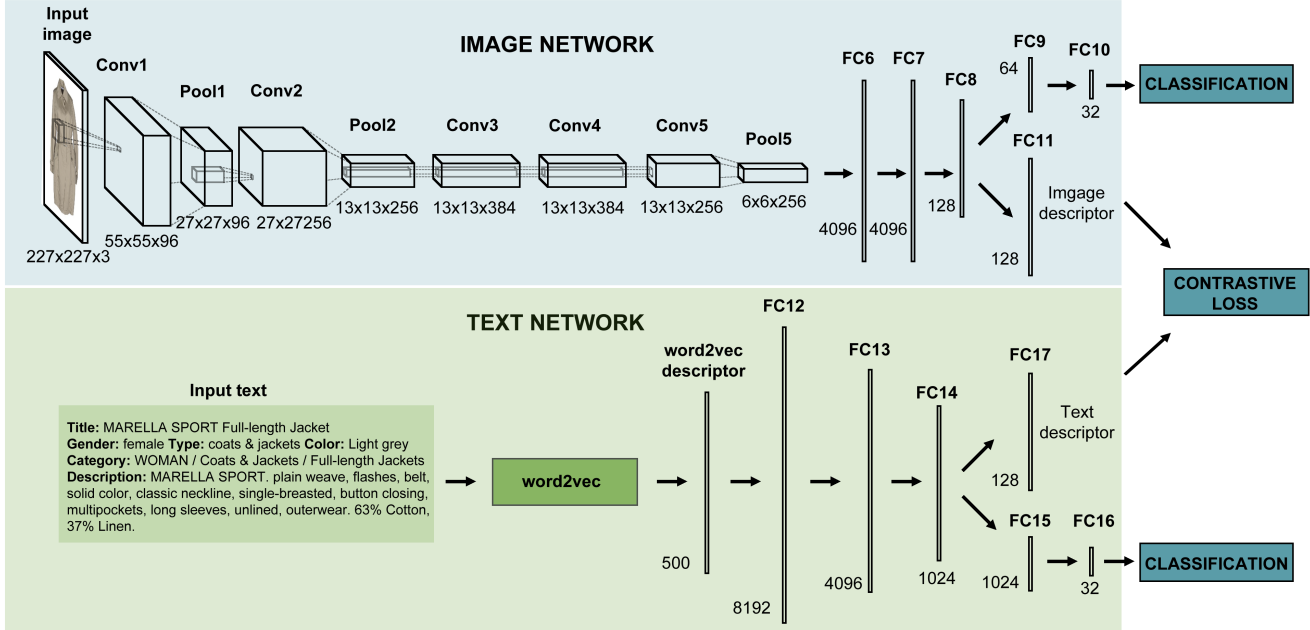
Fig. 2. Architecture of the neural network used. Conv, Pool and FC refer to convolutional, pooling and fully connected layers, respectively. When sizes of two dimensions are equal, some of them are omitted for clarity. Fully connected layers are uni-dimensional. Text descriptor and Image descriptor are the embedded vectors describing the input text and image, respectively.

2. RELATED WORK

Computer Vision for Fashion: Interest of computer vision researchers in Fashion has increased in the past years. Many works focus on clothing parsing, i.e., assigning a semantic label to each pixel of an image [4, 5, 6], others work at a higher level trying to infer deductions from the clothes, such as the person occupation [7], its social tribe [8] or its fashionability [9, 10]. Nevertheless, some of the more practical tasks might be clothing retrieval and classification [11, 12], which we tackle in this paper.

Retrieval task consists on finding similar items given a query. The usual pipeline for image retrieval is formed by three steps: extracting local image descriptors (such as Fisher Vectors [13, 14, 15]), reducing the dimensionality and indexing. For text retrieval, classical approaches looked for repetitions of the query words in a document, while newer latent semantic models [16, 17] use more powerful distributed text representations capable of learn the context of words and meaning of documents. There is recently a great effort focused on word embeddings and their applications [18, 19, 20, 21]. According to [22], current image retrieval techniques can be distributed into: text-based, content-based, composite and interactive approaches. Our method allows to retrieve texts or images with any kind of query via a common embedding for image and text.

The idea of combining models within different domains of a dataset has already been treated. [23, 24, 25, 26]. Most of the approaches train with one source domain and then regularize their classifiers to work with the target domain [27, 28]. In our case, we simultaneously train with data from both domains, producing a common space specifically learned for the retrieval task.

3. METHOD

Our joint multi-modal embedding approach consists of a neural network with two branches: one for image and one for text. The image branch is based on a Convolutional Neural Network (CNN) which converts a 227×227 pixel image into a fixed-size 128-dimensional vector. The text branch is based on a multi-layer neural network and uses as an input features extracted by a pre-trained word2vec network which are converted into a fixed-size 128-dimensional vector. Both branches are trained jointly such that the 128-dimensional output space becomes a joint embedding by minimizing the distance between related image-text pairs and maximizing the distance between unrelated image-text pairs. Two auxiliary classification networks are also used during training that encourages the joint embedding to also encode semantic concepts. An overview can be seen in Fig. 2.

3.1. Image Network

The image network branch is based on the AlexNet [29] architecture pre-trained on a fashion subset of ImageNet. The last layer is removed and replaced with a smaller Fully-Connected layer that has 128-dimensional outputs ($FC8$). This is further split into two branches: one for classification and one for the embedding. The classification branch has two fully connected layers ($FC9$ and $FC10$) and outputs the score of the different classes. The embedding branch has a single layer which outputs the 128-dimensional feature vector for the embedding ($FC11$). All fully connected layers $FC8 - FC11$ consist of the fully connected layer itself, followed by a batch normalization [30] layer and a Rectified Linear Unit (ReLU) layer.

3.2. Text Network

As preprocessing, we first delete numbers and punctuation marks, and then switch all characters to lower-case. Afterwards, we train from scratch a word2vec [17] model using our training set, with 500 dimensions using bigrams, with a context window of 3 words and ignoring words appearing less than 5 times in the dataset. The input for the text branch of the network are the descriptors computed averaging the word2vec distributed representations for all the words in each text [31].

The text network consists of 3 common fully-connected layers that output 1024-dimensional features ($FC12$-$FC14$). Afterwards the network splits into two branches: the classification branch and embedding branch. The classification branch consists once again of two additional layers ($FC15$ and $FC16$) and the output is the score of the different classes. The embedding branch outputs 128-dimensional vectors for the joint embedding. All fully connected layers in the text network are formed by the fully connected layer itself, followed by a batch normalization layer and a ReLU layer.

3.3. Training

For training we assume we have a large dataset of corresponding text-image pairs with class labels. The class labels are used for the classification losses and for randomly sampling negatives for training the embedding.

Training of both the text network and image network is done jointly by encouraging similar text-images pairs to have a small distance between the embedded vectors, while having dissimilar text-image pairs have a large distance. Images and their associated text are used as positive pairs, while unrelated image-text pairs are obtained by randomly sampling images and texts from unrelated

Description: MAURO GRIFONI. denim, solid color, mid rise, dark wash, front closure, button, zip, multipockets, logo, slim fit. 84% Cotton, 14% Elastomultiester, 2% Elastane.
Title: MAURO GRIFONI Denim Pants
Gender: female
Color: Blue
Type: denim
Category: WOMAN / Denim / Denim Pants

Fig. 3. Example of a product's image and text data.

categories. This is done by using the contrastive loss [32]:

$$L_C(v_I, v_T, y) = (1 - y)\frac{1}{2}\left(\|v_I - v_T\|_2\right)^2$$
$$+ (y)\frac{1}{2}\left\{\max\left(0, m - \|v_I - v_T\|_2\right)\right\}^2 \quad (1)$$

where v_I and v_T are two embedded vectors corresponding to the image and the text respectively, and y is a label that indicates whether or not the two vectors are compatible ($y = 0$) or dissimilar ($y = 1$), and m is a margin for the negatives.

The fully training loss consists of both the contrastive loss and the weighted sum of the cross entropy classification losses:

$$L_C(v_I, v_T, y) + \alpha L_X(C_I(v_I), L_I) + \beta L_X(C_T(v_T), L_T) \quad (2)$$

where L_X is the cross entropy loss, $C_I(v_i)$ is the output of the image classification network, L_I is the image label, $C_T(v_T)$ is the output of the text classification network, L_T is the text label, and α and β are two weighting hyperparameters..

4. RESULTS

Next, we describe the results obtained by applying our method to a Fashion e-commerce dataset, in which we train a common embedding where distances between text and images referring to products of the same category are considerably smaller than distances between those of different categories. We compare against existing approaches, analyze the different text features, and look at classification results with the auxiliary networks.

We train the network for 100,000 iterations with batches of 64 samples (forming in each iteration 64 correlated pairs image-text and 64 non-correlated pairs) with $\alpha = \beta = 1$. Training is done using stochastic gradient descent with backpropagation. We use an initial

learning rate of 10^{-3} and decrease it by $5 \cdot 10^{-4}$ every $10,000$ iterations with momentum 0.95.

4.1. Dataset

The dataset we use consists of 431,841 images of fashion products with associated texts, classified in 32 categories (vest, hats, boots, polo, jewelry, skirt, clutch/wallet, cardigan, shirt, dress, backpack, swimwear, suits, travel bags, glasses/sunglasses, pants/leggings, flats, shorts, coat/cape, tops, pump/wedge, sweatshirt/hoodie, sandals, crossbody/messenger bag, blazer, top handles, belts, jacket, other accesories, jumpsuits, sweater and joggers). The textual information for each product comes separated in different fields such as description, title, gender, type, color and category. See Fig. 3 for an example of a product in the dataset. We use 60% of the dataset for training, 30% for test and 10% for validation, and train the model using different combinations of textual information associated to the images to check the influence of the different types of text.

4.2. Retrieval

In order to evaluate our method, we compute all the 128-dimensional descriptors of images and texts in the testing set. Then, use the text as queries to obtain the images, and vice-versa. Looking at the position at which the exact match is, we compute the median rank for each case. The resultant values are below 2%, meaning that the exact match is usually closer than the 98% of the dataset, beating the result obtained by KCCA[1] and by our same architecture substituting the word2vec by a classical Bag of Words.

for an input image is in the top 2% of the results (see Table ?? for details). Then we compute the same thing for text descriptors as queries, and obtain similar values. Furthermore, we observe that the top results for a query are coherent with the category of the query, and that using more textual information in general increases the performance of the resulting embedding, specially using very meaningful fields such as category or title, where the product type and category is explicitly written. Table ?? shows exhaustive results of the models evaluated.There we can also see the recall at different levels. It is relevant to notice that for all the models, in most cases (around 80%) the exact match is in the top 5% of the nearest products.

We compare this metrics with two baselines: a version of our method replacing word2vec by Bag of Words and KCCA

4.3. Classification

In parallel to the ranking task, we are training a classification task. This one, intended to help clustering in a certain way the products of the same category in the common embedding, maintains high accuracy values ($> 95\%$ in some cases, as seen in Table ??) for the 32 clothing categories defined in the dataset.

5. CONCLUSIONS

In this paper, we have presented an approach for joint multi-modal embedding with neural networks with a focus on the fashion domain. Our approach is easily amenable to large existing e-commerce datasets by exploiting readily available images and their associated metadata. By training the embedding such that distances correspond to similarities, our approach can be easily used for retrieval tasks. Furthermore, our auxiliary classification networks help encourage the embedding to have semantic meaning, making it suitable as features for classification tasks.

6. ACKNOWLEDGMENTS

---

[1]The KCCA model has been trained with less descriptors (only 10000) due to memory errors when trying to use the whole training set

Table 1. Results of our method compared to KCCA and our method using Bag of Words for text representation.

Model	Median rank		Image		Text		Accuracy		Diff.
	Img v. txt	Txt v. img	f@5%	f@10%	f@5%	f@10%	Text	Image	
KCCA	52.42%	46.65%	3.70	7.59	3.90	9.59	-	-	-
Bag of Words	4.50%	4.54%	53.18	75.02	53.14	74.20	99.78%	71.73%	0.327576
word2vec	1.61%	1.63%	77.90	89.24	77.47	89.78	99.97%	90.06%	0.44

7. REFERENCES

[1] "The Ultimate List of E-Commerce Stats for Holiday 2016," http://blog.marketingadept.com/the-ultimate-list-of-e-commerce-marketing-stats-for-holiday-2016/, Accessed: 2017-01-23.

[2] M Hadi Kiapour, Xufeng Han, Svetlana Lazebnik, Alexander C Berg, and Tamara L Berg, "Where to buy it: Matching street clothing photos in online shops," in CVPR, 2015.

[3] Francis R Bach and Michael I Jordan, "Kernel independent component analysis," JMLR, vol. 3, no. Jul, pp. 1–48, 2002.

[4] Kota Yamaguchi, M Hadi Kiapour, and Tamara L Berg, "Paper doll parsing: Retrieving similar styles to parse clothing items," in CVPR, 2013.

[5] Jian Dong, Qiang Chen, Wei Xia, Zhongyang Huang, and Shuicheng Yan, "A deformable mixture parsing model with parselets," in CVPR, 2013.

[6] Si Liu, Jiashi Feng, Csaba Domokos, Hui Xu, Junshi Huang, Zhenzhen Hu, and Shuicheng Yan, "Fashion parsing with weak color-category labels," IEEE Transactions on Multimedia, vol. 16, no. 1, pp. 253–265, 2014.

[7] Zheng Song, Meng Wang, Xian-sheng Hua, and Shuicheng Yan, "Predicting occupation via human clothing and contexts," in CVPR, 2011.

[8] M Hadi Kiapour, Kota Yamaguchi, Alexander C Berg, and Tamara L Berg, "Hipster wars: Discovering elements of fashion styles," in ECCV, 2014.

[9] Kota Yamaguchi, Tamara L Berg, and Luis E Ortiz, "Chic or social: Visual popularity analysis in online fashion networks," in ACMMM, 2014.

[10] Edgar Simo-Serra, Sanja Fidler, Francesc Moreno-Noguer, and Raquel Urtasun, "Neuroaesthetics in fashion: Modeling the perception of fashionability," in CVPR, 2015.

[11] Wei Di, Catherine Wah, Anurag Bhardwaj, Robinson Piramuthu, and Neel Sundaresan, "Style finder: Fine-grained clothing style detection and retrieval," in CVPR Workshops, 2013.

[12] Lukas Bossard, Matthias Dantone, Christian Leistner, Christian Wengert, Till Quack, and Luc Van Gool, "Apparel classification with style," in ACCV, 2012.

[13] Florent Perronnin and Christopher Dance, "Fisher kernels on visual vocabularies for image categorization," in CVPR, 2007.

[14] Florent Perronnin, Yan Liu, Jorge Sánchez, and Hervé Poirier, "Large-scale image retrieval with compressed fisher vectors," in CVPR, 2010.

[15] Florent Perronnin, Jorge Sánchez, and Thomas Mensink, "Improving the fisher kernel for large-scale image classification," in ECCV, 2010.

[16] David M Blei, Andrew Y Ng, and Michael I Jordan, "Latent dirichlet allocation," JMLR, vol. 3, no. Jan, pp. 993–1022, 2003.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, "Distributed representations of words and phrases and their compositionality," in NIPS, 2013.

[18] Fernando Diaz, Bhaskar Mitra, and Nick Craswell, "Query expansion with locally-trained word embeddings," arXiv preprint arXiv:1605.07891, 2016.

[19] Debasis Ganguly, Dwaipayan Roy, Mandar Mitra, and Gareth JF Jones, "Word embedding based generalized language model for information retrieval," in SIGIR, 2015.

[20] Mihajlo Grbovic, Nemanja Djuric, Vladan Radosavljevic, and Narayan Bhamidipati, "Search retargeting using directed query embeddings," in WWW, 2015.

[21] Bhaskar Mitra, Eric Nalisnick, Nick Craswell, and Rich Caruana, "A dual embedding space model for document ranking," arXiv preprint arXiv:1602.01137, 2016.

[22] B Dinakaran, J Annapurna, and Ch Aswani Kumar, "Interactive image retrieval using text and image content," Cybern Inf Tech, vol. 10, pp. 20–30, 2010.

[23] Sean Bell and Kavita Bala, "Learning visual similarity for product design with convolutional neural networks," ACM Transactions on Graphics (SIGGRAPH), vol. 34, no. 4, pp. 98, 2015.

[24] Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in CVPR, 2013.

[25] Raghuraman Gopalan, Ruonan Li, and Rama Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in CVPR, 2011.

[26] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in CVPR, 2012.

[27] Alessandro Bergamo and Lorenzo Torresani, "Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach," in NIPS, 2010.

[28] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell, "Adapting visual category models to new domains," in ECCV, 2010.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in NIPS, 2012.

[30] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in ICML, 2015.

[31] Vedran Vukotić, Christian Raymond, and Guillaume Gravier, "Multimodal and crossmodal representation learning from textual and visual features with bidirectional deep neural networks for video hyperlinking," in Proceedings of the ACM workshop on Vision and Language Integration Meets Multimedia Fusion, 2016.

[32] Raia Hadsell, Sumit Chopra, and Yann LeCun, "Dimensionality reduction by learning an invariant mapping," in CVPR, 2006.