FASHION SMART RECOMMENDATION

An B. Ha
Advanced Program in Computer Science, Falcuty of Information Technology
University of Science, VNU-HCM
hban@apcs.vn

Bao G. Dinh
Advanced Program in Computer Science, Falcuty of Information Technology
University of Science, VNU-HCM
dhgbao@apcs.vn

Hieu M. Nguyen
Advanced Program in Computer Science, Falcuty of Information Technology
University of Science, VNU-HCM
nmhieu@apcs.vn

Huy N. Doan
Advanced Program in Computer Science, Falcuty of Information Technology
University of Science, VNU-HCM
dnhuy17@apcs.vn

ACM Reference format:
An B. Ha, Bao G. Dinh, Hieu M. Nguyen, and Huy N. Doan. 2019. FASHION SMART RECOMMENDATION. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 7 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

ABSTRACT

Nowadays, online shopping is becoming more and more popular. You just need to sit at home, pick something you want to buy, purchase and it will be delivered after a few days. And online fashion shopping needs an effective recommendation services for customers. The services should be able to (i) suggesting an item that matches a set of clothes, and (ii) can pick clothes that fit the requirement of customers. To do this we follow the instruction from ACM MM'17 paper "Learning Fashion Compatibility with Bidirectional LSTMs". The paper proposed to learn a visual-semantic embedding and the compatibility relationships among fashion items. Following the instructions from the paper, with given fashion items, we train a bidirectional LSTM(Bi-LSTM) model. With the trained model, we will be able to predict the next item that fits on previous ones. By doing this, we can trained the model to learn their compatibility relationships.Further, we learn a visual-semantic space by regressing image features to their semantic representations aiming to inject attribute and category information as a regularization for training the LSTM. The trained network can also give grade to a given outfit, so that it can compared the compatibility to others. We use the updated Polyvore dataset, and the results show better performance than alternative methods.

1) INTRODUCTION

Fashion is one of important taste in our life. It can be able to display personality and shaping culture. Nowadays, shopping on the internet becoming popular tendency and increasing continuously. So, making online shopping is as convenient possible is motivation for creating a techniques that can recommend a set of fashion items effectively in two purposes (1) recommend an items that can mix with an existing set and (2) create an outfit (set of items) by giving text/image. However, the main challenging things we have to solve is modeling and conclude the compatibility of massive amount of different fashion categories. There are a lot of deep studies relating to fashion which have been created on automatic fashion analysis in the various kind of community. But, most of them concentrate on clothing parsing [9, 26], clothing recognition[12], or clothing retrieval [10]. Even though, there are some investigation in fashion recommendation [6, 8, 10], they cannot combine different items to form an outfit [10] or just can only achieve one of two forms discussed above [6,8]. We hope the recommendation can take multiple input from users and give the most compatible result. For examples, when user give a keywords like "meeting", or any image of a formal shirt, or both of them, to generate a set of fashion items for a meeting occasion.

Giving keywords to recommendation is computing compatibility of fashion items. We think that a compatible outfit should have two key properties: (1) items in one outfit should be compatible by eyes and have similar style; (2) these items must be sufficient without redundancy (e.g., an outfit with only a T-shirt and a trouser is not compatible, neither an outfit has 2 pairs of shoes). A possible solution can be given is using semantic attributes, e.g, "black jeans" can match with "running shoes". But let the machine can understand these attributes is very difficult. To decrease this issue, researchers advise learning the distance between every two of fashion items by utilizing metric learning [15] or a Siamese network [24]. However, these work just calculate approximate compatibility of pair of items than whole outfit. One can calculate the compatibility of an outfit with some voting plan using all items in the set, but it could cause a high computational cost when the collection is big and could not create coherence of all items in set. Besides, some works [8,21] effort to calculate the popularity or "fashionability" of an outfit, unfortunately they cannot generate the outfit.

To avoid the above limitations, authors firstly use Inception-V3 CNN model[22] to transform an image to a feature vector. Then using a layer bidirectional LSTM (Bi-LSTM)[3] which is considered a set of fashion items as a sequence with a specific arrange - top to bottom and then to accessories, each image in set as a time step.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

At each time step, with the available image, authors train the Bi-LSTM model to predict the next items in outfit. Moreover, beside generate the next compatible items, we also learn a visual-semantic embedding by transform the image features into a language representation of their comments. This not only provide semantic attribute and category information of current input for training LSTM, but also solve the problem with multiple inputs from users. Finally,the model is trained end-to-end to jointly learn the compatibility relationships as well as the visual-semantic embedding. After training model, there are three task to evaluate: (1) Fill-in-the-blank: result an sufficient outfit with one missing item, suggest an item that is suitable with available items; (2) Outfit generation: Creating an set of items with multiple input from users; (3) Compatibility prediction: Predict the compatibility of a given outfit. We perform this experiments on Polyvore dataset.

2) RELATED WORKS

Fashion has a high impact on our everyday lives. This also shows in the growing interest in clothing-related applications in the vision community.

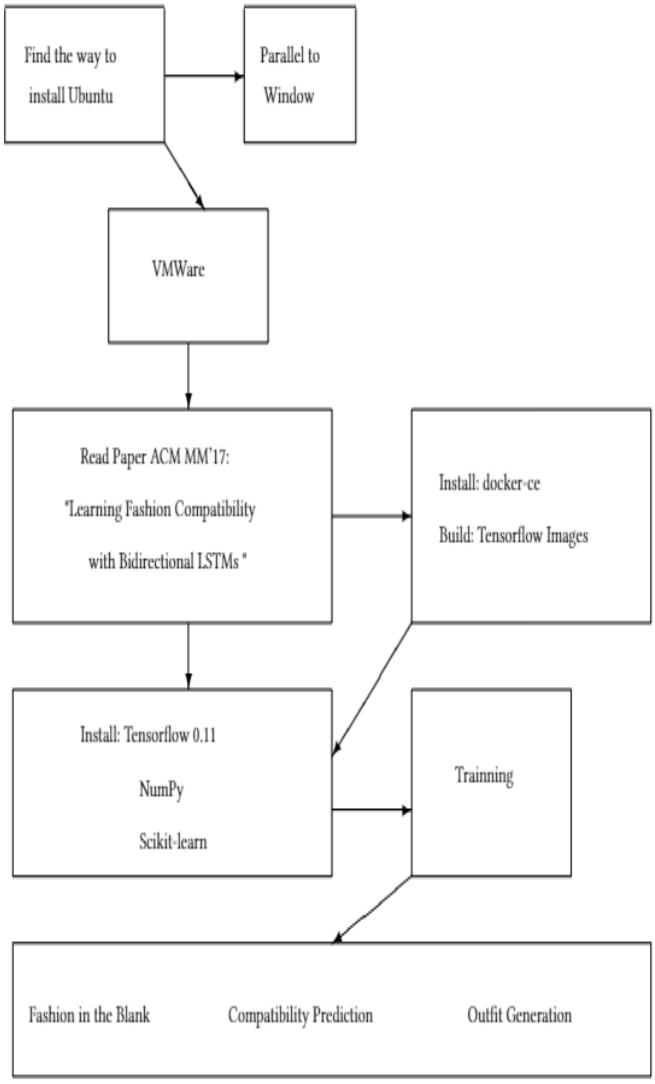
Fashion Recognition and Retrieval. There is a growing interest in identifying fashion items in images due to the huge potential for commercial applications. Most recent works utilize standard segmentation methods, in combination with human pose information, to parse different garment types [25, 27] for effective retrieval. Liuet al. proposed a street-to-shop application that learns a mapping between photos taken by users with product images [11]. Hadi etal. further utilized deep learning techniques to learn the similarity between street and shop images [5]. Recently, Liu et al. introduced FashionNet to learn fashion representations that jointly predicts clothing attributes and landmarks [12]. In contrast to these works focusing on retrieval tasks, our goal is to learn the visual compatibility relationships of fashion items in an outfit.

Fashion Recommendation. As discussed previously, there are a few approaches for recommending fashion items. Liu et al. introduced an occasion-based fashion recommendation system with a latent SVM framework that relies on manually labeled attributes [10]. Hu et al. proposed a functional tensor factorization approach to generate an outfit by modeling the interactions between user and fashion items. Recently, Li et al. trained an RNN to predict the popularity of a fashion set by fusing text and image features . Then they constructed a recommendation by selecting the item that produces the highest popularity score when inserted into a given set. However, the results were no better than random. In contrast to these approaches, our method learns the compatibility relationships among fashion items together with a visual-semantic embedding, which enables both item and outfit recommendation.

More related to our line of work are recent applications such as learning semantic clothing attributes , identifying people based on their outfits, predicting occupation and urban tribes , outfit similarity , outfit recommendations , and predicting outfit styles . Most of these approaches address very specific problems with fully annotated data. In contrast, the model we propose is more general, allowing to reason about several properties of one's photo: the aesthetics of clothing, the scenery, the type of clothing the person is wearing, and the overall fashionability of the photograph. We do not require any annotated data, as all necessary information is extracted by automatically mining a social website.

Our work is also related to the recent approaches that aim at modeling the human perception of beauty. In the authors addressed the question of what makes an image memorable, interesting or popular. This line of work mines large image datasets in order to correlate visual cues to popularity scores (defined as e.g., the number of times a Flickr image is viewed), or "interestingness" scores acquired from physiological studies. In our work, we tackle the problem of predicting fashionability. We also go a step further from previous work by also identifying the highlevel semantic properties that cause a particular aesthetics score, which can then be communicated back to the user to improve her/his look. The closest to our work is which is able to infer whether a face is memorable or not, and modify it such that it becomes. The approach is however very different from ours, both in the domain and in formulation. Parallel to our work, Yamaguchi et al. investigated the effect of social networks on votes in fashion websites.

3) METHOD


Find the way to install Ubuntu
Parallel to Window
VMWare
Read Paper ACM MM'17: "Learning Fashion Compatibility with Bidirectional LSTMs "
Install: docker-ce
Build: Tensorflow Images
Install: Tensorflow 0.11
NumPy
Scikit-learn
Trainning
Fashion in the Blank
Compatibility Prediction
Outfit Generation

We have performed a detailed quantitative evaluation on the fill-in-the-blank fashion prediction task. The same effort is also made for other higher level task suck as outfit generation and fashion compatibility prediction.

Implementation Details:

Bidirectional LSTM. We use 2048D CNN features derived from the GoogleNet InceptionV3 model [22] as the image representation, and transform the features into 512D with one fully connected layer before feeding them into the Bi-LSTM. The number of hidden units of the LSTM is 512, and we set the dropout rate to 0.7.

Visual-semantic Embedding. The dimension of the joint embedding space is set to 512, and thus WI R2048×512 and WT R2757×512, where 2757 is the size of the vocabulary. We fix the marginm = 0.2 in Eqn. 5.

Joint Training. The initial learning rate is 0.2 and is decayed by a factor of 2 every 2 epochs. The batch size is set to 10, and thus each mini batch contains 10 fashion outfit sequences, around 65 images and their corresponding descriptions. Finally, we fine-tune all layers of the network pre-trained on ImageNet. We stop the training process when the loss on the validation set stabilizes.

3.1 Learning Fashion Compatibility with Bi-LSTMs

Bidirectional LSTMs are used as they are an extension of traditional LSTMs that can improve model performance on sequence classification problems. Bidirectional LSTMs train two instead of one LSTMs on the input sequence, [] enables them to learn relationships between two time steps, and the use of memory units regulated by different cells facilitates exploiting long-term temporal dependencies. In our problem, in order to take advantage of the representation power of LSTM, we treat an outfit as a sequence and each image in the outfit as an individual time step, and employ the LSTM to model the visual compatibility relationships of outfits. Given a fashion image sequence F = x1, x2, ..., xN , xt is the feature representation derived from a CNN model for the t-th fashion item in the outfit. At each time step, we first use a forward LSTM to predict the next image given previous images; learning the transitions between time steps serves as a proxy for estimating the compatibility relationships among fashion items. More formally, we minimize the following objective function:

Ef (F;f) = 1 N ÕN t=1 loPr (xt+1 |x1, ..., xt ;f), (1)

where f denotes the model parameters of the forward prediction model and Pr (), computed by the LSTM model, is the probability of seeing xt+1 conditioned on previous inputs.

More specifically, the LSTM model maps an input sequence x1, x2, . . . , xN to outputs via a sequence of hidden states by computing the following equations recursively from t = 1 to t = N:

it = (Wxi xt +Whiht1 +Wci ct1 + bi),
ft = (Wxf xt +Whf ht1 +Wcf ct1 + bf),
ct = ft ct1 + it tanh(Wxc xt +Whcht1 + bc),
ot = (Wxoxt +Whoht1 +Wcoct + bo),
ht = ot tanh(ct),

where xt , ht are the input and hidden vectors of the t-th time step, it , ft , ct , ot are the activation vectors of the input gate, forget gate, memory cell and output gate, W is the weight matrix between vector and (e.g., Wxi is weight matrix from the input xt to the input gate it), b is the bias term of and is the sigmoid function.

Following [16] that utilizes softmax output to predict the next word in a sentence, we append a softmax layer on top of ht to calculate the probability of the next fashion item conditioned on previously seen items:

Pr (xt+1 |x1, ..., xt ;f) = Íexp(ht xt+1) xX exp(ht x) , (2)

where X contains all images (in multiple outfits) from the current batch. This allows the model to learn discriminative style and compatibility information by looking at a diverse set of samples. Note that one can choose X to be the whole vocabulary [17] as in sentence generation tasks; however this is not practical during training our model due to the large number of images and high-dimensional image representations. Therefore, we set X to be all possible choices in the batch of xt+1 to speed up training, instead of choosing from hundreds of thousands of images from the training data.

Given a fashion item, it makes intuitive sense that predicting the next item can be performed in the reverse order also. For example, the next item for "pants" could be either "shirts" or "shoes". Therefore, we also build a backward LSTM to predict a previous item given the items after it:

Eb (F;b) = 1 N Õ0 t=N1 loPr (xt |xN , ..., xt+1;b), (3) and Pr (xt |xN , ..., xt+1;b) = exp(~h Í t+1xt) xX exp(~ht+1x) , (4)

where ~ht+1 is the hidden state at time t + 1 of the backward LSTM, and b denotes the backward prediction model parameters. Note that we add two zero vectors x0 and xN+1 in F so that the bidirectional LSTM learns when to stop predicting the next item. Since an outfit is usually a stylish ensemble of fashion items that share similar styles (e.g., color or texture), by treating an outfit as an ordered sequence, the Bi-LSTM model is trained explicitly to capture compatibility relationships as well as the overall style of the entire outfit (knowledge learned in the memory cell). This makes it a very good fit for fashion recommendation.

3.2 Visual-semantic Embedding

Fashion recommendation should naturally be based on multimodal inputs (exemplar images and text describing certain attributes) from users. Therefore, it is important to learn a multimodal embedding space of texts and images. Instead of annotating images with labels or attributes, which is costly, we leverage the weakly-labeled web data, i.e., the informative text description of each image provided by the dataset, to capture multimodal information. To this end, we train a visual-semantic embedding by projecting images and their associated text into a joint space, which is widely used when modeling image-text pairs [7]. Given a fashion image from an outfit, its description is denoted as S = w1,w2, ...,wM where wi represents each word in the description.We first represent the i-th wordwi with one-hot vector ei , and transform it into the embedding space by vi = WT ei whereWT represents the word embedding matrix. We then encode the description with bag-of-words v = 1M Í i vi . LettingWI denote the image embedding matrix, we project the image representation x into the embedding space and represent it as f = WI x.

3.3 Joint Modeling

Given a fashion output, the Bi-LSTM is trained to predict the next or previous item by utilizing the visual compatibility relationships. However, this is not optimal since it overlooks the semantic information and also prevents users from using multimodal input to generate outfits. Therefore, we propose to jointly learn fashion compatibility and the visual-semantic embedding with an aim to

incorporate semantic information in the training process of the Bi-LSTM. The framework can be easily trained by Back-Propagation through time (BPTT) [3] in an end-to-end fashion, in which gradients are aggregated through time. The only difference compared to a standard Bi-LSTM model during backpropagation is that the gradients of the CNN model now stem from the average of two sources (See Figure 2), allowing the CNN model to learn useful semantic information at the same time. The visual semantic embedding not only serves as a regularization for the training of Bi-LSTM but also enables multimodal fashion recommendation as will be demonstrated in the next section.

3.4 Compared other approaches

To demonstrate the effectiveness of our approach for modeling the compatibility of fashion outfits, we compare with the following alternative methods: SiameseNet [24]. SiameseNet utilizes a Siamese CNN to project two clothing items into a latent space to estimate their compatibility.

To compare with SiameseNet, we train a network with the same structure by considering fashion items in the same outfit as positive compatible pairs and items from two different outfits as negative pairs. The compatibility of an outfit is obtained by averaging pairwise compatibility, in the form of cosine distance in the learned embedding, of all pairs in the collection. For fair comparisons, the embedding size is also set to 512.We also normalize the embedding with 2 norm before calculating the Siamese loss, and set the margin parameter to 0.8.

SetRNN [8]. Given a sequence of fashion images, SetRNN predicts the fashion set popularity using an RNN model. We use the popularity prediction of SetRNN as the set compatibility score.

Visual-semantic Embedding (VSE). We only learn a VSE by minimizing Ee in Eqn. 5 without training any LSTM model. The resulting embeddings are used to measure the compatibility of an outfit, similar to SiameseNet.

Bi-LSTM. Only a bidirectional LSTM is trained without incorporating any semantic information.

F-LSTM+VSE. Jointly training the forward LSTM with visualsemantic embedding, i.e., minimizing Ef + Ee .

B-LSTM+VSE. Similarly, only a backward LSTM is trained with visual-semantic embedding, i.e. minimizing Eb + Ee .

Bi-LSTM+VSE. Our full model by jointly learning the bidirectional LSTM and the visual-semantic embedding.

The first two approaches are recent works in this line of research and the remaining methods are used for ablation studies to analyze the contribution of each component in our proposed framework. The hyper-parameters in these methods are chosen using the validation set.

4) EXPERIMENTAL RESULTS

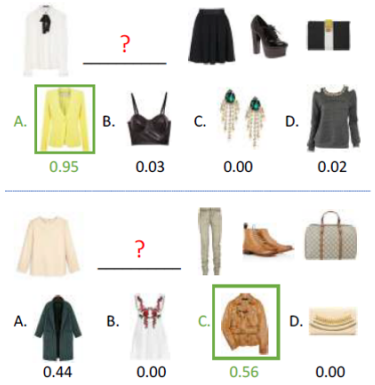
5.1 Fill-in-the-blank Fashion Recommendation

We inherit from the CM MM'17 paper "Learning Fashion Compatibility with Bidirectional LSTMs" paper the Fill-in-the-blank task. With a given sequence of fashion items, an item need to be chosen from multiple choices that is compatible with other items to fill in the blank. In real life, users may want to get suggestions to choose a piece of clothes to fit with the clothes he/she wearing.

From the Polyvore set, for each outfit, one item is randomly selected and is replaced with a blank, and 3 selected items from other outfits to make a multiple choice set. Once the Bi-LSTM-VSE is trained, we solve the fill-in-the-blank task.

Method	FITB accuracy	Compatibility AUC
SetRNN [8]	29.6%	0.53
SiameseNet [24]	52.0%	0.85
VSE	29.2%	0.56
F-LSTM + VSE	63.7%	0.89
B-LSTM + VSE	61.2%	0.88
Bi-LSTM	66.7%	0.89
Bi-RNN + VSE	63.7%	0.85
Bi-GRU + VSE	67.1%	0.89
Bi-LSTM + VSE (Ours)	**68.6%**	**0.90**

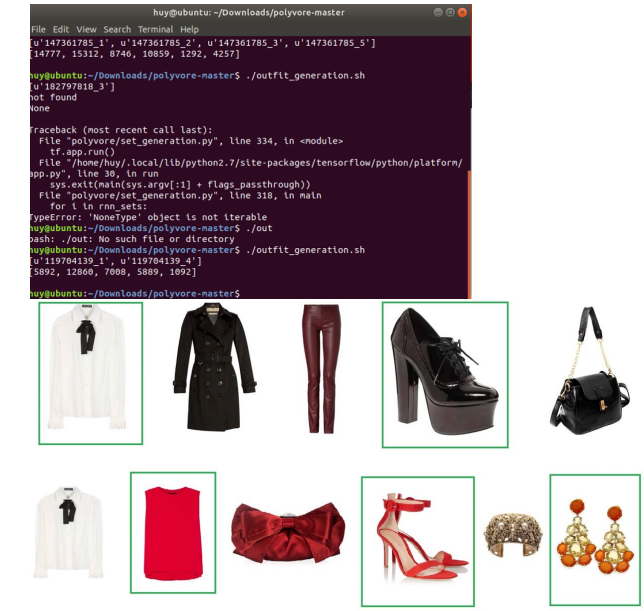
From the table, we can see that the Bi-LSTM with the visual-semantic embedding, and the resulting full model achieves the best performance with an accuracy of 68,6%, 1,9 percentage points higher than Bi-LSTM. The paper also investigate different RNN architectures by replacing LSTM cells with gated recurrent unit (GRU) and basic RNN cells. GRU and LSTm are better than basic RNN by better addressing the "vanishing gradients" effect and better modeling the temporal dependencies. The fill-in-the-blank experiments show that LSTM is more suitable for modeling compatibility of fashion items.

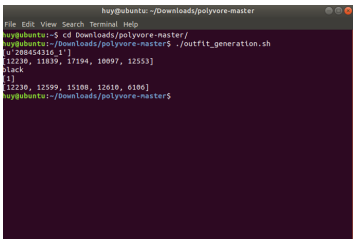


From the picture above, the paper visualize sample result of the filling-in-the-blank task. Combining Bi-LSTM and visual semantic embedding can not only detect what kinds of fashion item is missing, but also selects the fashion item that is most compatible to the query items and matches their style as well.

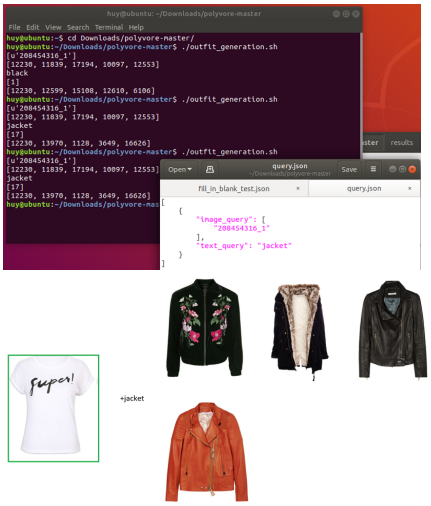
5.2 Fashion Outfit Generation

We now can see some test on how to utilize the paper proposed framework to generate an outfit from multimodal specifications such as images or text from users. **Generate Outfits from Query Images**.





By adding a specific color "Black", to the "text_query", the Fashion-Outfit-Generation test able to choose outfit base on the color we want. Not only color, style of clothes such as jacket,... can also be chosen from the dataset as the below pictures, we can see the result.



By learning a visual-semantic embedding together with the Bi-LSTM, the model can generate outfit from text query. This can be done by first generating an initial outfit using Bi-LSTM base on the given fashion items which is the dataset from Polyvore. We conducted experiments on different types of fashion recommendation task using the newly collected Polyvore dataset, and the result show that the proposed method can effectively learn the compatibility of fashion outfits.

5. CONCLUSION

We propose to jointly train a Bi-LSTM model and a VSE (visual-semantic embedding) for fashion compatibility learning. By considering the outfits as a sequence with each item in it as an time step, we utilize a Bi-LSTM model to predict the next item conditioned on previously seen ones, while training a VSE to provide category and attribute information in the training process of the Bi-LSTM. We conducted experiments on different types of fashion recommendation tasks by using Polyvore dataset that we have collected, and the results demonstrate that our method can learn the compatibility of fashion outfits effectively. Since fashion compatibility might vary from one person to another(height ,weight, personallity,...), modeling user-specific compatibility and style preferences is one of our future research directions.

REFERENCES

[1] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014).

[2] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. 2015. Long-term Recurrent Convolutional Networks for Visual Recognition and Description. In CVPR.

[3] Alex Graves. 2012. Supervised sequence labelling with recurrent neural networks.

Springer.

[4] Alex Graves, Abdel-rahman Mohamed, and Geoffrey E. Hinton. 2013. Speech recognition with deep recurrent neural networks. In ICASSP.

[5] MHadi Kiapour, Xufeng Han, Svetlana Lazebnik, Alexander C Berg, and Tamara L Berg. 2015. Where to buy it: Matching street clothing photos in online shops. In ICCV.

[6] Yang Hu, Xi Yi, and Larry S Davis. 2015. Collaborative fashion recommendation:

a functional tensor factorization approach. In ACM Multimedia.

[7] Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2015. Unifying visualsemantic

embeddings with multimodal neural language models. TACL (2015).

[8] Yuncheng Li, LiangLiang Cao, Jiang Zhu, and Jiebo Luo. 2016. Mining Fashion Outfit Composition Using An End-to-End Deep Learning Approach on Set Data.

arXiv preprint arXiv:1608.03016 (2016).

[9] Xiaodan Liang, Liang Lin, Wei Yang, Ping Luo, Junshi Huang, and Shuicheng Yan. 2016. Clothes co-parsing via joint image segmentation and labeling with application to clothing retrieval. IEEE TMM (2016).

[10] Si Liu, Jiashi Feng, Zheng Song, Tianzhu Zhang, Hanqing Lu, Changsheng Xu, and Shuicheng Yan. 2012. Hi, magic closet, tell me what to wear!. In ACM Multimedia.

[11] Si Liu, Zheng Song, Guangcan Liu, Changsheng Xu, Hanqing Lu, and Shuicheng Yan. 2012. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In CVPR. 3330–3337.

[12] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. 2016. Deepfashion:

Powering robust clothes recognition and retrieval with rich annotations. In CVPR.

[13] Tegan Maharaj, Nicolas Ballas, Aaron Courville, and Christopher Pal. 2016. A

dataset and exploration of models for understanding video data through fill-inthe-

blank question-answering. arXiv preprint arXiv:1611.07810 (2016).

[14] Amir Mazaheri, Dong Zhang, and Mubarak Shah. 2016. Video Fill in the Blank with Merging LSTMs. arXiv preprint arXiv:1610.04062 (2016).

[15] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In ACM SIGIR.

[16] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernock'y, and Sanjeev Khudanpur.

2010. Recurrent neural network based language model.. In Interspeech. [17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013.

Distributed representations of words and phrases and their compositionality. In NIPS.

[18] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. 2015. Beyond Short Snippets: Deep Networks for Video Classification. In CVPR.

[19] Jose Oramas and Tinne Tuytelaars. 2016. Modeling Visual Compatibility through

Hierarchical Mid-level Elements. arXiv preprint arXiv:1604.00036 (2016).

[20] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In CVPR.

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

[22] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew

Wojna. 2015. Rethinking the Inception Architecture for Computer Vision. arXiv preprint arXiv:1512.00567 (2015).

[23] Kristen Vaccaro, Sunaya Shivakumar, Ziqiao Ding, Karrie Karahalios, and Ranjitha

Kumar. 2016. The Elements of Fashion Style. In Proceedings of the 29th Annual Symposium on User Interface Software and Technology. ACM, 777–785. [24] Andreas Veit, Balazs Kovacs, Sean Bell, Julian McAuley, Kavita Bala, and Serge

Belongie. 2015. Learning visual clothing style with heterogeneous dyadic cooccurrences.

In CVPR.

[25] Kota Yamaguchi, M Hadi Kiapour, and Tamara L Berg. 2013. Paper doll parsing:

Retrieving similar styles to parse clothing items. In ICCV.

[26] Kota Yamaguchi, M Hadi Kiapour, Luis E Ortiz, and Tamara L Berg. 2015. Retrieving

similar styles to parse clothing. IEEE TPAMI (2015).

[27] Wei Yang, Ping Luo, and Liang Lin. 2014. Clothing co-parsing by joint image segmentation and labeling. In CVPR.

[28] Ting Yao, Tao Mei, and Chong-Wah Ngo. 2015. Learning Query and Image Similarities with Ranking Canonical Correlation Analysis. In ICCV.

[29] Licheng Yu, Eunbyung Park, Alexander C Berg, and Tamara L Berg. 2015. Visual

madlibs: Fill in the blank description generation and question answering. In Proceedings of the IEEE International Conference on Computer Vision. 2461–2469. [30] Linchao Zhu, Zhongwen Xu, Yi Yang, and Alexander G Hauptmann. 2015. Uncovering

temporal context for video question and answering. arXiv preprint arXiv:1511.04670 (2015).