Multi-task Fashion Style learning

Xin Wang
Textile Institute
Donghua Unversity
Shanghai, China
wangxin19930411@163.com

Yueqi Zhong*
Textile Institute
Donghua Unversity
Shanghai, China
Key Lab of Textile Science and Technology
Ministry of Education, China
zhyq@dhu.edu.cn

Abstract—In this paper, our solution for JD fashion challenge is presented, which includes several particular optimization methods for this task that is threshold optimization, statistical weighted binary cross-entropy(WBCE), stochastic weight average(SWA), voting model ensemble. Several out-of-box models are evaluated for this task. The result tells us that DenseNet121 is the best single model with 0.6239 F2 score. SWA strategy can obviously improve each model performance by increasing their generalization ability. After voting 5 models, the final F2 arrives 0.6498.

Index Terms—CNN, fashion, style, multitask

I. Introduction

The goal of multi-task fashion style recognition is to predict multiple style (e.g. sports, natural Punk etc.) based on one image. Benefits of this technology includes alleviating cold-starting problem in fashionrecommendation [1], achieving a better recommendation performance [2], supporting more advanced research like fashion compatiblity problem [3], fashion trend analysis and prediction [4].

JD AI Fashion-Challenge fashion style recognition task constructed a dataset including 54908 images with 13 style style attributes for each images to satisify multi-task fashion style recognition problem. In this paper, our training strategy for this competition will be introduced. Weighted binary cross entropy(WBCE), threshold optimization, stochastic weight average(SWA) and voting ensemble are four elements contribute to model performance. This strategy achieved 0.6498 F2 score and 4th place in final leaderboard.

The structure of this paper is as follows: first section is the exploratory of this dataset; then WBCE and threshold optimization will be discussed respectively; third, the performance of single models will be compared; finally two ensemble technologies SWA and voting will be introduced which help us to achieve the final score.

II. Dataset

Style recognition dataset is composed of 54908 images with 13 style attributes for each images. The examples of these fashion images can be found in Fig. 1. All of these images are female clothes. These clothes can be upper clothes, bottom clothes or both. It can be found that there are different photo contexts in these images. The photo could be token in a white background or street, also, there could be cooccurrence of person and clothes or only clothes. The clothes parts are most at the center of these images.

Fig. 1. Examples in dataset

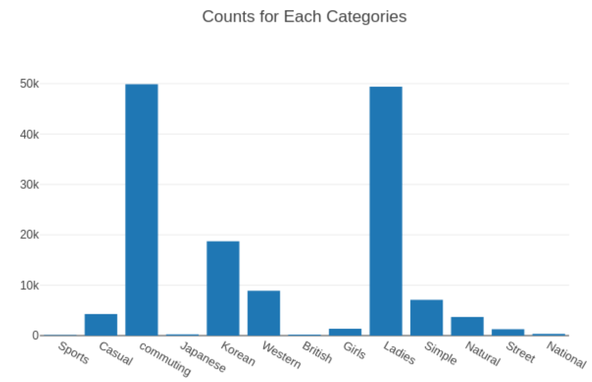

Fig. 2. Counts for each category shows the imbalance in the dataset.

Label imbalance is a frequent problem in machine learning, and it also appears in this dataset. Fig. 2 shows the results of counting 13 labels, it can be found that there are severe imbalance between these labels. The number of commuting labels is about 500 times of the number of sports. It is crucial to adopt corresponding tricks to overcome this problem and will be discussed in Section III-C. As preparation for training, the dataset was split into train and validation set with a ratio of 8:2. Images are normalized before fed into model and only random horizontal flip are used to augment the dataset. We think margin part of images and color are discriminative to recognize fashion style in images, so no random crop and color jittering are used.

III. Training

A. Architecture

Fig. 3 shows the overall architecture used in this task. Totally 5 single models(DenseNet121, DenseNet161 [5], InceptionV4 [6], ResNeXt101_32x4d, ResNeXt101_64x4d [7]) are incorporated to vote for the final decision. We think this task as a multi-task problem and make each model outputs a vector of 13 dimensions after sigmoid activation. After experimenting different loss function, weighted binary cross entropy(WBCE) is proved to be the best choice for this problem. To transform these output vectors into binary format i.e. 0 or 1, a threshold is needed. We observed that different style could have different distribution in their results, it means they need different thresholds. To solve that, we iterate each value in validation set outputs to find the optimized threshold, which is proved can benefit the model performance a lot. Finally, SWA [8](a kind of run time ensemble) and voting are used to ensemble models to achieve the final score.

During the training, we first freeze all layers except the last fully connection layer to train 7 epochs, then free all parameters to train another 24 epochs. Mean F2 score of fed validation batches is used to supervised the training to save the best model. Pytorch framework is used to implement models. Training one model on single PC equipped with i5-6500@3.2GHz CPU, 8GB memory and GTX1080Ti GPU takes about 6 hours.

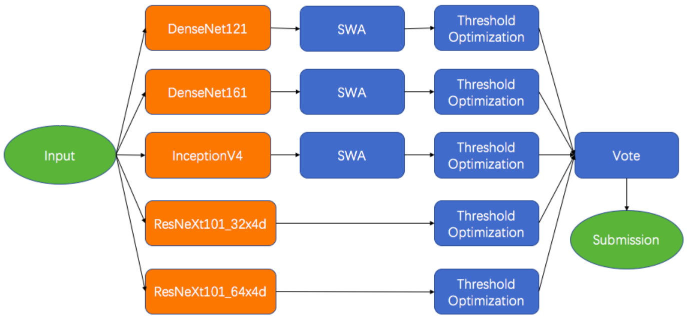

Fig. 3. Model architecutre.

B. Threshold optimization

The motivation of threshold optimization is the observation that the outputs of each style have different distributions, simply use 0.5 as the threshold does not match the facts. Our optimization method is let S be the set of all values of outputs, then the optimized threshold is:

$$t^* = \text{argmax}_{t \in S} F2(S, t) \tag{1}$$

Note that the outputs are from validation set, so the optimization performance should be evaluated on test set. Each style will have different thresholds. A baseline resnet18 [9] model with 0.5 threshold and optimized threshold have been compared. The former achieve 0.40 F2 score on test set and the later achieve 0.55 F2 score.

C. Overcome imbalance

Different methods have been tried to overcome the imbalance in this dataset includes WBCE, oversample, Soft F2 loss, Focal loss [10]. Specifically, WBCE multiply different weight on each output dimension as definition in (2). The weight of each style dimension is based on its count in dataset, less count will have larger weight, formula (3) is used to generated weights based on statistics:

$$\mathcal{L} = w(-y\log(p) + (1-y)\log(p)) \tag{2}$$

$$w = \alpha^{-\log(n)} \tag{3}$$

where α is set to 1.5, n is the count of label. Oversample mean repeatedly feed minority samples into model to fit them better. According to Fig. 2, Sports, Japanese, British, National are chosen to be oversampled. Different oversample levels(10×, 20×, 100×) have been experimented. Focal loss will give more punishment on hard example, and is defined as:

$$\mathcal{FL}(p_t) = (1-p_t)^{\gamma}\log(p_t) \tag{4}$$

$$p_t = \begin{cases} p & \text{y=1} \\ 1-p & \text{otherwise} \end{cases} \tag{5}$$

TABLE I shows the their performance based on resnet18 model. Obviously, WBCE without any oversample achieve the best performance 0.6000 F2 score.

Finally, we evaluate several typical single model with threshold optimization and WBCE. The results are shown in TABLE II. Different models have their own advantages, and DenseNet121 has the best mean F2 score 0.6169.

TABLE I
Different methods to overcome imbalance

Method	Checked							
BCE	√	√	√	√			√	
WBCE					√	√		
F2 loss							√	
Focal loss								√
Oversample@10		√						
Oversample@20			√					
Oversample@100				√		√		
Performance	0.5500	0.5405	0.5317	0.5691	0.6000	0.5902	0.5691	0.4251

TABLE II
Performance of different single model with WBCE and threshold optimization.

Model	Sp[a]	Ca	Co	Ja	Ko	We	Br	Gi	La	Si	Nu	St	Ni	MF2
ResNet18	0.259	0.627	0.981	0.439	0.806	0.712	0.368	0.444	0.978	0.608	0.716	0.483	0.380	0.600
DenseNet121	0.254	0.628	0.981	0.453	0.795	0.692	0.476	0.405	0.978	0.592	0.737	0.496	0.535	0.617
DenseNet161	0.237	0.643	0.981	0.467	0.795	0.692	0.407	0.433	0.978	0.578	0.734	0.477	0.464	0.607
InceptionV4	0.260	0.629	0.981	0.467	0.792	0.691	0.321	0.388	0.978	0.585	0.745	0.488	0.434	0.597
ResNeXt101_32	0.202	0.665	0.982	0.433	0.820	0.731	0.336	0.460	0.979	0.634	0.750	0.515	0.466	0.613
ResNeXt101_64	0.270	0.652	0.982	0.426	0.804	0.719	0.453	0.436	0.979	0.607	0.736	0.498	0.418	0.614
SE-ResNet101 [11]	0.237	0.643	0.981	0.439	0.801	0.711	0.313	0.414	0.978	0.586	0.729	0.457	0.454	0.596

[a] Sp, Ca, Co, Ja, Ko, We, Br, Gi, La, Si, Nu, St, Ni, MF2 are abbreviations of sports, casual, commuting, Japanese, Korean, western, British, girls, ladies, simple, natural, Street, national respectively.

IV. Ensemble

We utilized Cyclic learning rate(CLR) and Stochastic weight averaging(SWA) for ensemble. SWA [8] is a run-time ensemble method which can lead to a wider, more generalized model with extra parameters. The intuition of SWA comes from empirical observation that local minima at the end of each learning rate cycle tend to accumulate at the border of areas on loss surface, taking the average of several such points could achieve a wider solution. formally, it works like:

$$w_{SWA} \leftarrow \frac{w_{SWA} \cdot n_{models} + w}{n_{models} + 1} \tag{6}$$

where w is the model traverse the weight space, w_{SWA} is the model store averaged parameters used for prediction. In my experiment, SWA benefit DenseNet121, DenseNet161, InceptionV4 model, but I failed to implement it in ResNeXt101_32x4d, ResNeXt101_64x4d. The performance improvement of SWA is shown in Fig. 4.

After that, we use 5 models are shown in Fig. 3 to vote for decision, votes of one label greater than 3 will be written to true, otherwise false. Finally, the performance on test set is 0.6498.

V. Conclusion

Multi-task style recognition is affected by various factors like clothes parts, clothes combination, context, human posture etc. There are massive information worth to be mined here and computer vision can play an important role in this task. In this paper, our strategy incorporating WBCE, threshold optimization, SWA and voting ensemble is introduced, which is proved to be able to perform much better than baseline resnet18 model and achieve 0.6498 mean F2 score and 4th place in the competition, especially WBCE and SWA can increase the performance a lot without add extra computational burden.

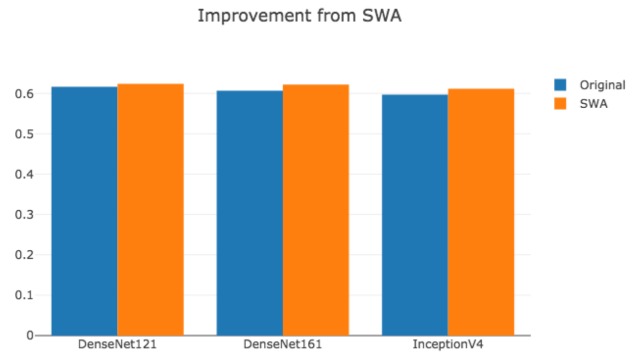

Fig. 4. Improvement after SWA.

There is still a lot of room for improvement. For example, The dependency and exclusiveness between style labels could be utilized as prior Information which could further boost the performance. Fashion keypoints oriented attention model [12] also have potentialities to filter noisy information effectively. Some common tricks like test time augmentation(TTA), online hard example mining(OHEM), larger train set proportion could also of value to this task.

Acknowledgment

This work is supported by National Natural Science Foundation of China (Grant No.61572124).

References

[1] Y. Hu, X. Yi, and L. S. Davis, “Collaborative fashion recommendation: A functional tensor factorization approach,” in ACM International Conference on Multimedia, 2015, pp. 129–138.

[2] J. Mcauley, C. Targett, Q. Shi, and A. V. D. Hengel, “Image-based recommendations on styles and substitutes,” in International ACM SIGIR Conference on Research and Development in Information Retrieval, 2015, pp. 43–52.

[3] R. He, C. Packer, and J. Mcauley, “Learning compatibility across categories for heterogeneous item recommendation,” in IEEE International Conference on Data Mining, 2017, pp. 937–942.

[4] Z. Alhalah, R. Stiefelhagen, and K. Grauman, “Fashion forward: Forecasting visual style in fashion,” in IEEE International Conference on Computer Vision, 2017, pp. 388–397.

[5] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[6] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-resnet and the impact of residual connections on learning.” in AAAI, vol. 4, 2017, p. 12.

[7] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. IEEE, 2017, pp. 5987–5995.

[8] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, and A. G. Wilson, “Averaging weights leads to wider optima and better generalization,” arXiv preprint arXiv:1803.05407, 2018.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[10] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” IEEE transactions on pattern analysis and machine intelligence, 2018.

[11] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” 2018.

[12] W. Wang, Y. Xu, J. Shen, and S.-C. Zhu, “Attentive fashion grammar network for fashion landmark detection and clothing category classification,” 2018.