IDE RankNet: Estimating the difficulty of visual search in an image

Dongliang Ma

School of Internet of Things Engineering

Jiangnan University

Wuxi, Jiangsu, China

mdl.viper@gmail.com

Haipeng Zhang

School of Internet of Things Engineering

Jiangnan University

Wuxi, Jiangsu, China

zhpmatrix@gmail.com

Jun Sun

School of Internet of Things Engineering

Jiangnan University

Wuxi, Jiangsu, China

sunjun_wx@hotmail.com

***Abstract*—Estimating the difficulty of visual search in images is an interesting work that can be applied to weakly supervised object localization and semi-supervised object classification, and has potential applications in object detection. In this paper, we proposed a simple loss function based on learning to rank and applied it to an end-to-end multitask neural network for estimating difficulty scores of images. Our model shows better results for predicting the ground-truth visual search difficulty scores produced by human annotators in PASCAL VOC2012.**

Keywords—Visual search, Multitask Learning, Estimating difficulty score, Neural network.

I. INTRODUCTION

Humans can intuitively understand the content of images, and often reach a consensus that some images are more difficult to visual search tasks than others. However, this kind of task is quite challenging for computer. It is a subjective task which may be influenced by human emotional factors. Estimating the difficulty of visual search in an image may be affected by the following factors: background, complexity of scene, amount of objects, and whether they are occluded.

At present, there is little work on the topic of image difficulty[1][2][3]. Russakovsky et al.[1] measure difficulty by the size and number of ground-truth bounding-boxes in order to quantify difficulty (even at test time). Vijayanarasimhan and Grauman[2] attempt to predict the difficulty of an image base on the time it takes for humans to segment it, however the image segmentation task[2] is conceptually different from our visual search task. For example, a truncated object can be easily segmented but difficult to find and identify. Radu et al.[3] use the specially established dataset that close to human visual level to estimate the difficulty of images(Figure 1). They build a regression model based on deep features learned with convolutional neural network. Moreover, they regard the neural network as a feature extractor to get information to the regression model. Different from Radu's works, we have established a new end-to-end neural network regression model to predict the difficulty scores of images, so that the model can automatically predict the human's assessment of the difficulty of visual search in images.

Estimating the difficulty of images can be used as a criterion for distinguishing images. Some previous works have used it for weakly supervised object localization[5][6] and semi-supervised object classification[3], and potential application in object detection[7][8]. We used the PASCAL VOC2012[4] images with the difficulty annotations established by Radu to delve into the relationship between image and difficulty. In terms of the estimating difficulty of visual search in images, we have three main contributions in this paper:

(1)We proposed an end-to-end neural network model to estimate the difficulty scores of images. We named it the IDE RankNet(Image Difficulty Estimating RankNet).

(2)We designed Kendall RankLoss for optimizing the Kendall's τ correlation coefficient[9].

(3)We used a multitask alternative strategy to train our model.

The rest of the paper is organized as follows. Section Ⅱ introduces related works. Section Ⅲ focuses on our work. Section Ⅳ reports our experimental results. Finally, section Ⅴ draws conclusions and future works introductions.

II. RELATED WORK

A. Learning to Rank

Learning to rank[36] has attracted widespread attention in the field of machine learning and has been applied in the fields of information retrieval[34], collaborative filtering[35], and recommendation algorithms[35]. Using rank information may produce better results. According to the different training samples, it can be roughly divided into three categories:

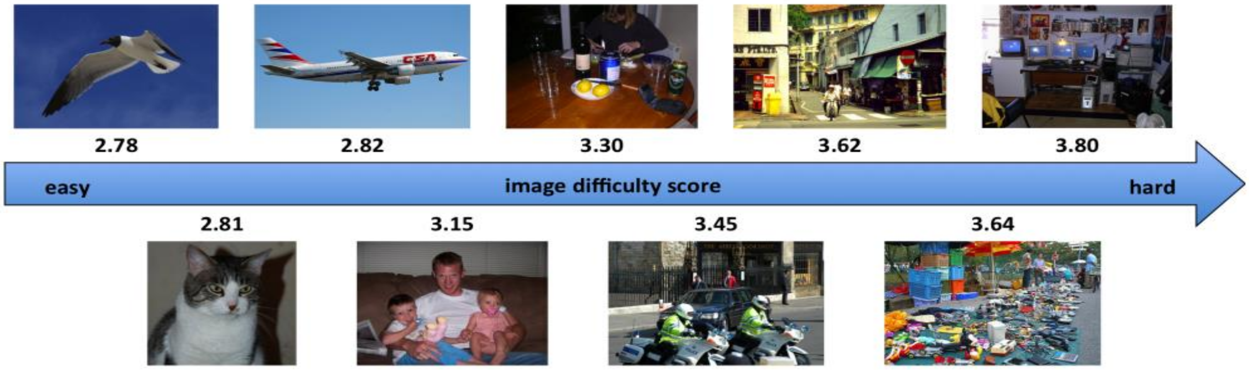


Figure 1. Dataset created by Radu et al. Each image has a corresponding difficulty score which is ranged between two and nine.The harder it is to identify the object within the image, the higher the corresponding difficulty score, and vice versa.

(1) Pointwise, the rank function is obtained from the sample of the dataset. The main solution is to convert the classification problem of the dataset into the classification and regression problem of a single sample. The algorithms applied to pointwise mainly include Discriminative model for IR[10], McRank[11] and so on.

(2)Pairwise, which takes two samples with different annotations in the dataset as sample pairs, and then converts the problem based on the sample pairs into a binary classification problem. The main application algorithms are: rankSVM[12], RankBoost[13], RankNet[14] and so on.

(3)Listwise, the sequence of the entire dataset is regarded as a sample, mainly by directly optimizing the evaluation method and defining the loss function to obtain the loss function. The main application algorithms are: AdaRank[15], SVM-MAP[16], ListNet[17], ListMLE[18] and so on.

In the field of computer vision facial age estimation, the rank information is widely used, and [19][20][21][22] have achieved good results. Inspired by their work, we used the rank information of the images to train neural network and designed a special loss function for the Kendall's τ correlation coefficient to optimize the entire network in the visual search task.

B. *Multitask learning*

Compared to training separate models for each task, multitask learning is designed to improve the learning efficiency and predictive accuracy of each task[39][40]. It can be thought of as an approach to inductive transfer, which improves generalization by sharing the domain information between other related tasks. It does this by using a shared representation to learn multiple tasks – what is learned from one task can help learn other tasks[23][41]. At present, multitask deep learning has been widely used in many fields such as object detection[24], face recognition[25], fine-grained vehicle classification[26], facial key points in attribute classification[27]. The basic assumption of multitask learning is that there is a correlation between multiple tasks, and the model can use the correlation between tasks to promote learning. Estimating of the difficulty in images has strong order, and the images with more information are more difficult, and the others with less information are less difficult, as shown in Figure 1. When the model approximates the rank information of the predicted difficulty scores and the rank information of the ground-truth, the generalization ability of the model is stronger. Therefore, combining with rank information may improve the generalization ability of the model. We validate our hypothesis in the experimental part of Section Ⅳ.

III. OUR APPROACH

A. *Network structure*

We use ResNet152[28] which has strong performance presently as the backbone of the IDE RankNet, and change the number of output nodes of the last full connection layer in the original model to 1, so that the network can output the difficulty score directly. At the same time, same as most object detection and segmentation tasks, we used ImageNet[29] pre-trained model parameters for initialization, and then fine-tuned the entire network to make the model fit our work. It is worth mentioning that we resize the images to 224*224 to fit the number of nodes in fully connected layer in IDE RankNet.

B. *Kendall RankLoss*

When we are training the model, if the batch size of the neural network is large enough, we can forward the difference between the rank information of the predicted difficulty scores and the ground-truth rank information in each batch as the error to backpropagate to the network, the entire network can continuously reduce the error.

By the definition of Kendall's τ correlation coefficient, the range of its value is between -1 and 1. When the rank information of all samples is exactly the same as the rank information of the real values, Kendall's τ correlation coefficient takes a value of 1, and vice versa. Supposedly, there are n samples for each mini-batch $\{x_{i,}y_{true_i}\}_{i=1}^{N}$, where x_i represents the i-th image in the batch, y_{true_i} represents the true label and $y_{predict_j}$ represents the corresponding predicted difficulty score by IDE RankNet, then we given the definition of Kendall's τ correlation coefficient on the batch as follows:

$$\tau = \frac{\sum_{i\in N}\sum_{(j<i)\in N} sgn\left(y_{true_i} - y_{\mathrm{true_j}}\right) sgn\left(y_{\mathrm{predict_i}} - y_{predict_j}\right)}{n(n-1)/2} \qquad (1)$$

we designed Kendall RankLoss for optimizing Kendall's τ correlation coefficient:

$$1-\frac{\sum_{i\in N}\sum_{(j<i)\in N} sgn\left(y_{true_i}-y_{true_j}\right) sgn\left(y_{predict_i}-y_{predict_j}\right)}{n(n-1)/2} \quad (2)$$

$$= 1 - c\sum_{i\in N}\sum_{(j<i)\in N} sgn\left(y_{true_i}-y_{true_j}\right) sgn\left(y_{predict_i}-y_{predict_j}\right)$$

As shown in equation (2), since the batch size is determined, we can replace it with a constant term. The interval of the RankLoss function is [0, 2]. We should minimize the RankLoss function during training. We use ω instead of the entire network parameters. The network update process is:

$$\frac{\partial rankloss}{\partial \omega} = -c\sum_{i\in N}\sum_{(j<i)\in N} sgn'\left(\frac{\partial y_{predict_i}}{\partial \omega} - \frac{\partial y_{y_{predict_j}}}{\partial \omega}\right) \quad (3)$$

$$\omega = \omega - \eta \frac{\partial rankloss}{\partial \omega} \quad (4)$$

In addition, we know that the loss function not only needs to meet the target task, but also needs to be derivative, so that the purpose of updating the network parameters can be achieved in the back propagation phase by the gradient descent method. As shown in Figure 2(a), the sgn function is non-differentiable, and Kendall RankLoss does not make the entire network effective learning.

Therefore, we made the following change to Kendall RankLoss: replaced the sgn function with the hardtanh function (as shown in Figure 2(b)), so that $rankloss \approx rankloss_{modified}$

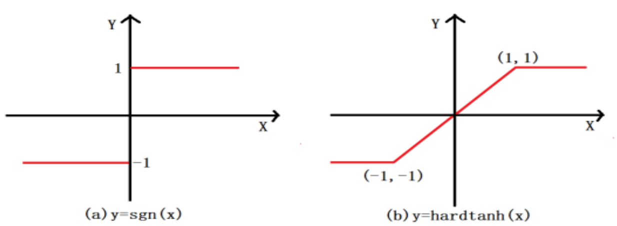


Figure 2. (a) the sign function of a real number x is -1 when x is negative, 0 when x is zero, or +1 when x is positive. (b) the hardtanh function of a real number x is -1 when x is less than -1, or +1 when x is greater than +1. Otherwise the hardtanh function is equivalent to the function y=x.

C. Multitask alternative strategy

According to experience, we believe that when the mean square error between the predicted values and the ground-truth becomes smaller, the difference has the same trend between the rank information of the predicted values and the ground-truth. We hope that the generalization ability of the model is better, whether it is reflected by the mean square error or the Kendall's τ correlation coefficient or other aspects. However, there are also differences between MSE and Kendall's τ correlation coefficient. For example, suppose we have three items in our set whose true value label is $\varphi_{true} = \{2, 3, 4\}$, for each of the two different models $\{\mu_1, \mu_2\}$, the predicted values for the sets are $\varphi_1 = \{2, 4, 3\}$ and $\varphi_2 = \{6, 8, 10\}$. It is clear that μ_2 is better than μ_1 in the performance of the rank; but in terms of MSE, the performance of μ_1 is better. Usually, tasks with certain correlations are trained in the same network, and better generalization performance can be achieved. Therefore, in this paper, we use multitasking to improve the generalization performance of our model.

On the other hand, for $\forall(i, j) \in N$, the Kendall RankLoss designed in this paper can only be used for backpropagation under the condition of $\left|y_{predict_i} - y_{predict_j}\right| \le 1$. This $rankloss_{modified}$ does not completely solve the non-differentiable problem. In addition, the value range of Kendall RankLoss we designed is [0,2] , and the range of MSE is$(0, +\infty)$; what's more, Kendall RankLoss is affected by the interaction between the predictions of the same batch, so its convergence is slow. Drawing on the previous work which solved the ranking problems by using the gradient[30], we used multitask alternative strategy to train our image difficulty estimation model. Algorithm 1 shows the learning algorithm for the IDE RankNet.

Algorithm 1 Learning Algorithm for the IDE RankNet.

Input: training data $\{(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)\}$
Parameter: number of iterations T, learning rate η,
Initialize parameters ω , the set of loss functions $\{loss_{mse}\ , rankloss_{modified}\}$, batch size n, alternative frequency f.
for t=1 to T do
    if t % f = 0
        $loss = loss_{mse}$
    else
        $loss = rankloss_{modified}$
    for n to m do
        Input data $\{(x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)\}$ of a batch to Neural Network and compute the difficulty score with current ω
        Compute gradient Δω using loss
        Update $\omega = \omega - \eta\Delta\omega$
    end for
end for

Output Neutral Network model ω

IV. EXPERIMENTS

A. Evaluations Metrics

The same as Radu et al.'s work[3], we use MSE (Mean Squared Error) and Kendall's τ correlation coefficient to evaluate the generalization performance of the model. MSE is one of the most commonly used metrics in model evaluation, which reflects the degree of value of difference between predicted difficulty scores and ground-truth. The smaller the value of MSE, the better the accuracy of the model describing the experimental data. MSE is related to ground-truth. You get a smaller MSE if you choose to scale the labels between 0 and 1. Unlike MSE, Kendall's τ correlation coefficient is not affected. We use Kendall's τ correlation coefficient as a measure of similarity between predictions and labels.

B. Benchmark Dataset

The images from PASCAL VOC2012 in the benchmark dataset are used as the difficulty dimensioning source. The dataset contains a total of 11540 training and test images, and the dataset contains 20 objects (including airplanes, boats, cats, dogs, etc.) that have been labeled with categories, outlines and borders. This task is on a crowd-sourcing platform named CrowdFlower, after 736 trusted annotator observe the information in the image, the time required to answer the question is used as a measure of the image difficulty and convert it into image difficulty score. They designed a series of related processing methods, such as clearing outliers to ensure data reality. Based on the usual visual search tasks, they propose an explanation of the difficulty of the image close to the human visual level. Since the background of each image in the PASCAL VOC2012 dataset is different, the density of the objects is different, the number, size and appearance of the objects are different, we can apply this dataset to our visual search task.

In this paper, the dataset is divided into training, validation, and test sets in a 2:1:1 ratio using the same way as the previous work. We trained our model using 5,770 images in the training set, and use others to test and validate the MSE and Kendall's τ correlation coefficients.

C. Implementation Details

The proposed model is implemented on an open source deep learning library named PyTorch[38]. When trained with a batch size of 200 on 4 GPUs (Tesla K80), the whole training processing takes about 7~8 hours. For optimization algorithm, we used SGD with momentum = 0.9 and weight decay = 5×10^{-4} for adjusting learning rate. Learning rate is set to 1×10^{-3} for the first 160 iterations, and decayed by 10 at 200 and 240 iterations. Noticeably, we set the alternative frequency is 3, which means 2 epochs for training Kendall RankLoss and 1 epoch for MSE repeatedly, this parameter is found by grid search. Details will be described in each experiment individually.

D. Baseline

We have tried four popular backbone networks of MobileNet[31], Vgg16[32], ResNet101 and ResNet152[28]. The results are shown in Table 1. Therefore, we used the ResNet152 as the backbone of the image difficulty estimating task.

E. Data augment

The same as most computer vision tasks, we also use data augment to improve the generalization performance of the model in the image preprocessing stage. The data augment methods we used are: randomly changing the contrast, brightness, hue, saturation of the image, flipping the image, and noise perturbating each pixel in the image. Through data augment, we improve the generalization of the model to (MSE：0.200 Kendall's τ : 0.405).

TABLE 1. RESULTS OF MSE AND KENDALL'S τ CORRELATION COEFFICIENTS WITH DIFFERENT BACKBONE NETWORKS

	Backbone networks	***Metrics***	
		MSE	***Kendall's τ***
1	MobileNet	0.348	0.203
2	Vgg16	0.259	0.335
3	ResNet101	0.243	0.372
4	ResNet152	0.223	0.387

F. Multitasking Alternative strategy to train the IDE RankNet

With a single task training network, you can often only get the local minimum of the tasks. Even the local minimum that may be achieved between related tasks may not be the same. Multitask learning can make the generalization performance of the whole model tradeoff between different tasks, so we can use a certain strategy to make an acceptable local optimal solution between MSE and Kendall's τ correlation coefficients.

We have tried the polynomial weighting method commonly used in computer vision multitasking to design the loss function, as shown in equation (5):

$$\text{loss} = \lambda * rankloss_{modified} + (1-\lambda) * loss_{mse} (\lambda \in [0,1]) \quad (5)$$

Kendall's τ correlation coefficients is less affected by lambda than MSE, it also indirectly illustrates the problem that Kendall RankLoss is difficult to be optimized, as illustrated in Figure 3. At the same time, we found that MSE slowed down relatively fast in early training, while Kendall RankLoss had a slower gradient update throughout the training. Therefore, we adopt the multitask alternative strategy to train our network, using Kendall RankLoss as the main task, and MSE as the auxiliary task, increasing the frequency of use of Kendall RankLoss, to make the IDE RankNet get an acceptable local minimum between the Kendall's τ correlation coefficients and the MSE metrics.

G. Result analysis

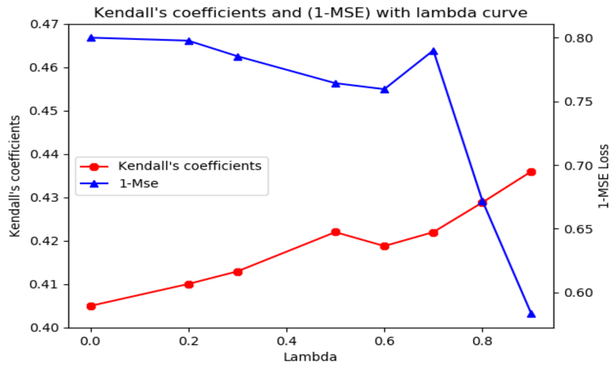
As shown in Table 2, ablation experiments have been done to analyze IDE RankNet models. Exp.1 of the best end-to-end model is used as the basic setting for comparison.

1) Kendall RankLoss is essential

In Exp.2, we use Kendall RankNet to optimize Kendall's τ correlation coefficient. The RankLoss is very important because it uses rank information to optimize our model. With RankLoss, we can increase Kendall's τ correlation coefficient to 6.4%.

2) Multitask Learning contributes to a better model

In Exp.3, we use a multitask joint training network. With multitask learning, we can increase the Kendall's τ correlation coefficient by 1.2% with a slight increase in MSE.



Task weights Lambda	Task weights 1-Lambda	Kendall's τ	1-MSE
0	1	0.405	0.800
0.2	0.8	0.410	0.797
0.3	0.7	0.412	0.785
0.5	0.5	0.422	0.764
0.6	0.4	0.418	0.759
0.7	0.3	0.425	0.790
0.8	0.2	0.428	0.672
0.9	0.1	0.435	0.583

Figure 3. Polynomial loss function: Kendall RankLoss and MSE with lambda curve. The figure and table illustrate the advantages of multi-task learning for optimizing Mse and Kendall's τ correlation coefficients. We use the formula of 1-mse to observe the regulation. It shows that using task uncertainty weights can even perform better compared to single task learning. We can get a relatively satisfactory result with lambda = 0.7 through fine-grained grid search.

3) Multitask alternative strategy matters

In Exp.4, we used multitask alternative strategy for training, resulting in an obvious improvement on Kendall's τ correlation coefficient, and leading to a slight improvement on MSE. Since Kendall's τ correlation coefficient is difficult to optimize, we believe that this operation is effective.

Figure 4 shows the correlation between the ground-truth and the predicted difficulty scores. The cloud of points forms a slanted Gaussian with the principal component oriented almost diagonally, indicating a strong correlation between the predicted and the ground-truth scores.

TABLE 2. INVESTIGATION OF IDE RANKNET WITH DIFFERENT SETTINGS ON TEST DATASET.

	Backbone networks	Metrics: MSE	Metrics: Kendall's τ
1	ResNet152 + MSE	0.200	0.405
2	ResNet152 + Kendall RankLoss	17.04	0.464
3	ResNet152 + (MSE + KE, lambda=0.7)	0.21	0.425
4	ResNet152 + (MSE / KE, f = 3)	0.222	0.467

H. Other work

During the whole experiment, we found that there is a serious data imbalance problem in the dataset. As shown in Figure 4, the number of images with high difficulty score is less, and there is no better result for this part of the data. We tried to solve this problem by upsampling, adding Gaussian noise [33], SMRT [34], and etc, but did not make a good progress. In future, we will also delve into the issue.

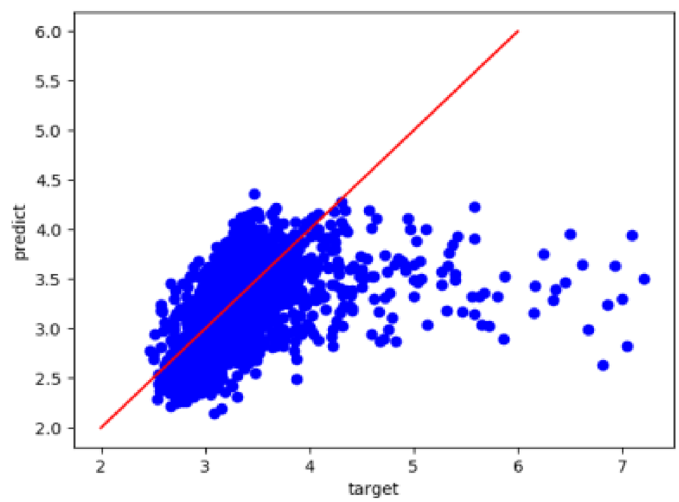


Figure 4. Correlation between ground-truth and predicted difficulty scores. The least squares regression line is almost diagonal suggesting a strong correlation.

V. CONCLUSION AND FUTURE WORK

IDE RankNet, a novel image difficulty estimating algorithm is proposed in this paper. Different from prevalent image difficulty estimating methods, IDE RankNet makes use of rank information and alternative training strategy to get a better result.

Previous work has confirmed that this task can be used as an heuristic method for distinguishing images, which has achieved good results in object localization, semi-supervised object classification and object detection in weak supervised learning. Applications of IDE RankNet will be explored on some other tasks.

REFERENCES

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, K. A., A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 2015.

[2] S. Vijayanarasimhan and K. Grauman. Whats It Going to Cost You?: Predicting Effort vs. Informativeness for MultiLabel Image Annotations. In Proceedings of CVPR, 2009.

[3] Ionescu, Radu Tudor, et al. "How Hard Can It Be? Estimating the Difficulty of Visual Search in an Image." computer vision and pattern recognition (2016): 2157-2166.
[4] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 Results, 2012.
[5] Shi M, Ferrari V. Weakly Supervised Object Localization Using Size Estimates[J]. 2016.
[6] Tao Q, Yang H, Cai J. Exploiting Web Images for Weakly Supervised Object Detection[J]. 2017.
[7] Wang, Keze, et al. "Towards Human-Machine Cooperation: Self-supervised Sample Mining for Object Detection." (2018).
[8] Soviany P, Ionescu R T. Optimizing the Trade-off between Single-Stage and Two-Stage Object Detectors using Image Difficulty Prediction[J]. 2018.
[9] Croux C, Dehon C. Influence functions of the Spearman and Kendall correlation measures.[J]. Statistical Methods & Applications, 2010, 19(4):497-515.
[10] Nallapati R. Discriminative models for information retrieval[C]// ACM, 2004:64-71.
[11] Li P, Burges C, Wu Q. McRank: Learning to rank using multiple classification and gradient boosting [C]// Proceedings of the 21st Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, 2007:845-852
[12] Cao Y, Xu J, Liu T Y, et al. Adapting ranking SVM to document retrieval[C]// International Acm Sigir Conference on Research & Development in Information Retrieval. ACM, 2006:186-193.
[13] Freund Y, Iyer R, Schapire R E, et al. An efficient boosting algorithm for combining preferences[J]. Journal of Machine Learning Research, 1999, 4(6):170--178.
[14] Burges C, Shaked T, Renshaw E, et al. Learning to rank using gradient descent[C]// International Conference on Machine Learning. ACM, 2005:89-96.
[15] Xu, Jun, Li, et al. AdaRank: a boosting algorithm for information retrieval[J]. Sigir, 2007:391-398.
[16] Yue Y, Finley T, Radlinski F, et al. A support vector method for optimizing average precision[C]// International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2007:271-278.
[17] Cao Z, Qin T, Liu T Y, et al. Learning to rank:from pairwise approach to listwise approach[C]// International Conference on Machine Learning. ACM, 2007:129-136.
[18] Xia F, Liu T Y, Wang J, et al. Listwise approach to learning to rank:theory and algorithm[C]// International Conference on Machine Learning. ACM, 2008:1192-1199.
[19] D. Cao, Z. Lei, Z. Zhang, J. Feng, and S. Li. Human age estimation using ranking svm. CCBR, pages 324–331, 2012.
[20] K. Chang, C. Chen, and Y. Hung. Ordinal hyperplanes ranker with cost sensitivities for age estimation. CVPR, pages 585–592, 2011.
[21] C. Li, Q. Liu, J. Liu, and H. Lu. Learning ordinal discriminative features for age estimation. CVPR, pages 2570–2577, 2012.
[22] Niu Z, Zhou M, Wang L, et al. Ordinal Regression with Multiple Output CNN for Age Estimation[C]// Computer Vision and Pattern Recognition. IEEE, 2016:4920-4928.
[23] Caruana, R. (1997). Multitask Learning. Machine Learning, 28(1), 41–75. doi: 10.1023/A:1007379606734.
[24] Sun Y, Wang X, Tang X. Deep Learning Face Representation by Joint Identification-Verification[J]. 2014, 27:1988-1996.
[25] Zhang X, Zhou F, Lin Y, et al. Embedding Label Structures for Fine-Grained Feature Representation[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2016:1114-1123.
[26] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]// International Conference on Neural Information Processing Systems. MIT Press, 2015:91-99.
[27] Zhang Z, Luo P, Chen C L, et al. Facial Landmark Detection by Deep Multi-task Learning[C]// European Conference on Computer Vision. 2014:94-108.
[28] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2016:770-778.
[29] Deng J, Dong W, Socher R, et al. ImageNet: A large-scale hierarchical image database[C]// Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009:248-255.
[30] Caruana R, Baluja S, Mitchell T. Using the Future to "Sort Out" the Present: Rankprop and Multitask Learning for Medical Risk Evaluation[J]. Proceedings of Nips', 1995, 8:959--965.
[31] Howard A G, Zhu M, Chen B, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications[J]. 2017.
[32] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
[33] Liu R, Hall L O, Bowyer K W, et al. Synthetic minority image over-sampling technique: How to improve AUC for glioblastoma patient survival prediction[C]// IEEE International Conference on Systems, Man and Cybernetics. IEEE, 2017:1357-1362.
[34] Salton G, Harman D. Information retrieval[J]. 2003:777.
[35] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C]// International Conference on World Wide Web. ACM, 2001:285-295.
[36] Liu T Y. Learning to rank for information retrieval[J]. Foundations and Trends in Information Retrieval, 2009, 3(3):225-331.
[37] Synthetic Minority Reconstruction Technique, https://github.com/tgsmith61591/smrt.
[38] Paszke, A., Gross, S., Chintala, S., Chanan, G.: Pytorch (2017)
[39] [39] S. Thrun. Is learning the n-th thing any easier than learning the first? In Advances in neural information processing systems, pages 640–646. MORGAN KAUFMANN PUBLISHERS, 1996. 2
[40] J. Baxter et al. A model of inductive bias learning. J. Artif. Intell. Res.(JAIR), 12(149-198):3, 2000. 2
[41] Ruder S. An Overview of Multi-Task Learning in Deep Neural Networks[z]. 2017.