# Deep Partial Rank Aggregation for Personalized Attributes

**Qianqian Xu [1], Zhiyong Yang [2,3], Zuyao Chen [4], Yangbangyan Jiang [2,3], Xiaochun Cao [2,3,6], Yuan Yao [7], Qingming Huang [1,4,5,6]**

[1] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, China
[2] State Key Laboratory of Information Security, Institute of Information Engineering, CAS, Beijing, China
[3] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[4] School of Computer Science and Technology, University of Chinese Academy of Sciences, Beijing, China
[5] Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing, China
[6] Peng Cheng Laboratory, Shenzhen, China
[7] Department of Mathematics, Hong Kong University of Science and Technology, Hong Kong, China
xuqianqian@ict.ac.cn, {yangzhiyong, jiangyangbangyan, caoxiaochun}@iie.ac.cn, chenzuyao17@mails.ucas.ac.cn, yuany@ust.hk, qmhuang@ucas.ac.cn

## Abstract

In this paper, we study the problem of how to aggregate pairwise personalized attributes (PA) annotations (*e.g.*, Shoes A is more comfortable than B) from different annotators on the crowdsourcing platforms, which is an emerging topic gaining increasing attention in recent years. Given the crowdsourced annotations, the majority of the traditional literature assumes that all the pairs in the collected dataset are distinguishable. However, this assumption is incompatible with how humans perceive attributes since indistinguishable pairs are ubiquitous for the annotators due to the limitation of human perception. To attack this problem, we propose a novel deep prediction model that could simultaneously detect the indistinguishable pairs and aggregate ranking results for distinguishable pairs. First of all, we represent the pairwise annotations as a multi-graph. Based on such data structure, we propose an end-to-end partial ranking model which consists of a deep backbone architecture and a probabilistic model that captures the generative process of the partial rank annotations. Specifically, to recognize the indistinguishable pairs, the probabilistic model we proposed is equipped with an adaptive perception threshold, where indistinguishable pairs could be automatically detected when the absolute value of the score difference is below the learned threshold. In our empirical studies, we perform a series of experiments on three real-world datasets: LFW-10, Shoes, and Sun. The corresponding results consistently show the superiority of our proposed model.

## Introduction

Personalized attributes (PA) are semantic features describable in words, such as texture, color, mood. Typical instances include comfortable or high heeled for shoes, and smiling or crying for human faces, etc. Introducing PA to multimedia/computer vision community opens up a number of interesting possibilities, as shown in recent literature (Fu et al. 2014, 2016; Singh and Lee 2016; Kovashka and Grauman 2017; Parikh and Grauman 2011; Jing et al. 2017; Bampis et al. 2018; Squalli-Houssaini et al. 2018). For example, estimating interestingness attribute (Fu et al. 2014) from images/videos would be helpful for media-sharing websites (*e.g.*, Youtube); estimating attributes of consumer goods such as shininess of shoes (Fu et al. 2016) plays a central role in improving online shopping experiences; other applications might include web advertising and video summarization. And the list goes on.

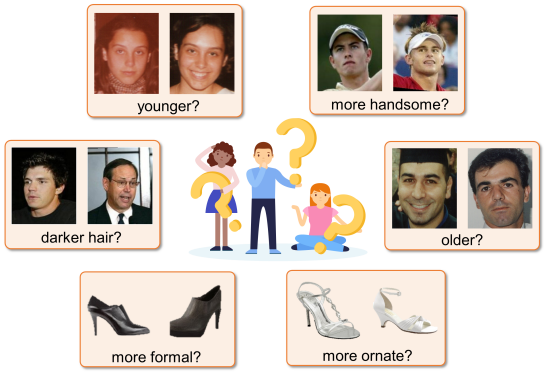


Figure 1: Examples of indistinguishable pairwise comparisons.

*Given the importance of PA, our first problem starts with how to measure the strength of such attributes quantitatively*. At the first glance, one can realize it by simply specifying a discrete score. For example, one could score the interestingness of a movie as $1, 2, 3, \cdots, 5$, with 5 representing the most interesting ones and 1 representing the least interesting ones. However, the potential issue is that different people often exhibit dissimilar interpretations of the score. For instance, some annotators think score 3 is large enough for good movies, while the others even give a boring movie the same score. This makes such a scoring rule not suitable, especially when we are searching for a consensus over personalized opinions. In order to obtain more reliable annotations

Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

and thus learn better aggregation models, recent studies turn to an alternative approach with pairwise comparisons. In a pairwise comparison test, an individual is asked to choose which one has a stronger presence of a given attribute. Since both objects in a given pair take the other one as a reference point, the scaling issue is thus largely alleviated. However, introducing pairwise comparisons has its own problem, that is, not all the pairs are distinguishable. As shown in Fig.1, indistinguishable pairs are actually ubiquitous in the crowdsourcing platforms. The annotators often have no clue about how to make decision on such queries. Facing this trouble, we could allow the annotators to abstain from the choice, when it is too hard to tell which one in a given pair is really better. In fact, we could find a variety of visual applications, where a "I can't decide" option is inevitable. For example, in subjective multimedia quality assessment (Chen et al. 2009; Xu et al. 2011), videos and images of the same content are to be evaluated for its quality; in online shopping systems, one would like to choose the most comfortable shoes out of a given pair; in the human-age estimation task, the users are required to choose the younger person. In all these scenarios, some pairs are easy to distinguish, while others are not.

In all these examples, if a rater is not sufficiently certain regarding the relative order of the two items, he may abstain from his choice decision and instead declare these two as indistinguishable, which we call partial ranking. In this way, a dataset with abstention of this kind provides us information about possible ties or equivalent classes of items in partial orders.

Given the above arguments, in this paper, we measure the strength of attributes with the pairwise comparisons with abstention. Based on this setting, our goal is to aggregate the personalized annotations of such partial rankings on the crowdsourcing platforms, which leads to consensus comprehensions of the attributes.

Though there is a considerable amount of work on pairwise ranking (Fu et al. 2014), the literature on learning partial rankings from such pairwise comparison data with abstentions is relatively sparse. In (Cheng et al. 2010), it produces partial ranking by thresholding a (valued) pairwise preference relation, *i.e.*, by a "$\alpha$-cut" of preference relation. However, it leaves the optimal choice of hyper-parameter $\alpha$ to various heuristics and needs to know in advance the preference relation between every pair of items (*i.e.*, $n(n-1)/2$ pairs in total for $n$ items), which requires a large number of comparisons, being too prohibitive in modern applications. To fill in this gap, in (Xu et al. 2018), it proposes a novel framework to learn partial ranking based on extended probabilistic models, in which the threshold $\alpha$, can be automatically learned from pairwise comparison data with abstentions via convex optimization. Moreover, (Yu and Grauman 2015) explores this problem from a "Just Noticeable Difference (JND)" perspective to decide whether a difference in PA is perceptible.

These work mentioned above, either does not have prediction power (Cheng et al. 2010; Xu et al. 2018), or aims to predict the partial rankings based on limited representation of low-level image features (Yu and Grauman 2015). Different from these work, our goal in this paper is to leverage the strong representation power of deep neural networks to aggregate the partial ranks for PA from a deep perspective. As an overall summary, we list our main contributions as follows:

- Based on the crowdsourcing data, we propose a deep framework to aggregate pairwise comparisons for personalized attributes when some of the pairs are suffering from an indistinguishable difference. To the best of our knowledge, our framework offers the first attempt for partial ranking prediction in the presence of indistinguishable/indistinguishable pairs with a deep end-to-end framework.
- In the core of the framework lies the unified probabilistic model, which formulates the annotation process of the users with specific consideration of abstention. Different from the majority of the traditional methods, the proposed framework could simultaneously learn to detect indistinguishable pairs and to predict the aggregated results for distinguishable pairs. Based on this model, we propose a novel loss function with the Maximum Likelihood Estimation framework.
- Moreover, we also propose a prediction scheme for the comparison results of unseen image pairs, without the help of crowdsourcing annotations.

## Related Work

**Personalized Attributes.** As mentioned in our introduction, PA has been widely studied in recent years. It has inspired a number of useful applications, including image/video interestingness (Fu et al. 2014), memorability (Jing et al. 2017; Squalli-Houssaini et al. 2018), and quality of experience (Chen et al. 2009; Bampis et al. 2018) prediction, etc. Typically one learns PA in the learning-to-rank setting: the training data is ordered (*e.g.*, we are told image A has it more than B), and a ranking function is optimized to preserve those orderings. This could be realized via popular learning to rank models such as RankSVM (Joachims 2002), RankBoost (Freund et al. 2003), RankNet (Burges et al. 2005), GBDT (Friedman 2001), and DART (Rashmi and Gilad-Bachrach 2015). However, such models assume that all images are orderable. However, this assumption is inconsistent with humans perception. In fact, 40% of the time human asked to compare images for a relative attribute declare that no difference is perceptible (Yu and Grauman 2014). As shown in Fig.1, within a given attribute, sometimes we can perceive a comparison, sometimes we can not. To address this issue, (Yu and Grauman 2015) develops a non-parameterized Bayesian local learning strategy to separate distinguishable from indistinguishable pairs at test time. Different from this traditional line of research, we study the partial rank aggregation problem under the context of deep learning. With a parameterized style, our model scales better than the non-parameterized model. Moreover, equipped with better feature representation power, we show experimentally that the proposed end-to-end framework could achieve better ranking prediction.

**Learning with a Reject Option.** The notion of abstention could be traced back to the classification community (Chow

1970), where abstention is often formulated as a reject option. Specifically, a classifier might reject to decide a class prediction if making no decision is considered less harmful than making an unreliable and hence potentially false decision. Nowadays, this framework has been successfully applied to a wild range of classification variants including binary classification (Herbei and Wegkamp 2006; Grandvalet et al. 2009; Yuan and Wegkamp 2010; El-Yaniv and Wiener 2010), multicategory classification (Zhang, Wang, and Qiao 2018), multi-label classification (Pillai, Fumera, and Roli 2013) and confidence set learning (Wang and Qiao 2018). Though we also expect to model the possibility of abstaining from a choice, our work adopts a completely different setting from this line of research. First of all, the reject option in classification problems models the ambiguity of whether a given object belongs to a given class; whereas the indistinguishable state in our paper models the ambiguity of whether a pair of two objects have a significant difference concerning a given personalized attribute. Secondly, under the context of classification, the goal for including a reject option is to improve the robustness of the classifier and the reject option itself does not have a clear semantic meaning; whereas our goal in this paper is to recognize the indistinguishable comparisons explicitly, and the indistinguishable option itself has a clear semantic meaning showing that the given pair has similar presence of a given attribute.

**Partial Ranking.** Finally, we review the related work on partial ranking, namely, the methods for learning to rank with partial orders where not all the instances are distinguishable. Recently, worth mentioning is the work on a specific type of partial orders, namely linear orders of unsorted or tied subsets (partitions, bucket orders) (Gionis et al. 2006; Lebanon and Mao 2008). However, the problems addressed in these studies are different from our goals. As a representative work, (Cheng et al. 2010) starts to consider the partial ranking problem from the learning perspective. The idea is that it produces predictions in the form of partial order by thresholding a (valued) pairwise preference relation, *i.e.*, by an "$\alpha$-cut" of preference relation. However, it lacks a solid principle to decide the hyper-parameter $\alpha$ as the threshold. Moreover, it needs to know in advance the preference relation between every pair of items. To learn the threshold automatically, (Xu et al. 2018) proposes an extended probabilistic model for partial order ranking which could solve these problems in (Cheng et al. 2010). However, these methods do not have predictive power for new PA comers. (Yu and Grauman 2015) addresses this problem from a "just noticeable differences (JND)" perspective, together with a limited representation of low-level image features, to decide whether a difference in personalized attributes is perceptible. Different from these studies, in this paper, we propose a deep probability model which not only offers the first attempt to integrate partial rankings for PA, but also exhibits strong prediction power for the choices of new PA alternatives.

## Methodology

In this section, we first clarify the problem definition for our model. Specifically, it includes the definition of our training dataset, the elaboration of our goal in this paper, and the input and output of our proposed model. Then we formulate our proposed model, which includes a probabilistic ranking model, a deep representation framework, a maximum likelihood-based loss function and a label prediction scheme.

### Problem Definition

We aim to learn a PA partial ranking aggregation model, where each comparison corresponds to a local ranking between a pair of images with respect to the given PA. We begin our discussion with a brief introduction of the traditional pairwise comparison training data. First, our training data contains $n$ object images to be compared. We then choose $N$ pairs from the pool of the training object images to form the pairwise comparisons. For a given pair $(i, j)$ including two images $i$ and $j$, we denote the corresponding raw input as $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Furthermore, we invite $U$ annotators from the crowdsourcing platforms to label the pairs. Mathematically, the annotation results could be represented as a multi-graph. We define the graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. $\mathcal{V}$ is the set of vertexes which contains all the distinct image items occurred in the comparisons. $\mathcal{E}$ is the set of comparison edges. Each time the comparison pair $(i, j)$ is labeled by a new user, we add an edge $(i, j)$ to the set $\mathcal{E}$. Since multiple users take part in the annotation process, it is natural to observe multi-edges between two vertexes. Traditionally, the pairwise comparison training data only provides two options for the annotation. Now we could denote the labeling results as a function $y : \mathcal{E} \to \{-1, 1\}$. For a given PA $\mathcal{A}$, and a given user $u$, if the user thinks that $\mathcal{A}$ has a stronger presence in $i$, then the pair is labeled as $y_{ij}^u = 1$. Equivalently we also denote this as a relation: $i \overset{u}{\succ} j$. If the opposite is the case, the user then labels the pair as $y_{ij}^u = -1$, and we denote this as $j \overset{u}{\succ} i$. So far we have clarified the setting of traditional pairwise annotation for PA. We see that the traditional setting assumes that any pair $(i, j)$ in the dataset must be distinguishable, in a way that either $i \overset{u}{\succ} j$ or $j \overset{u}{\succ} i$ holds. However, indistinguishable pairs are ubiquitous in real-world problems. Taking the data in Fig.2.1 as an example, here the PA in question is *smiling*, and we have five object images $\{V_1, V_2, \cdots, V_5\}$ and three annotators. Among the five images, we find that $V_2$ and $V_3$ are hardly distinguishable. In fact, this phenomenon is well-justified by the limitation of human perception. According to psychology studies, it is impossible for human beings to notice arbitrary small difference. Instead, there is a minimum level of stimulation, known as Just Noticeable Difference (JND) (Stern and Johnson 2010), such that only when the difference between two objects is higher than JND could it be noticeable at least half the time. This motivates us to include an extra relation beyond $\succ$ and $\prec$. Specifically, when the user $u$ could not differentiate $i$ and $j$ with respect to the given PA $\mathcal{A}$ and would like to abstain from the current choice, we provide an alternative option as $y_{ij}^u = 0$, which could be equivalently expressed as $i \overset{u}{\approx} j$.

With the third state considered, we come to a novel labeling function $y : \mathcal{E} \to \{-1, 0, 1\}$. Then the corresponding

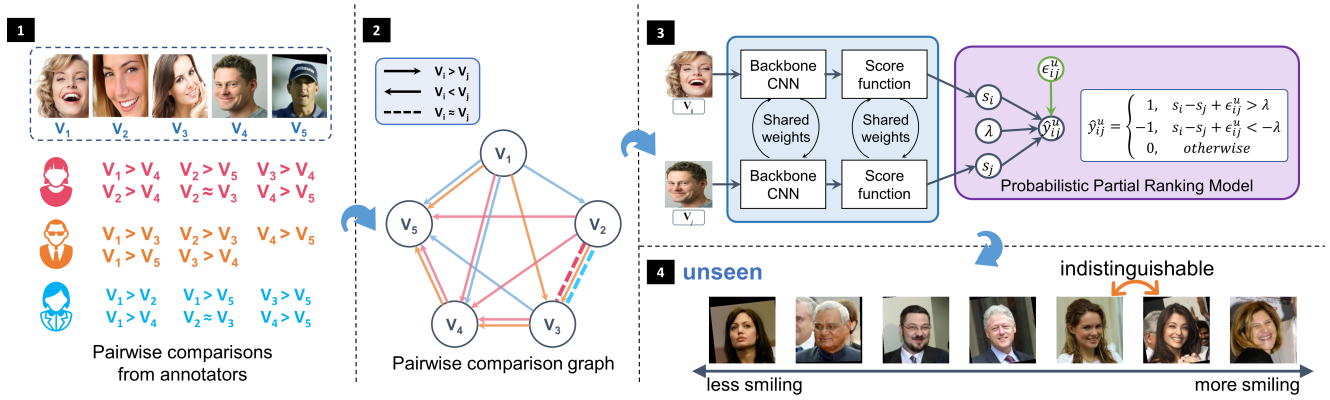


Figure 2: Overview of our approach. (1) This shows an example of the training dataset for a PA smiling. The upper half shows the training set images and the lower half shows the annotations collected via crowdsourcing platforms. (2) This is an instantiation of the multi-graph formed by the annotations in (1). Here, arrows stand for distinguishable annotations and dotted lines stand for indistinguishable/abstention results. (3) In the training phase, we provide an end-to-end deep prediction model to aggregate personalized partial ranks with the presence of abstention. (4) In the testing phase, the proposed model is expected to predict a consensus local ranking score between unseen images. Moreover, the model is expected to recognize indistinguishable/indistinguishable pairs.

$y_{ij}^u$ becomes:

$$\begin{cases} y_{ij}^u = 1, & i \overset{u}{\succ} j, \quad (i,j) \in \mathcal{D}_u; \\ y_{ij}^u = -1, & j \overset{u}{\succ} i, \quad (i,j) \in \mathcal{D}_u; \\ y_{ij}^u = 0, & j \overset{u}{\approx} i, \quad (i,j) \in \mathcal{D}_u. \end{cases} \tag{1}$$

where $\mathcal{D}_u$ is the set of all pairs labeled by user $u$. We present an instantiation of the annotations in Fig.2.1-Fig.2.2. The upper half of this figure shows the training image items and the lower half shows the annotations $y_{ij}^u$ collected from a set of three users. Correspondingly, the edge-labeled graph is shown in Fig.2.2. The edges standing for distinguishable results are labeled as arrows which always point from the weaker nodes to the stronger nodes. The abstention annotations are labeled as dotted lines. Moreover, different colors stand for different annotators.

Now we are ready to elaborate our goal in this paper. *Given the object images and the comparison annotations $\{y_{ij}^u\}$ in the presence of abstention labels, our goal is then to construct an end-to-end deep learning model which is able to (a) aggregate personalized partial rankings for PAs into consensus results based on the training data, and to (b) further apply the learned model to predict the such consensus ranking results for unknown image pairs.*

Moreover, the input and the output of our proposed model are then defined as the following.

**Input.** The input of our deep model is the multi-graph $\mathcal{G}$ mentioned previously, the annotations $\{y_{ij}^u\}$ and the image items, where each time a specific edge $(i,j)$ and a specific annotation $y_{ij}^u$ is fed to the network.

**Output.** Our model will output the relative score $s_i$ and $s_j$ for the input and a threshold $\lambda$ which is necessary for judging the abstention state.

*Note that, in the rest of the paper, whenever the $y_{ij}^u$ occurs again, it refers to the new labeling process expressed in Eq.(1).*

## A Deep PA Partial Ranking Aggregation Model

Now we propose a probabilistic partial ranking model capturing the generative process of the annotations $y_{ij}^u$. We assume that each training object image has an aggregated consensus preference score toward the underlying PA, where a higher score indicates a stronger presence of the PA, and the corresponding score list of the training images is $\boldsymbol{s} = [s_1, \cdots, s_n]$. Accordingly, when $i$ and $j$ form an edge in $\mathcal{G}$, we expect to observe a consensus score difference $s_i - s_j$. For a given user $u$, due to his/her personalized comprehension toward the attribute, he/she will provide a score difference of $s_i - s_j + \epsilon_{ij}^u$, where $\epsilon_{ij}^u \sim \mathcal{P}$ is a random variable indicating the personalized deviation from the consensus. As mentioned in the previous subsection, due to the limitation of human perception, we could not notice arbitrary small difference. The difference becomes noticeable only when its magnitude is more significant than a threshold $\lambda$. Then, for a specific user $u$, and a specific observation $(i,j)$, we assume that $y_{ij}^u$ is produced by comparing the personalized score difference $s_i - s_j + \epsilon_{ij}^u$ with the threshold $\lambda$. Moreover, we assume that a distinguishable result is claimed when the absolute value of the difference $|s_i - s_j + \epsilon_{ij}^u|$ is greater than $\lambda$, otherwise $u$ will observe indistinguishable result. In other words, in our model, user $u$ would choose $y_{ij}^u = 1$, if the observed score difference $s_i - s_j + \epsilon_{ij}^u$ is greater than the threshold $\lambda$. To the opposite, if $s_i - s_j + \epsilon_{ij}^u$ is smaller than $-\lambda$, then user $u$ would choose $y_{ij}^u = -1$. If none of them is the case, and $s_i - s_j + \epsilon_{ij}^u$ has a smaller magnitude than $\lambda$, the user would claim that $i$ and $j$ are not distinguishable. Above all, $y_{ij}^u$ is obtained from the following rule:

$$y_{ij}^u = \begin{cases} 1, & s_i - s_j + \epsilon_{ij}^u > \lambda; \\ -1, & s_i - s_j + \epsilon_{ij}^u < -\lambda; \\ 0, & \text{else.} \end{cases} \qquad \epsilon_{ij}^u \sim \mathcal{P}. \tag{2}$$

According to Eq.(2), we could predict the annotation $y_{ij}^u$ once we know $s_i$ and $s_j$. However, in real-world problems,

we do not know the score list $\boldsymbol{s}$ in advance. In this sense, we turn to provide an estimation of the consensus scores from the raw images $\boldsymbol{x}$. Specifically, as shown in Fig.2.3, we employ a deep Siamese (Chopra, Hadsell, and LeCun 2005; Norouzi, Fleet, and Salakhutdinov 2012; Wang et al. 2014) convolutional neural network to estimate the scores $\boldsymbol{s}$ and the relative difference $s_i - s_j$ for the image pairs. In our model, the input is an edge $(i, j)$ in the graph $\mathcal{G}$. Following the convention of the Siamese convolutional neural network, the weights in the network are shared across the two branches. Each branch of the network is fed with one image of the pair. Given this architecture, to obtain high-level representations of the image, the raw inputs are first fed to a convolution backbone architecture with weights $\Theta_b$. Then to estimate the score for $i$ and $j$, the outputs of the backbone are fed to a scoring function with weights $\Theta_s$. Denote $\Theta = \{\Theta_b, \Theta_s\}$, and denote the estimated score for $i$ and $j$ as $s(\boldsymbol{x}_i, \Theta)$ and $s(\boldsymbol{x}_j, \Theta)$ respectively, then we have:

$$s(\boldsymbol{x}_i, \Theta) = Score(Backbone(\boldsymbol{x}_i, \Theta_b), \Theta_s), \tag{3}$$
$$s(\boldsymbol{x}_j, \Theta) = Score(Backbone(\boldsymbol{x}_j, \Theta_b), \Theta_s). \tag{4}$$

Together with the probabilistic model for annotations in Eq.(2) and the formulation of estimated scores in Eq.(3)-(4), now we turn to construct a loss function to learn a suitable estimation of the scores such that the learned scores $\{s(\boldsymbol{x}, \Theta)\}$ match the annotations $\{y_{ij}^u\}$ as much as possible. Specifically, we adopt the Maximum Likelihood Estimation (MLE) framework. According to the principle of MLE, the learned estimation $s(\boldsymbol{x}, \Theta)$ should maximize the likelihood to observe the annotations in the training set. To derive the likelihood function, let us first derive the possibility to observe a given annotation $y_{ij}^u$. We define two auxiliary variables $\Delta_{ij}^+$ and $\Delta_{ij}^-$ as :

$$\begin{aligned} \Delta_{ij}^+ &= \ \lambda - s(\boldsymbol{x}_i, \Theta) + s(\boldsymbol{x}_j, \Theta), \\ \Delta_{ij}^- &= -\lambda - s(\boldsymbol{x}_i, \Theta) + s(\boldsymbol{x}_j, \Theta). \end{aligned} \tag{5}$$

Recall Eq.(2), $\epsilon_{ij}^u$ subjects to a distribution $\mathcal{P}$. Now we assume that the Cumulative Distribution Function (CDF) of $\mathcal{P}$ is $F(\cdot)$ such that $F(x) = P\left\{\epsilon_{ij}^u \leq x; \Theta\right\}$, where $P\left\{\mathcal{B}; \Theta\right\}$ is the possibility to observe the event $\mathcal{B}$ parameterized by $\Theta$. Practically, we assume that $\epsilon_{ij}^u$ subjects to a logistic distribution with a CDF

$$F(x) = \frac{1}{1 + \exp(-x)}.$$

Now we could derive the probability to observe $y_{ij}^u = 1, 0, -1$, respectively. According to Eq.(2) and Eq.(5), we have:

$$\begin{aligned} P\{y_{ij}^u = \ \ 1; \Theta, \lambda\} &= P\{\epsilon_{ij}^u > \Delta_{ij}^+; \Theta, \lambda\} \\ &= 1 - F\big(\Delta_{ij}^+\big); \\ P\{y_{ij}^u = \ \ 0; \Theta, \lambda\} &= P\{\Delta_{ij}^- < \epsilon_{ij}^u \leq \Delta_{ij}^+; \Theta, \lambda\} \\ &= F(\Delta_{ij}^+) - F(\Delta_{ij}^-); \\ P\{y_{ij}^u = -1; \Theta, \lambda\} &= P\{\epsilon_{ij}^u \leq \Delta_{ij}^-; \Theta, \lambda\} \\ &= F(\Delta_{ij}^-). \end{aligned}$$

Then we could estimate the possibility to observe the annotation $y_{ij}^u$ as:

$$P\left\{y_{ij}^u; \Theta, \lambda\right\} = \prod_{q \in \{-1,0,1\}} P\left\{y_{ij}^u = q; \Theta, \lambda\right\}^{\left[y_{ij}^u = q\right]},$$

where $[\mathcal{B}] = 1$ if event $\mathcal{B}$ happens, otherwise $[\mathcal{B}] = 0$. By simply taking a negative logarithm transformation over $P\left\{y_{ij}^u; \Theta\right\}$, we come to the negative log-likelihood function for a given annotation:

$$\begin{aligned} &- log\big(P\left\{y_{ij}^u; \Theta, \lambda\right\}\big) \\ &= \sum_{q \in \{-1,0,1\}} - \left[y_{ij}^u = q\right] \log(P\left\{y_{ij}^u = q; \Theta, \lambda\right\}). \end{aligned}$$

From a global view, we denote $P\left\{\mathcal{Y}; \Theta, \lambda\right\}$ as the possibility to simultaneously observe all the training annotations, where $\mathcal{Y} = \{y_{ij}^u\}_{(u,i,j)}$ is the set for all the personalized annotations in the training data.

$$P\left\{\mathcal{Y}; \Theta, \lambda\right\} = \prod_u \prod_{(i,j) \in \mathcal{D}_u} P\left\{y_{ij}^u; \Theta, \lambda\right\}$$

Then we reach the negative log-likelihood function for the whole training set:

$$\begin{aligned} \mathcal{L}(\Theta, \lambda) &= -\log\left(P\left\{\mathcal{Y}; \Theta, \lambda\right\}\right) \\ &= -\sum_u \sum_{(i,j) \in \mathcal{D}_u} \log\left(P\left\{y_{ij}^u; \Theta, \lambda\right\}\right). \end{aligned}$$

Since the negative logarithm function is strictly decreasing, maximizing the likelihood is equivalent to minimizing $\mathcal{L}(\Theta, \lambda)$. This means that we could train the network through the following optimization problem:

$$\min_{\Theta, \lambda} \mathcal{L}(\Theta, \lambda).$$

At the end of the training phase, we obtain a network with the learned parameter $\Theta, \lambda$, as well as the aggregated score list $\boldsymbol{s}$ with the personalized effect eliminated.

As shown in Fig.2.4, during the test phase, our model will predict the consensus partial ranking labels for unseen images without the help of crowdsourcing annotators. More precisely, given the test pair $(k, m)$, we expect to predict the consensus label $y_{km}$. If $k \succ m$, $y_{km} = 1$; if $k \prec m$, $y_{km} = -1$; and if $k \approx m$, $y_{km} = 0$. Taking the objects in Fig.2.4 as examples, here our PA is smiling. If $k$ is a more smiling person, $m$ is a less smiling person, and the difference is significant, then we come to a label $y_{km} = 1$. If the opposite is the case, then we come to a label $y_{km} = -1$. Otherwise, if the difference between two persons is not significant, then we come to a label $y_{km} = 0$. To predict the $y_{km}$ from the raw images, we feed $\boldsymbol{x}_k$ and $\boldsymbol{x}_m$ to the trained model, and obtain the predicted scores $s(\boldsymbol{x}_k, \Theta)$ and $s(\boldsymbol{x}_m, \Theta)$. Since the personalized deviation in the annotation process is modeled by $\epsilon_{ij}^u$, the predicted score difference $s(\boldsymbol{x}_k, \Theta) - s(\boldsymbol{x}_m, \Theta)$ could be regarded as a reasonable estimation of the consensus score difference with the noise removed. In this way, we predict the consensus label $y_{km}$ with $\hat{y}_{km}$ with the following formulation:

$$\hat{y}_{km} = \begin{cases} 1, & s(\boldsymbol{x}_k, \Theta) - s(\boldsymbol{x}_m, \Theta) > \ \lambda; \\ -1, & s(\boldsymbol{x}_k, \Theta) - s(\boldsymbol{x}_m, \Theta) < -\lambda; \\ 0, & \text{else.} \end{cases} \tag{6}$$

## Discussion

Under mild assumptions, we show that the proposed decision rule Eq.(6) could potentially provide a consistent result with the consensus comparison order. Given a finite object image set $\mathcal{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n\}$, assume that there is a consensus comparison relation $\prec$, such that $y \prec x$ if and only if image $x$ has a stronger presence of a given PA than image $y$. We formulate the indistinguishable relation $\approx$ as $\neg(x \prec y) \wedge \neg(y \prec x)$. Furthermore, we assume that $\prec$ forms a semi-order (and thus a partial order) in the sense that (Luce 1956):

- For all $x, y \in \mathcal{X}$, it is not possible for both $x \prec y$ and $y \prec x$ to be true.
- For all $x, y, z, w \in \mathcal{X}$, if it is true that $x \prec y, y \approx z$, and $z \prec w$, then it must also be true that $x \prec w$.
- For all $x, y, z, w \in \mathcal{X}$, if it is true that $x \prec y, y \prec z, y \approx w$, then it cannot also be true that $x \approx w$ and $z \approx w$ simultaneously.

Note that semi-order is a weak assumption in our problem, since all the three constraints should be naturally satisfied by a comparison relation for PA. With the above-mentioned assumptions, we have the following proposition.

**Proposition 1.** *Given any finite $\mathcal{X}$ and a semi-order $\prec$ on $\mathcal{X}$. Furthermore, define $x \approx y$ as $\neg(x \prec y) \wedge \neg(y \prec x)$. There exists a real-valued function $s^\star(x)$ on $\mathcal{X}$ with range $[0, 1]$, and a $\lambda^\star > 0$, such that:*

$$\begin{cases} k \prec m, & s^\star(\boldsymbol{x}_m) - s^\star(\boldsymbol{x}_k) > \lambda^\star; \\ m \prec k, & s^\star(\boldsymbol{x}_m) - s^\star(\boldsymbol{x}_k) < -\lambda^\star; \\ m \approx k, & \textit{else}. \end{cases}$$

*Proof.* According to Thm.3 in (Fishburn 1970), there exists a real-valued function $u(\cdot)$ on $\mathcal{X}$ such that:

$$y \prec x \text{ if and only if } u(y) + 1 < u(x). \tag{7}$$

Let $f(x)$ be a strict monotone function such that $f\big(u(x)\big) \in [0, 1]$, $\forall x \in \mathcal{X}$ and that:

$$f\big(u(x)\big) < f\big(u(y)\big), \text{ if } u(x) < u(y), \ \forall x, y \in \mathcal{X}.$$

Then by choosing $s^\star = f \circ u$ and

$$\lambda^\star = \min_{x,y \in \mathcal{X}, u(x) > u(y)+1} f\big(u(x)\big) - f\big(u(y)\big)$$

$\forall x, y \in \mathcal{S}$, we have:

$$\begin{aligned} y \prec x &\Longleftrightarrow u(y) + 1 < u(x) \\ &\Longleftrightarrow f\big(u(y)\big) + \lambda^\star < f\big(u(x)\big). \\ x \prec y &\Longleftrightarrow u(x) + 1 < u(y) \\ &\Longleftrightarrow f\big(u(x)\big) + \lambda^\star < f(u(y)). \\ x \approx y &\Longleftrightarrow |f\big(u(y)\big) - f\big(u(x)\big)| \leq \lambda^\star. \end{aligned}$$

Then we reach Eq.(7). □

Prop.1 shows that we can find a reasonable estimation of $\prec$ with Eq.(6), if $s(\cdot, \Theta), \lambda$ in the network could give a good approximation of $s^\star, \lambda^\star$. Fortunately, the most recent studies on the universality of deep neural networks (Zhou 2020) tend to support the approximation performance of deep neural networks. This suggests that we could obtain a reasonable performance from the proposed method.

| Dataset | No.Pairs | No.Images | No.Classes |
|---|---|---|---|
| LFW-10 Dataset | 50,000 | 2000 | 10 |
| Shoes Dataset | 61,879 | 14,658 | 6 |
| Sun Dataset | 45,694 | 14,000 | 5 |

Table 1: Dataset summary.

# Experiments

In this section, experiments are exhibited on three benchmark datasets (see Tab.1) to illustrate the validity of the analysis above and applications of the methodology proposed.

## Datasets

**LFW-10**. The LFW-10 dataset (Sandeep, Verma, and Jawahar 2014) consists of 2,000 face images, which are chosen from the Labeled Faces in the Wild (Huang et al. 2008) dataset. More specifically, it includes 10 personalized attributes, like smiling, big eyes, etc. Each pair was labeled by 5 people. As our goal is to predict PA from labels with ties, we do not conduct any pre-processing steps like majority voting to merge these labels. The images are split to 1000/1000 to create training/testing pairs. The resulting dataset has 50,000 annotated sample pairs, with 500 training and testing pairs per attribute. Specifically, pairs labeled as "0" account for 41.09% of the total pairs.

**Shoes**. The Shoes dataset is collected from (Kovashka and Grauman 2015) which contains 14,658 online shopping images. For each attribute, there are at least 190 users who take part in the annotation procedure, and each user is assigned with 50 images. Note that this dataset uses instance-wise feedbacks (each query only involves an evaluation for one object) rather than pairwise feedbacks. We then adopt a sampling strategy to produce pairwise feedback data. Specifically, we randomly sample positive annotations and negatives annotations from each user's records to form the pairs we need. We randomly select 2000 distinct pairs for each attribute, where each pair contains a positive instance and a negative instance. Whereas we sample 30% of indistinguishable pairs for each attribute, where each pair contains only positive instances or negative instances. Finally, this yields to a volume of 61,879 pairwise annotations for our dataset.

**Sun**. The SUN Attribute dataset is a well-known large-scale scene attribute dataset including roughly 14,000 images and a taxonomy of 102 discriminative attributes. Recently, the personalized annotations over five attributes are collected with hundreds of annotators. For each person, 50 images are labeled based on their own comprehension and preference. Overall, this dataset contains 64,900 annotations collected from different users. Again, the Sun data only collected instance-wise feedbacks. Here we use the same sampling strategy as Shoes dataset to generate pairwise comparison results. As a result, we obtain a volume of 45,694 pairwise annotations for our Sun dataset.

## Competitors

To show the effectiveness, we compare our method with 11 competitors, which fall into four categories:

| Types | Backbone | Algorithm | Bald | D.Hai | B.Eye | GLook | Masc. | Mouth | Smile | Teeth | Foreh. | Young | Aver. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Shallow | - | LinearR | .2907 | .3642 | .2301 | .3186 | .2718 | .3153 | .3535 | .2947 | .3476 | .4178 | .3204 |
| | | LogisticR | .3657 | .4257 | .2481 | .3631 | .3180 | .3480 | .3413 | .3456 | .3555 | .4826 | .3594 |
| | | RankNet | .3695 | .4189 | .2553 | .3729 | .3162 | .3518 | .3530 | .3487 | .3568 | .4822 | .3625 |
| | | RankSVM | .3356 | .4135 | .2427 | .3737 | .3020 | .3355 | .3627 | .3333 | .3709 | .4781 | .3548 |
| Tree | - | RankBoost | .3669 | .4303 | .2400 | .3619 | .3100 | .3204 | .3551 | .3307 | .3289 | .4538 | .3498 |
| | | GBDT | .3627 | .4181 | .2404 | .3517 | .2958 | .3299 | .3673 | .3377 | .3568 | .4599 | .3520 |
| | | DART | .3648 | .4253 | .2436 | .3487 | .2993 | .3346 | .3509 | .3281 | .3380 | .4640 | .3497 |
| Prob. | - | JND-NonPar | .3668 | .3756 | .3924 | .3860 | .4072 | .3448 | .3680 | .3584 | .3868 | .3688 | .3755 |
| | | Ex-Prob | .4004 | .3712 | .4152 | .3652 | .4592 | .3280 | .3444 | .3624 | .3836 | .3788 | .3808 |
| Deep | AlexNet | ranking@0.5 | .4248 | .3588 | .5904 | .3716 | .5420 | .3988 | .3472 | .4612 | .3564 | .2020 | .4053 |
| | | CE@0.5 | .4192 | .3648 | .5812 | .3716 | .5340 | .4004 | .3712 | .4508 | .3716 | .2448 | .4110 |
| | | Ours-MLE | **.4572** | **.4460** | **.5712** | **.4396** | **.6252** | **.4288** | **.4400** | **.4824** | **.4352** | **.3700** | **.4696** |
| | VGG-16 | ranking@0.5 | .5240 | .4300 | .7540 | .4460 | .6760 | .4760 | .3940 | .4180 | .5620 | .2200 | .4900 |
| | | CE@0.5 | .5020 | .4520 | .7820 | .4640 | .6760 | .4760 | .4160 | .4080 | .5700 | .1660 | .4912 |
| | | Ours-MLE | **.5240** | **.5040** | **.7720** | **.4420** | **.7020** | **.5280** | **.4700** | **.4900** | **.5780** | **.3320** | **.5342** |
| | ResNet-50 | ranking@0.5 | .5200 | .4400 | .7820 | .4400 | .7020 | .4760 | .3980 | .4140 | .5740 | .1660 | .4912 |
| | | CE@0.5 | .4500 | .4200 | .6960 | .4360 | .6480 | .4740 | .3960 | .4180 | .5280 | .2400 | .4706 |
| | | Ours-MLE | **.5300** | **.5160** | **.7640** | **.4420** | **.7400** | **.5180** | **.4320** | **.4800** | **.5560** | **.3920** | **.5370** |

Table 2: Experimental results (ACC) of 10 attributes on LFW-10 dataset.

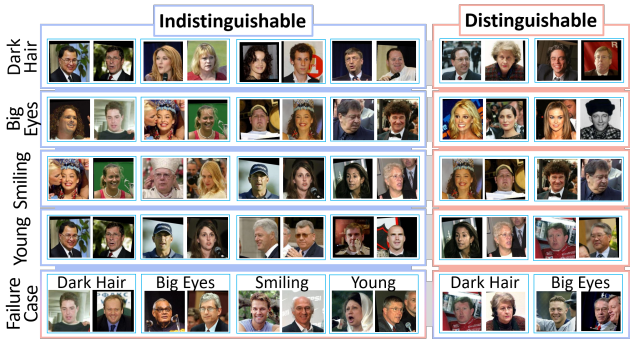


Figure 3: Example prediction results on LFW-10 dataset. Each row shows pairs for a particular attribute. The top four rows illustrates success cases. Left panel: pairs our proposed method correctly predicted as indistinguishable; Right panel: pairs correctly predicted as distinguishable by our method. The bottom row illustrates failure cases by our method; *i.e.*, the bottom left pair is indistinguishable for DarkHair, but we predict it distinguishable.

**Traditional and Shallow Models:**

- **LinearR**: uses least squares problem for learning to rank.
- **LogisticR**: uses logistic regression for learning.
- **RankSVM** (Joachims 2002): With the modification of input features by $\boldsymbol{x}_{ij} = (\boldsymbol{x}_i - \boldsymbol{x}_j)$, RankSVM turns the learning to rank problem to a standard SVM with input $\{\boldsymbol{x}_{ij}, y_{ij}\}_{(i,j)}$. It is used to show the superiority of our fine-grained model.
- **RankNet** (Burges et al. 2005): To show the effectiveness of using a deeper network, we compare our method with the classical RankNet model, where a traditional three layer structure is used instead of a deeper architecture.

**Tree-based Ensemble Models:**

- **RankBoost** (Freund et al. 2003): Besides the deep learning framework, it is also known that the ensemble-based methods could also serve a model for hierarchical learning and representation. In this sense, we compare our method with the RankBoost model, one of the most classical tree-based ensemble methods.
- **GBDT** (Friedman 2001): Gradient Boosting Decision Tree (GBDT) extends the idea of boosting, which generates a weak learner in each iteration by learning the recursive residual. It has gained surprising improvements in many traditional tasks and competitions.
- **DART** (Rashmi and Gilad-Bachrach 2015): Recently, the well-known dropout trick has also been applied to tree-based learning, be it the dart method. We also record its performance to show the superiority of our method.

**Probabilistic Models:**

- **JND-NonPar** (Yu and Grauman 2015): To show the power of our proposed scheme, we compare our method with JND-NonPar, which provides an indistinguishable pair recognition scheme with a stage-wise non-parametric probabilistic model.
- **Ex-Prob** (Xu et al. 2018): We also compare our method with Ex-Prob, which adopts extended probabilistic models for partial ranking.

**Deep Models:**

- **ranking@0.5**: The end-to-end baseline model with AlexNet/VGG16/ResNet50 as backbone architecture, ranking loss function and a fixed threshold 0.5.
- **CE@0.5**: The end-to-end baseline model with AlexNet/VGG16/ResNet50 as backbone architecture, cross-entropy loss function and a fixed threshold 0.5.

| Types | Backbone | Algorithm | Comf. | Fash. | Form. | Poi. | Bro. | Orn. | Aver. |
|---|---|---|---|---|---|---|---|---|---|
| Shallow | - | LinearR | .4400 | .3598 | .3226 | .5183 | .3665 | .4080 | .4025 |
| | | LogisticR | .4400 | .3476 | .3097 | .4817 | .3478 | .4080 | .3891 |
| | | RankNet | .4457 | .3476 | .3161 | .4634 | .3292 | .4138 | .3860 |
| | | RankSVM | .4229 | .3598 | .2968 | .5610 | .3292 | .3621 | .3886 |
| Tree | - | RankBoost | .4286 | .3476 | .2903 | .4573 | .3168 | .4080 | .3748 |
| | | GBDT | .4171 | .3354 | .2903 | .3841 | .3230 | .4023 | .3587 |
| | | DART | .4057 | .3537 | .3032 | .4329 | .2981 | .3966 | .3650 |
| Prob. | - | JND-NonPar | .5314 | .6098 | .6258 | .5732 | .5901 | .5805 | .5851 |
| | | Ex-Prob | .6343 | .6159 | .6581 | .5122 | .6273 | .6207 | .6114 |
| Deep | AlexNet | ranking@0.5 | .7273 | .7137 | .8305 | .4646 | .8731 | .6841 | .7155 |
| | | CE@0.5 | .6930 | .6483 | .7969 | .4154 | .8093 | .6334 | .6660 |
| | | Ours-MLE | **.8271** | **.7251** | **.8088** | **.4985** | **.8834** | **.7332** | **.7460** |
| | VGG-16 | ranking@0.5 | .7303 | .7384 | .8006 | .4923 | .8313 | .6825 | .7126 |
| | | CE@0.5 | .6855 | .6607 | .7565 | .5200 | .8512 | .6268 | .6835 |
| | | Ours-MLE | **.8174** | **.7109** | **.7797** | **.4554** | **.8292** | **.7797** | **.7287** |
| | ResNet-50 | ranking@0.5 | .7213 | .7213 | .7976 | .5385 | .8745 | .7091 | .7271 |
| | | CE@0.5 | .6654 | .6948 | .7267 | .5015 | .8374 | .6259 | .6753 |
| | | Ours-MLE | **.8279** | **.7014** | **.8245** | **.4769** | **.8779** | **.7656** | **.7457** |

Table 3: Experimental results (ACC) on Shoes dataset.


Figure 4: Example predictions on Shoes dataset.

## Implementation Details

- **LFW-10**. Since the first three types of competitors adopt non-deep models, we employ a stage-wise training strategy to improve their performance for the sake of fairness. More precisely, we first extract the pre-trained features from AlexNet (Krizhevsky, Sutskever, and Hinton 2012) and then feed them to the competitors. For the deep learning methods, we implement the models using library Pytorch (Paszke et al. 2019), and train the network jointly for all PAs. Moreover, AlexNet/VGG16/ResNet50 are used as the backbones and the weights are initialized with pre-trained features on ImageNet (Deng et al. 2009). For training, we use a mini-batch size of 128 image pairs for SGD. We set the initial learning rate to 1e-3 and fix the momentum to 0.9. We train these networks for 300 epochs, and the learning rate is reduced by a factor of 10 every 40 epochs. We use random crops of size 227x227 from our 256x256 input image during training and resize all images to 227x227 for testing.
- **Shoes**. The implementation follows the same settings with LFW-10 dataset.
- **Sun**. It follows the same settings with LFW-10 dataset, except that the pre-trained features are initialized with models pretrained on CelebA (Liu et al. 2015).

## Comparative Results

**LFW-10**. Tab.2 reports the test accuracy (ACC) for each attribute. We see that our method (marked with **bold**) consistently outperforms all the benchmark algorithms by a significant margin. This validates the effectiveness of our method. In particular, it can be observed that: (1) The performance of end-to-end deep methods are better than all non-deep methods, which suggests the strong representation power of end-to-end neural networks in PA prediction tasks. (2) For end-to-end models, ranking loss and cross-entropy loss show comparable results on this dataset. (3) Moreover, since our model learns $\lambda$ automatically and adaptively, it enjoys a significant improvement with respect to a fixed threshold 0.5. In addition, Fig.3 shows qualitative prediction examples returned by AlexNet, while other two backbones exhibit similar results. Here we see the subtleties of confusing pairs. In the success cases, for the left panel of image pairs, our proposed method can predict them as indistinguishable, while previous methods were usually forced to make a binary comparison. Meanwhile, those that are distinguishable (right panel) may have only subtle differences. A number of failure cases are also shown. Some of them are caused by unique view points (*e.g.*, for 'dark hair' attribute, the man has sparse scalp, so it is hard to tell who has darker hair); others are caused by the unsatisfactory feature representation,

| Types | Algorithm | Rust. | Clut. | Mod. | Open. | Soot. | Aver. |
|---|---|---|---|---|---|---|---|
| Shallow | LinearR | .3000 | .4943 | .4337 | .3832 | .4615 | .4146 |
| | LogisticR | .2933 | .4406 | .3855 | .3892 | .4725 | .3962 |
| | RankNet | .2933 | .4713 | .3916 | .3713 | .4835 | .4022 |
| | RankSVM | .2600 | .6130 | .3373 | .3293 | .4396 | .3959 |
| Tree | RankBoost | .2867 | .4291 | .3614 | .3832 | .4725 | .3866 |
| | GBDT | .2667 | .3870 | .3675 | .3832 | .4689 | .3746 |
| | DART | .2733 | .4138 | .3735 | .3533 | .4469 | .3722 |
| Prob. | JND-NonPar | .5733 | .5172 | .4759 | .4551 | .5311 | .5105 |
| | Ex-Prob | .6400 | .3870 | .6205 | .5569 | .4103 | .5229 |
| Deep-AlexNet | ranking@0.5 | .8061 | .3903 | .7809 | .7718 | .3980 | .6294 |
| | CE@0.5 | .7947 | .4085 | .8013 | .7780 | .3980 | .6361 |
| | Ours-MLE | **.8688** | **.3722** | **.8613** | **.8045** | **.4260** | **.6665** |
| Deep-VGG-16 | ranking@0.5 | .8254 | .3783 | .7563 | .7430 | .5196 | .6445 |
| | CE@0.5 | .7845 | .3722 | .7874 | .7290 | .4008 | .6148 |
| | Ours-MLE | **.9031** | **.3763** | **.8736** | **.8419** | **.4008** | **.6791** |
| Deep-ResNet-50 | ranking@0.5 | .8356 | .4085 | .7606 | .7438 | .4623 | .6422 |
| | CE@0.5 | .8092 | .3883 | .7649 | .6846 | .4134 | .6121 |
| | Ours-MLE | **.8826** | **.3883** | **.8661** | **.7804** | **.4553** | **.6745** |

Table 4: Experimental results (ACC) on Sun dataset.

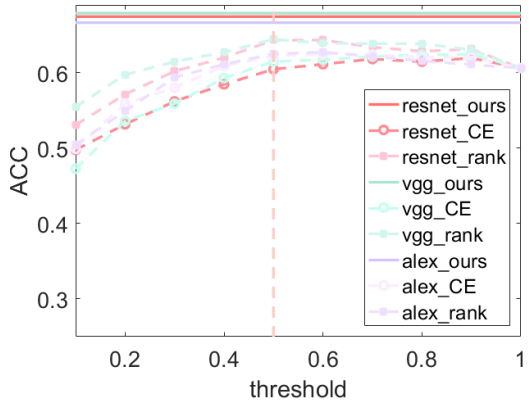


Figure 5: ACC vs. Threshold on Sun Dataset.

*e.g.*, in 'young' attribute, as 'young' would be a function of multiple subtle visual cues like face shape, skin texture, hair color, whereas something like baldness or smiling has a better visual focus captured well by part-based features.

**Shoes**. Similar to the LFW-10 datasets, Tab.3 again shows that the performance of our proposed end-to-end model is significantly better than that of other competitors. Moreover, some prediction examples computed by our method are illustrated in Fig.4. In the top six rows with successful detection examples, we see how our method can correctly predict various instances that are indistinguishable, even though the raw images can be quite diverse (*e.g.*, a sports shoe and a flat leisure shoe are equally pointy). Similarly, it can detect a difference even when the image pair is fairly similar (*e.g.*, a high boot and high-heeled dance shoe are distinguishable for brown even though the colors are close). The failure cases are mostly caused by ambiguity: both images have this attribute with similar degree. This thus corresponds to a truly ambiguous case which can go either way.


Figure 6: Example predictions on Sun dataset.

**Sun**. Tab.4 again shows that the performance of our proposed model significantly outperforms other competitors on this dataset. Moreover, just like the other two datasets, with the threshold being fixed as $\lambda = 0.5$, we could observe significant performance degradations. This suggests that introducing an adaptive threshold is necessary for recognizing the indistinguishable pairs. To see the performance of other thresholds, Fig.5 shows the ACC against threshold ranging from 0.1 to 1, from which we could observe that setting the threshold as 0.5 tends to induce a better performance for the competitors. We thus only show the results of threshold@0.5 in the competitive experiments above. Besides, we illustrate some of the prediction results on the dataset in Fig.6. From this figure, we see that, in most cases, our method could successively recognize the indistinguishable pairs, and could provide a correct ranking result when the underlying pair is distinguishable, even when the images being compared have completely different backgrounds (say the examples for Cluttered and Modern).

## Conclusion

With the help of online crowdsourcing platforms, this work explores a challenging problem, namely, how to correctly learn aggregated pairwise PA ranking results from personalized opinions, when some of the pairs suffer from an intrinsically imperceptible difference. We propose an end-to-end deep partial ranking model with a multi-graph formulation of the annotation data, a deep feature learning module, and a probabilistic partial rank aggregation model which takes into consideration the limitation of human perceptions. Specifically, an adaptive threshold $\lambda$ is parameterized together with the ranking scores. In this model, indistinguishable pairs could be automatically detected when the absolute value of the score difference is below the learned threshold $\lambda$. Putting all these together, we obtain an end-to-end deep learning framework based on an MLE-induced loss function. In our empirical studies, we perform a series of experiments on three real-world datasets: LFW-10, Shoes, and Sun. The corresponding results show the effectiveness and superiority of our proposed model.

## Acknowledgments

This work was supported in part by the National Key R&D Program of China under (Grant No. 2018AAA0102104), in part by National Natural Science Foundation of China (61931008, 61620106009, U1636214, 61971016, 61836002, 61672514, and 61976202), in part by Key Research Program of Frontier Sciences, CAS: QYZDJ-SSW-SYS013, in part by Beijing Natural Science Foundation (No. 4182079), in part by Youth Innovation Promotion Association CAS, and in part by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No. XDB28000000.

## References

Bampis, C. G.; Li, Z.; Katsavounidis, I.; and Bovik, A. C. 2018. Recurrent and Dynamic Models for Predicting Streaming Video Quality of Experience. *IEEE Transactions on Image Processing* 27(7): 3316–3331.

Burges, C.; Shaked, T.; Renshaw, E.; Lazier, A.; Deeds, M.; Hamilton, N.; and Hullender, G. 2005. Learning to rank using gradient descent. In *International Conference on Machine Learning*, 89–96.

Chen, K.-T.; Wu, C.-C.; Chang, Y.-C.; and Lei, C.-L. 2009. A crowdsourceable QoE evaluation framework for multimedia content. In *ACM International Conference on Multimedia*, 491–500.

Cheng, W.; Rademaker, M.; De Baets, B.; and Hüllermeier, E. 2010. Predicting Partial Orders: Ranking with Abstention. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 215–230.

Chopra, S.; Hadsell, R.; and LeCun, Y. 2005. Learning a similarity metric discriminatively, with application to face verification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 539–546.

Chow, C. 1970. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory* 16(1): 41–46.

Deng, J.; Dong, W.; Socher, R.; Li, L.; Kai Li; and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

El-Yaniv, R.; and Wiener, Y. 2010. On the foundations of noise-free selective classification. *Journal of Machine Learning Research* 11(5): 1605–1641.

Fishburn, P. C. 1970. Intransitive indifference with unequal indifference intervals. *Journal of Mathematical Psychology* 7(1): 144–149.

Freund, Y.; Iyer, R.; Schapire, R. E.; and Singer, Y. 2003. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research* 4(6): 933–969.

Friedman, J. H. 2001. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics* 29(5): 1189–1232.

Fu, Y.; Hospedales, T. M.; Xiang, T.; Gong, S.; and Yao, Y. 2014. Interestingness prediction by robust learning to rank. In *European Conference on Computer Vision*, 488–503.

Fu, Y.; Hospedales, T. M.; Xiang, T.; Xiong, J.; Gong, S.; Wang, Y.; and Yao, Y. 2016. Robust subjective visual property prediction from crowdsourced pairwise labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38(3): 563–577.

Gionis, A.; Mannila, H.; Puolamäki, K.; and Ukkonen, A. 2006. Algorithms for discovering bucket orders from data. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 561–566.

Grandvalet, Y.; Rakotomamonjy, A.; Keshet, J.; and Canu, S. 2009. Support vector machines with a reject option. In *Advances in Neural Information Processing Systems*, 537–544.

Herbei, R.; and Wegkamp, M. H. 2006. Classification with reject option. *Canadian Journal of Statistics* 34(4): 709–721.

Huang, G. B.; Mattar, M.; Berg, T.; and Learned-Miller, E. 2008. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on faces in 'Real-Life' Images: detection, alignment, and recognition*.

Jing, P.; Su, Y.; Nie, L.; and Gu, H. 2017. Predicting image memorability through adaptive transfer learning from external sources. *IEEE Transactions on Multimedia* 19(5): 1050–1062.

Joachims, T. 2002. Optimizing search engines using clickthrough data. In *ACM International Conference on Knowledge Discovery and Data Mining*, 133–142.

Kovashka, A.; and Grauman, K. 2015. Discovering attribute shades of meaning with the crowd. *International Journal of Computer Vision* 114(1): 56–73.

Kovashka, A.; and Grauman, K. 2017. Attributes for Image Retrieval. In *Visual Attributes*, 89–117. Springer.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 1097–1105.

Lebanon, G.; and Mao, Y. 2008. Non-parametric modeling of partially ranked data. *Journal of Machine Learning Research* 9(5): 2401–2429.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *IEEE International Conference on Computer Vision*, 3730–3738.

Luce, R. D. 1956. Semiorders and a theory of utility discrimination. *Econometrica, Journal of the Econometric Society* 178–191.

Norouzi, M.; Fleet, D. J.; and Salakhutdinov, R. R. 2012. Hamming distance metric learning. In *Advances in Neural Information Processing Systems*, 1061–1069.

Parikh, D.; and Grauman, K. 2011. Relative attributes. In *IEEE International Conference on Computer Vision*, 503–510.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, 8024–8035.

Pillai, I.; Fumera, G.; and Roli, F. 2013. Multi-label classification with a reject option. *Pattern Recognition* 46(8): 2256–2266.

Rashmi, K. V.; and Gilad-Bachrach, R. 2015. DART: Dropouts meet Multiple Additive Regression Trees. In *International Conference on Artificial Intelligence and Statistics*, 436–443.

Sandeep, R. N.; Verma, Y.; and Jawahar, C. 2014. Relative parts: Distinctive parts for learning relative attributes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3614–3621.

Singh, K. K.; and Lee, Y. J. 2016. End-to-End Localization and Ranking for Relative Attributes. In *European Conference on Computer Vision*, 753–769.

Squalli-Houssaini, H.; Duong, N. Q.; Gwenaëlle, M.; and Demarty, C.-H. 2018. Deep learning for predicting image memorability. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2371–2375.

Stern, M. K.; and Johnson, J. H. 2010. Just noticeable difference. *The Corsini Encyclopedia of Psychology* .

Wang, J.; Song, Y.; Leung, T.; Rosenberg, C.; Wang, J.; Philbin, J.; Chen, B.; and Wu, Y. 2014. Learning Fine-grained Image Similarity with Deep Ranking. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1386–1393.

Wang, W.; and Qiao, X. 2018. Learning Confidence Sets using Support Vector Machines. In *Advances in Neural Information Processing Systems*, 4934–4943.

Xu, Q.; Jiang, T.; Yao, Y.; Huang, Q.; Yan, B.; and Lin, W. 2011. Random partial paired comparison for subjective video quality assessment via HodgeRank. In *ACM International Conference on Multimedia*, 393–402.

Xu, Q.; Xiong, J.; Sun, X.; Yang, Z.; Cao, X.; Huang, Q.; and Yao, Y. 2018. A Margin-based MLE for Crowdsourced Partial Ranking. In *ACM International Conference on Multimedia*, 591–599.

Yu, A.; and Grauman, K. 2014. Fine-grained visual comparisons with local learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 192–199.

Yu, A.; and Grauman, K. 2015. Just noticeable differences in visual attributes. In *IEEE International Conference on Computer Vision*, 2416–2424.

Yuan, M.; and Wegkamp, M. 2010. Classification methods with reject option based on convex risk minimization. *Journal of Machine Learning Research* 11(1): 111–130.

Zhang, C.; Wang, W.; and Qiao, X. 2018. On Reject and Refine Options in Multicategory Classification. *Journal of the American Statistical Association* 113(522): 730–745.

Zhou, D.-X. 2020. Universality of deep convolutional neural networks. *Applied and computational harmonic analysis* 48(2): 787–794.