# Label Ranking by Directly Optimizing Performance Measures

**Qishen Wang  Ou Wu[*]  Ying Chen  Weiming Hu**
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
{qswang, wuou, ychen, wmhu}@nlpr.ia.ac.cn

## Abstract

Label ranking aims to map instances to an order over a predefined set of labels. It is ideal that the label ranking model is trained by directly maximizing performance measures on training data. However, existing studies on label ranking models mainly based on the minimization of classification errors or rank losses. To fill in this gap in label ranking, in this paper a novel label ranking model is learned by minimizing a loss function directly defined on the performance measures. The proposed algorithm, referred to as BoostLR, employs a boosting framework and utilizes the rank aggregation technique to construct weak label rankers. Experimental results reveal the initial success of BoostLR.

## Introduction

Different from learning to rank(Wu, Hu, and Gao 2011), the goal of label ranking is to assign an order over a set of predefined labels according to the nature of the input (Dekel Manning and Singer 2003). Take document categorization as an example, assuming that there are several categorical labels such as 'sports', 'education' and 'entertainment'. The relationship '$i \succ j$' represents label $i$ ranks higher than label $j$. Label ranking will map a document to an ordered list of labels, e.g., 'sports' $\succ$ 'entertainment' $\succ$ 'education'. Label ranking has been applied to social emotions predictions(Wang et al. 2011), and other applications.

Previous algorithms focus on minimizing classical rank losses to learn a label ranking model. While in many scenarios, the results of label ranking are evaluated by performance measures such as normalized discounted cumulative gain (NDCG) and mean average precision (MAP). Compared with classical rank losses, the performance measures can reflect the practical values of ranking results with respect to the application domains better. For example, in some applications only the labels ordered in top-$k$ are concerned. NDCG is just a measure that can focus more on top-$k$ results, while rank losses can not. Indeed, classical rank losses are also proven to be loosely related to some performance measures. Ideally, the label ranking model is learned so that the accuracy in terms of one of the performance measures is maximum. However, to our knowledge, there has been no study on label ranking by directly optimizing performance measures. In this paper, a novel algorithm, named BoostLR, is proposed by directly optimize performance measures.

## Methodologies

### Framework of the Algorithm

Label ranking aims to learn a mapping function $f$: $\mathcal{X} \rightarrow \mathcal{Y}$, where $f$ is chosen from a hypothesis class $\mathcal{F}$ such that a loss function $l$ :$\mathcal{F} \times \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ on training data is minimized, $\mathcal{X}$ is the instance space, and $\mathcal{Y}$ is the output space of all possible full ranking over a set of labels $\Gamma = \{1, 2, \cdots, K\}$. Under the boosting framework, $f$ is a strong label ranker defined as a linear combination of weak label rankers: $f(x_i) = \sum_{t=1}^{T} \alpha_t h_t(x_i)$, where $T$ is the sum of weak label rankers, $h_t$ is the optimal weak label ranker in $t$-th round, and $\alpha_t$ is its associated weight. In ranking, the learned function $f$ assigns a score to each label in $\Gamma$ for a given instance, and then the labels are ordered according to their corresponding scores.

The constructing process of $f$ in BoostLR is similar with that in AdaBoost. As for the weak label rankers, it is generated by directly optimizing performance measures. The details is stated as following subsection.

### Creation of Weak Label Rankers

We adopt the idea that each weak label ranker only deals with one feature dimension. In the $t$-th round a weak ranker can be written as

$$h_t\left(x(d)\right) = x\left(d\right) \cdot \mathbf{w}_t \tag{1}$$

where $\mathbf{w}_t$=$((\mathbf{w}_t(1), \cdots, \mathbf{w}_t(K))^T \in \mathbb{R}^K)$ is a $K$-dimensional vector over $\Gamma$ and $d \in D$ is the corresponding feature dimension. For a particular dimension $d$, $\mathbf{w}_t$ can be obtained by optimizing:

$$\begin{aligned} \mathbf{w}_t &= \arg\min_{\mathbf{w}_t \in \mathbb{R}^K} R(h_t(x(d))) \\ &= \arg\min_{\mathbf{w}_t \in \mathbb{R}^K} \sum_{i=1}^{N} P_t(i)[1 - Ev(\pi(x_i(d) \cdot \mathbf{w}_t), y_i)] \end{aligned} \tag{2}$$

where $P_t(i)$ is the weight of $x_i$ in the $t$-th round, $y_i$ is the ground truth label ranking for $x_i$, and $Ev(\pi(v), y_i) \in [0, 1]$ is a performance measure function used to measure the consistency between $y_i$ and $\pi(v)$ defined a ranking derived from the entries of a vector $v$ from large to small. The higher the value of $Ev(\pi(v), y_i)$, the lower the loss $R$. In practice, Eq.

[*] indicates the equal contribution to the first author.

Table 1: The NDCG values of the competing algorithms on UCI data sets.

| Data set | NDCG-1 | | | | NDCG-2 | | | | NDCG-3 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | BoostLR | IB-PL | IB-Mal | Lin-LL | BoostLR | IB-PL | IB-Mal | Lin-LL | BoostLR | IB-PL | IB-Mal | Lin-LL |
| authorship | **.8044** | .7009 | .3671 | .5068 | .7879 | **.8661** | .4708 | .6130 | 0.8743 | **.8867** | .6174 | .7343 |
| bodyfat | **.3577** | .3129 | .3117 | .2323 | **.4825** | .4466 | .4268 | .3848 | **.5887** | .5697 | .5424 | .5192 |
| calhousing | **.5583** | .4702 | 0.4201 | .4814 | **.6279** | .5806 | 0.5409 | .5831 | **0.7223** | .6982 | 0.6574 | .6996 |
| cpu-small | **.4698** | .4198 | .3657 | .4246 | **.5789** | .5217 | .4522 | .5246 | **.6691** | .6182 | .5422 | .6202 |
| elevators | **.4739** | .3641 | .2161 | .2833 | **.5457** | .4864 | .3230 | .3623 | **.6718** | .5942 | .4258 | .4600 |
| fried | **.5100** | .4987 | .3640 | .3659 | **.6189** | .5778 | .4576 | .4598 | **.7021** | .6513 | .5500 | .5515 |
| housing | **.3840** | .3203 | .3326 | .3351 | **.5316** | .4453 | .4783 | .4687 | **.6774** | .6007 | .6263 | .6379 |
| stock | .5722 | **.5841** | .3571 | .3167 | .6096 | **.6545** | .4563 | .5148 | .7005 | **.7076** | .5448 | .5708 |
| vowel | **.3478** | .2519 | .1901 | .1936 | **.4938** | .3441 | .2725 | .2888 | **.5735** | .4130 | .3628 | .3474 |
| wine | **.9045** | .7520 | .4695 | .7472 | **.8367** | .7645 | .5685 | .7655 | **.9372** | .8898 | .7897 | .8890 |
| Avg. Rank | 1.375 | 2.438 | 3.189 | 3 | 1.5 | 2.375 | 3.188 | 2.938 | 1.5 | 2.375 | 3.25 | 2.875 |

(2) is intractable to directly optimized as the search space of $\mathbf{w}_t$ is $\mathbb{R}^K$ and the performance measures (e.g., NDCG) are usually non-continuous. We instead introduce a near-optimal search procedure as follows.

For any two parameter vectors $\mathbf{w}_t^{'}$ and $\mathbf{w}_t^{''}$, if $\pi(\mathbf{w}_t^{'}) = \pi(\mathbf{w}_t^{''})$, their corresponding weak learners' performances are equal. Hence, the search space of Eq. (2) is reduced to $K!$ as $Ev$ is only sensitive to $\pi(\mathbf{w}_t)$. Note that if $x_i(d) \geq 0, \pi(x_i(d) \cdot \mathbf{w}_t) = \pi(\mathbf{w}_t)$ and if $x_i(d) < 0, \pi(x_i(d) \cdot \mathbf{w}_t) = Inver(\pi(\mathbf{w}_t))$, where $Inver(\pi)$ is the ranking by reversing $\pi$. Then the Eq.(2) is equivalent to

$$\begin{aligned}\mathbf{w}_t = \arg\max_{\mathbf{w_t} \in \mathbb{R}^K} \{ & \sum_{i,x_i(d) \geq 0} P_t(i) Ev[\pi(\mathbf{w}_t), y_i] \\ + & \sum_{j,x_j(d)<0} P_t(j) Ev[Inver(\pi(\mathbf{w}_t)), y_j]\}\end{aligned} \tag{3}$$

When $K$ is small, $\mathbf{w}_t$ can be achieved by a simple strategy that calculates all the $K!$ possible rankings in $\mathbb{R}^K$. However, when $K$ is large, this simple strategy is inapplicable. If $Ev$ is taken as a measure of the *consistency* between two rankings, Eq. (3) can be approximately transformed into:

$$\begin{aligned}\mathbf{w}_t = \arg\max_{\mathbf{w_t} \in \mathbb{R}^K} & \sum_{i,x_i(d) \geq 0} P_t(i) Ev[\pi(\mathbf{w}_t), y_i] \\ & + \sum_{j,x_i(d)<0} P(j) Ev[\pi(\mathbf{w}_t), Inver(y_j)]\end{aligned} \tag{4}$$

Let

$$\overline{y}_{(i)} = \begin{cases} y_i & \text{if} \quad x_i(d) \geq 0 \\ Inver(y_i) & \text{if} \quad x_i(d) < 0 \end{cases} \tag{5}$$

Then Eq.(4) equals to

$$\mathbf{w}_t = \arg\max_{\mathbf{w_t} \in \mathbb{R}^K} \sum_{i=1}^{N} P_t(i) Ev(\pi(\mathbf{w}_t), \overline{y}_i) \tag{6}$$

Eq.(6) aims to find an optimal ranking ($\pi(\mathbf{w}_t)$) that is consistent with all the $N$ rankings ($y_i$) as much as possible. This is a typical rank aggregation problem. The weighted linear combination method (Lee 1997) is considered here.

**Algorithm** 1 shows the process of creating weak label ranker. It is noteworthy that more than one weak ranker would likely be generated in the algorithm. In this case, the $h_t$ that makes the performance of current $f_t$ maximize will be chosen as the optimal weak ranker.

**Algorithm 1** Creating $h_t$ in the $t$-th round

```
Input: Samples $S = \{(x_1, y_1), \cdots, (x_N, y_N)\}$, $P_t$;
Output: Weak label ranker $h_t$.
Initialize $\mathbf{w}^{(d)} = 0, d = 1, \cdots, D$;
for (int $d = 1$; $i \leq D$; $d++$) do
    for (int $i = 1$; $i \leq N$; $i++$) do
        Calculate $\overline{y}_i$ using Eq. (5)
        $\mathbf{w}^{(d)} = \mathbf{w}^{(d)} + P_t(i)[\mathbf{1}_{K \times 1} - \overline{y}_i]$
    end for
    Normalizing $\mathbf{w}^{(1)}, \cdots, \mathbf{w}^{(D)}$ to $[0, 1]$;
    Calculate $E_t^{(d)} = \sum_{i=1}^{N} P_t(i) Ev(\pi[x_i(d)\mathbf{w}^{(d)}], y_i)$
end for
Find $d$ such that $E_t^{(d)}$ is the maximum and return $d$ and $\mathbf{w}_t$.
```

## Experiments

Sixteen UCI data sets compiled by (Cheng, Dembczynski and Hüllermeier 2010) are used in our experiments. Three existing state-of-the-art algorithms used in the experiments are: IB-PL proposed by (Cheng, Dembczynski and Hüllermeier 2010), IB-Mal proposed by (Cheng and Hüllermeier 2009), and Lin-LL proposed by (Dekel Manning and Singer 2003). As the initial experiments, NDCG is used to construct $Ev$ and measure the performances of the competing algorithms. In the calculation of NDCG for a particular data set, if the number of labels is smaller than six, NDCG@1, NDCG@2, and NDCG@3 are calculated; otherwise, NDCG@1, NDCG@3, and NDCG@5 are calculated. For convenience, the three NDCG values are called NDCG-1, NDCG-2, and NDCG-3. Each algorithm is performed 5-fold cross validation and the average results are reported.

As the limited space, only the results on partial UCI data sets are shown in Table 1. A two-step procedure recommended in (Demsar 2006) is used to compare the performance between each pair of algorithms. The comparison is based on the average ranks. At a level of 5%, BoostLR outperforms all the other competing algorithms.

## Conclusions

This paper has proposed a novel algorithm called BoostLR for label ranking. BoostLR can directly optimize the performance measures instead of optimizing either the pairwise ranking errors or rank losses that existing studies focus on. Initial experiments show that BoostLR outperforms several state-of-the-art label ranking algorithms.

## Acknowledgements

This work is partly supported by the NSFC (No. 61003115 and 60935002), the National 863 High-Tech $R\&D$ Program of China (No. 2012AA012504), the Natural Science Foundation of Beijing (No. 4121003), and Guangdong Natural Science Foundation (No. S2012020011081).

# References

Cheng, W.; Dembczynski, K.; and Hüllermeier, E. 2010. Label Ranking Methods based on the Plackett-Luce Model. In Proc. Int. Conf. on Machine learning (ICML): 215-222.

Dekel, O.; Manning, C. D.; and Singer, Y. 2003. Log-linear models for label ranking. In: Advances in Neural Information Processing Systems (NIPS) 16.

Wu, O.; Hu, W. M.; and Gao, J. 2011. Learning to Rank under Multiple Annotators. In Proc. International Joint Conference on Artificial Intelligence, 1571-1576.

Cheng, W., and Hüllermeier, E. 2009. A New Instance-Based Label Ranking Approach Using the Mallows Model. International Symposium on Neural Networks (1): 707 -716.

Demsar, J. 2006. Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research (JMLR), 7:1-30.

Wang, Q. S.; Wu, O.; Hu, W. M.; Li, W. Q.; and Yang, J. F. 2011. Ranking Social Emotions by Learning Listwise Preference. Asian Conference on Pattern Recognition, 164-168.

Lee, J. H. 1997. Analyses of multiple evidence combination. In Proc. ACM SIGIR Conference on Research and Development in Information Retrieval Conference, 267-276.