



ELSEVIER

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Group-enhanced ranking

Yuan Lin^a, Hongfei Lin^a, Kan Xu^a, Ajith Abraham^{b,c}, Hongbo Liu^{a,b,c,*}^a School of Computer, Dalian University of Technology, Dalian 116023, China^b Machine Intelligence Research Labs, Scientific Network for Innovation and Research Excellence, Auburn, WA 98071, USA^c School of Information, Dalian Maritime University, Dalian 116026, China

ARTICLE INFO

Article history:

Received 11 October 2013

Received in revised form

30 December 2013

Accepted 24 March 2014

Available online 2 October 2014

Keywords:

Information retrieval

Learning to rank

Groups

Loss functions

ABSTRACT

An essential issue in document retrieval is ranking, which is used to rank documents by their relevancies to a given query. This paper presents a novel machine learning framework for ranking based on document groups. Multiple level labels represent the relevance of documents. The values of labels are used to quantify the relevance of the documents. According to a given query in the training set, the documents are divided into several groups based upon their relevance labels. The group with higher relevance labels is always ranked upon the ones with lower relevance labels. Further a preference strategy is introduced in the loss functions, which are sensitive to the group with higher relevance labels to enhance the group ranking method. Experimental results illustrate that the proposed approach is very effective, with a 14 percent improvement on TD2003 dataset evaluated by MAP.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Ranking is the very important to design effective web search engines, since the ranking model directly influences the relevance of search results [1–3]. Many approaches were proposed to construct a model for ranking. There are the content-based approaches [4], such as the vector space model [5], BM25 [6] and the language model [7]. PageRank [8] and Hits [9] are famous link-based approaches. In the search environment, we usually have to confront a large amount of information. It becomes very difficult to tune the models with a great number of features [10,11]. Some new attempts are made by introducing machine learning methods to information retrieval to address the problem. Learning to rank [12–14] is an effective approach. However, its performance is dependent directly on the document samples and ranking loss function. In this paper, we present a novel group ranking framework, in which the loss is defined on the groups of documents with same relevant label. The documents with the same level label are categorized into one group, and the ranking task is reduced from ranking the multiple documents to ranking several groups. We further develop the loss functions by our preference strategy, which are sensitive to the group with higher relevance labels.

The rest of this paper is organized as follows. Related works are discussed in Section 2. In Section 3, we briefly discuss the loss functions for ranking. The proposed group ranking framework is presented in Section 4. Then we illustrate the experimental results and discussions in Section 5. Finally, we draw conclusions and point out the future works.

2. Related works

In learning to rank, there are mainly three methods: pointwise [15], pairwise [16] and listwise [17]. Pointwise samples single document using classification loss functions for ranking model [18]. Pairwise applies preference document pairs as training samples and also transform the ranking problem into classification [19]. Listwise defines its loss function to train the ranking model from dataset [20,21]. ListMLE and ListNet are two important kinds of listwise approaches. ListMLE [21] is a feature-based ranking algorithm that minimizes a probabilistic likelihood loss function. And its listwise samples are defined by the permutation probabilities in the Luce model [22]. ListNet [17] is a robust listwise approach based on cross entropy loss function. Listwise approach can achieve the better ranking accuracies than pairwise and pointwise approaches on most of datasets of Letor [23].

However, usually only top k positions of ranking play a key role in information retrieval [24]. Xia et al. [24] develop a top- k ranking framework through likelihood loss to improve the top- k ranking performance. The top- k ranking loss function is used to obtain the relevant documents on the top- k positions in the document list.

* Corresponding author at: Machine Intelligence Research Labs, Scientific Network for Innovation and Research Excellence, Auburn, WA 98071, USA.

E-mail addresses: yuanlin@dlut.edu.cn (Y. Lin), hflin@dlut.edu.cn (H. Lin), xukan@dlut.edu.cn (K. Xu), ajith.abraham@ieee.org (A. Abraham), lhb@dlut.edu.cn (H. Liu).

The number of the relevance documents is less than 10. The performance of top- k framework is determined by the number of relevant documents. It is ideal to set the value of k equal to the number of relevant documents [25]. But it is very difficult to do so [26]. Inspired by precious researches, we make an attempt to deal with it through group–group pair samples and preference strategy.

3. Loss functions for ranking

The ranking is optimized by minimizing a certain loss function using the training data. Likelihood and cross entropy functions are widely used in learning to rank. Here, we discuss only the basic ideas that are relevant to the present work.

3.1. Likelihood loss function

ListMLE [21] defines its probabilistic listwise loss function as follows:

$$L(f; x^q, y^q) = \sum_{s=1}^{n-1} \left(-f(x_{y_s^q}^q) + \ln \left(\sum_{i=s}^n \exp(f(x_{y_i^q}^q)) \right) \right) \quad (1)$$

where f is a ranking function, x^q is the document list to be ranked for query q , y^q is a randomly selected optimum permutation for query q , and n is the length of y^q . For any two documents x_i and x_j , x_i is ranked before x_j in y^q if $\text{label}(x_i) > \text{label}(x_j)$. y_i^q is denoted as the index of the object ranked at the i -th position in y^q . ListMLE is feasible to rank the documents in Letor dataset [23]. However, its ranking performance decreases as the scores of irrelevant documents increase. We will discuss this through some experimental results in Section 5.2.

3.2. Cross entropy loss for ranking

ListNet [17] introduces a probabilistic cross entropy loss function, as defined in the following equation:

$$L(f, y) = D(P(\pi|x; \psi_y) \| P(\pi|x; f(x))) \quad (2)$$

where D is cross entropy loss. $P(\pi|x; \psi_y)$ and $P(\pi|x; f(x))$ are Luce models based on permutation probabilities. The score vector of the ground truth permutation is produced by a mapping function $\psi_y(\cdot) : R_d \rightarrow R$, which is used to transform the order in a permutation, i.e., if $m > n$, then $\psi_y(m) > \psi_y(n)$. In order to optimize the top- k ranking accuracy, the mapping function only influences the order of documents within the top- k positions of the ground truth permutation. It also assigns a small value ϵ to all the remaining positions. The value is smaller than the score of any object ranked at the top- k positions, i.e. $\psi_y(x_{y_1}), \psi_y(x_{y_2}), \dots, \psi_y(x_{y_k}), \psi_y(x_{y_{k+1}}), \dots, \psi_y(x_{y_n})$, which compose a non-increasing sequence. So the loss becomes sensitive to the top- k subgroup order [24]. But all of the documents are considered as non-relevance after k position, which decreases the ranking performance.

4. Group ranking framework

Given a query in training set, the documents can be divided into several groups in which the documents with same labels are gathered together. A document pair is constructed by two groups, i.e., a group of documents with higher level label and a group of documents with lower level label. In this section, we introduce the group–group pair sampling, loss functions. Then our preference strategy is presented and the algorithm is summarized.

4.1. Group–group pair sample

Each query $q^{(i)}$ is associated with a list of documents $D^{(i)} = \{D_1^{(i)}, D_2^{(i)}, \dots, D_n^{(i)}\}$. $D_j^{(i)}$ denotes the group of documents with the same relevance judgement j . n is the number of relevance degree for the documents. Each list of documents $D^{(i)}$ is associated with a list of judgments (scores) $Y^{(i)} = \{Y_1^{(i)}, Y_2^{(i)}, \dots, Y_n^{(i)}\}$ where $Y_j^{(i)}$ denotes the judgment on the group document $D_j^{(i)}$ with respect to query $q^{(i)}$. For example, the relevance degree of $D_2^{(i)}$ is 2 when $j=2$. For the query $q^{(i)}$ with the relevance degree $n=3$ (0, 1, 2), the training sample is constructed as $D_g^{(i)} = \{D_{2,1}^{(i)}, D_{2,0}^{(i)}, D_{1,0}^{(i)}\}$. The group sample $D_{2,1}^{(i)}$ includes all the documents in the group $D_1^{(i)}$ and $D_2^{(i)}$. There are two types of label dataset used in this paper, such as OHSUMED with the relevance degree (0, 1, 2) and TD2003 with the relevance degree (0, 1).

4.2. Group ranking with loss functions

Different from the top- k ranking framework with listwise sample, our group ranking framework constructs samples by group pairs with different labels. The true loss of group ranking is defined as follows:

$$l_r(f(x), y) = \begin{cases} 0 & \text{if } \hat{y}_i = y_i \text{ where } \hat{y} = f(x); \\ 1 & \text{otherwise.} \end{cases} \quad (3)$$

where $i \in \{1, \dots, r\}$, and r is determined by the number of documents with the higher label in the group ranking samples. The expectation of group ranking loss can be re-written as follows:

$$L_g(f) = \int_{X \times Y} l_r(f(x), y) dP(x, y) \quad (4)$$

where X is the input space in while the elements are the group samples to be ranked, and Y is the output space in which the elements are permutations of groups. $P(x, y)$ is an unknown but fixed joint probability distribution of x and y . And the optimal ranking function with respect to the group ranking actual loss is

$$f(x) = \underset{G_r(j_1, j_2, \dots, j_r) \in G_r}{\text{argmax}} P(G_r(j_1, j_2, \dots, j_r) | x) \quad (5)$$

where $G_r(j_1, j_2, \dots, j_r)$ denotes a group sample in which all the permutations have the same top- r true loss, which is decided by the number of documents with higher relevance label. G_r denotes the collection of all top- r subgroups. In this paper, likelihood and cross entropy functions are adopted in our group–group samples.

4.2.1. Group ranking with likelihood loss

Considering likelihood loss function, the loss of the group sample is described as follows:

$$L_g(f; x^g, y^g) = \sum_{s=1}^r \left(-f(x_{y_s^g}^g) + \ln \left(\sum_{i=s}^n \exp(f(x_{y_i^g}^g)) \right) \right) \quad (6)$$

where x^g is a group sample, y^g is a ranked list of x^g , in which the documents with higher label are ranked upon the lower label group. In the group–group loss function, r is equal to the number of documents in the group with higher label. n is the length of optimum ranked list. As illustrated in Eq. (6), the loss becomes greater when increasing the scores of irrelevant documents obtained by ranking function. Our group rank framework ignores the increasing scores of the irrelevant documents in the ranked list for the likelihood loss, since the loss only depends on the increasing scores of relevant documents. The bigger scores of the relevant documents are, the smaller the loss is. We refer to the group method based on likelihood loss as GroupMLE.

4.2.2. Group ranking with cross entropy loss

Cross entropy can also be used to formulate the loss function for the group–group samples, which is described as follows:

$$L_g(\psi_g(y^g), z_g(f)) = - \sum_{g \in G} P_{\psi_g(y^g)}(g) \log(P_{z_g(f)}(g)) \quad (7)$$

where $z_g(f)$ is the score list for the group sample generated by ranking function f . y^g is the ground truth for the document list of the sample. G is the set of all possible permutations of the documents in the sample, $P_s(g)$ is the permutation probability of g according to s . For the group sample, the mapping function ψ_g is used to retain the value of the label of the group with the higher relevant label. But for the lower relevant label, the label will be set to a smaller value ϵ in order to make the loss function more sensitive to the documents with the higher relevant label in the new samples. We refer to the group method based on cross entropy loss as GroupCE.

4.2.3. Group ranking with preference loss

The group ranking can optimize the positions of relevant documents. However, for the labeled dataset with multiple relevant levels, the group samples with different preferences are usually considered as the equivalent ones in the training process. It may neglect the difference of original relevant labels for the group of documents. We introduce a relevance preference strategy, which lead to more significant improvements than original group ranking approaches.

For the group sample construction of binary label dataset, the importance of each sample is no differences. However, there are often multiple relevance judgments, such as 2 (definitely relevant), 1 (possibly relevant), 0 (irrelevant). Thus the group sample constructed by the documents with label 2 and documents with label 1 is indeed different from that constructed by the documents with label 2 and the documents with label 0. It may cause much more serious errors by mistaking the document with label 2 for label 0 than label 1. So it is imperfect that the group ranking framework considers all the samples as equivalent. In the case of learning to rank web documents, preference data are given in the form that one document is more relevant than another with respect to a given query. For each group sample, there is a preference for two group of documents, which reflects the relevance difference between the two labels. The preference is also useful for improving the group ranking loss function. The loss function for each sample is defined as follows:

$$L_g(f; x_i^g, y_i^g, p_i) = p_i L(f; x_i^g, y_i^g) \quad (8)$$

where x_i^g is the i -th group sample; p is the preference coefficient, which identifies the relevance of the documents into the learning process. p_i denotes the preference coefficient of i -th group sample, and is defined as follows:

$$p_i = \frac{\text{label}_h(x_i^g) - \text{label}_l(x_i^g)}{\sum_{j=1}^n (\text{label}_h(x_j^g) - \text{label}_l(x_j^g))} \quad (9)$$

where $\text{label}_h(x_i^g)$ and $\text{label}_l(x_i^g)$ are the two types of relevant labels in the group sample. In this way, the larger the difference between the two labels, the bigger the preference coefficient is. We introduce the relevant labels and preference into the loss function, which is used to improve the original group ranking loss function based on likelihood and cross entropy. We refer to the group methods improved by preference weighted loss as p-GroupMLE and p-GroupCE.

4.3. Group ranking algorithm

In our group ranking, the loss functions are presented in Eqs. (6)–(8), i.e. GroupMLE, GroupCE, p-GroupMLE and p-GroupCE, respectively. They are called as the ranking model f . The neural network and gradient descent are introduced to learn a ranking

model. The single layer linear neural network model is omitted the constant bias for simplicity [17]. The output nodes depend on the number of the features of dataset. The object function (4) is optimized by gradient descent. And the neural network weight W is updated iteratively. Once the group pairs are sampled from listwise samples, f is determined by neural network weight W , i.e. $f(X) = W \cdot X$. Group sample is transformed into a ranked list with respect to a given query. Our learning algorithm of group ranking is illustrated in Algorithm 1.

Algorithm 1. Group ranking algorithm.

Input:

Training data: a set of listwise samples
 $\{(X^q, Y^q) | (x_1^q, y_1^q), (x_2^q, y_2^q), \dots, (x_n^q, y_n^q)\}$;
 Test data: X_{test} .

Output:

Ranking list

- 1: Input $\{(X^q, Y^q) | (x_1^q, y_1^q), (x_2^q, y_2^q), \dots, (x_n^q, y_n^q)\}$;
- 2: Set the iteration number T , and learning rate η
- 3: Construct group samples from listwise samples
- 4: Obtain $\{(X^g, Y^g) | (x_1^g, y_1^g), (x_2^g, y_2^g), \dots, (x_n^g, y_n^g)\}$
- 5: Initialize neural network weight W
- 6: **for** $t=1$ to T **do**
- 7: **for** $i=1$ to n **do**
- 8: Compute gradient $\Delta W \leftarrow \frac{\partial L_g}{\partial W}$
- 9: Update $W \leftarrow W - \eta \cdot \Delta W$
- 10: **end for**
- 11: **end for**
- 12: Construct Neural Network model through W
- 13: Sort by $f(X_{test}) \leftarrow W \cdot X_{test}$
- 14: **return** Ranking list of X_{test}

5. Experimental results and discussions

5.1. Dataset

To evaluate the performance of our group ranking algorithm, Letor3.0 dataset provides various datasets with a larger file size. It is released by Microsoft Research Asia. Three representative instances were selected, i.e. OHSUMED, TD2003 and TD2004 in our experiments. The OHSUMED collection is derived from the MEDLINE dataset, which is popular in the information retrieval community. There are 106 queries in this collection and the total number of query–document pairs is 16 140. Each query–document pair is represented by 45-dimensional feature vector. The documents are manually labeled with absolute relevance judgments within the collection. There are three level labels: 2 (definitely relevant), 1 (possibly relevant) and 0 (irrelevant). The group samples based on these labels are three types: documents with label 2 and documents with label 1; documents with label 2 and documents with label 0; documents with label 1 and documents with label 0. For the one-group sample, the group with higher labels only selects one document; and for the group–group sample all the documents in the group with higher label are selected. The TD2003 collection is extracted from the topic distillation task of TREC2003. The goal of the topic distillation task is to find good websites about the query topic. There are 50 queries in this collection, and the total number of query–document pairs is 49 171. Each query–document pair is represented by 64-dimensional feature vector. There are two levels of relevance: 1 (relevant) and 0 (irrelevant). TD2004 is very similar to TD2003, which is extract from the dataset of the topic distillation task of TREC2004. It contains 75 queries and 74 170 documents with 64 features. Because there are two types of labels, there is only one type sample: documents with label 1 and documents with

label 0, for the group sample in .Gov collection. The dataset covers multiple classical features, such as term frequency, inverse document frequency, document length and their combinations [27,28]. And they are considered in many classical retrieval methods: BM25 [6], LMIR [7], Page Rank [8], etc. We adopt NDCG@N and MAP [29] to evaluate the ranking performance.

5.2. Listwise ranking

Suppose we have a hypothesis space with three hypothesis functions f_1 , f_2 and f_3 in ListMLE. There are six documents with respect to one query in Table 1 with their relevance labels. The indices of documents also construct optimum permutation, and f_1 , f_2 and f_3 are generated during iteration processes. We make a hypothesis that f_2 and f_3 are obtained after f_1 . According to Eq. (1), the loss of f_1 is calculated as follows:

$$-\ln\left(\frac{0.2}{0.2+0.3+0.1+0.1+0.2+0.1} \times \frac{0.3}{0.3+0.1+0.1+0.2+0.1} \times \frac{0.1}{0.1+0.1+0.2+0.1} \times \frac{0.1}{0.1+0.2+0.1} \times \frac{0.2}{0.2+0.1}\right).$$

The result is 5.9915. In the same way, it can be obtained that the loss of f_2 is 5.8579 and the loss of f_3 is 5.7991. The loss of ListMLE may be decreased. As matter of fact that the relevance labels of document 1 and 2 are equal, the descent loss is not necessary. While it is more serious problem caused by document 4 and 5, the loss is decreased by f_3 compared with f_2 . It is necessary to eliminate the influence of the scores of irrelevant documents.

As illustrated in Table 2, the top- k ranking performance of Top-10 ListMLE is better than that of ListMLE as evaluated by NDCG measures. Moreover, not only the top- k ranking performance is better, but also the ranking performance of the whole list is also better, as shown by MAP measure. The top- k ListNet is evaluated by NDCG@3. The results on Letor dataset are shown in Fig. 1. We tried different values of k (i.e., $k=1, 3, 10$, and the exact length of the ranked list: ListNet). As illustrated in Fig. 1, the top-3 ranking accuracies of ListNet are not significantly difference. Even if the performance is decreasing on TD2004. That may be caused by the selection of the value of k arbitrarily. The two documents with similar feature vectors and same label are annotated by different values based on the mapping functions.

5.3. Group ranking with likelihood loss

In this subsection, we evaluate the effectiveness of group ranking methods with likelihood loss function. As illustrated in Table 3, all the methods with modifying ListMLE algorithm significantly boost the ranking accuracies. * and # indicate significant improvement. The

Table 1
Toy example and models.

Doc no.	1	2	3	4	5	6
Label	1	1	1	0	0	0
$\exp(f_1(x))$	0.2	0.3	0.1	0.1	0.2	0.1
$\exp(f_2(x))$	0.3	0.2	0.1	0.1	0.2	0.1
$\exp(f_3(x))$	0.3	0.2	0.1	0.2	0.2	0.1

Table 2
Ranking accuracies on OHSUMED.

Methods	MAP	NDCG@1	NDCG@3	NDCG@10
ListMLE	0.4326	0.4196	0.4188	0.4190
Top-10 ListMLE	0.4441	0.5156	0.4772	0.4360

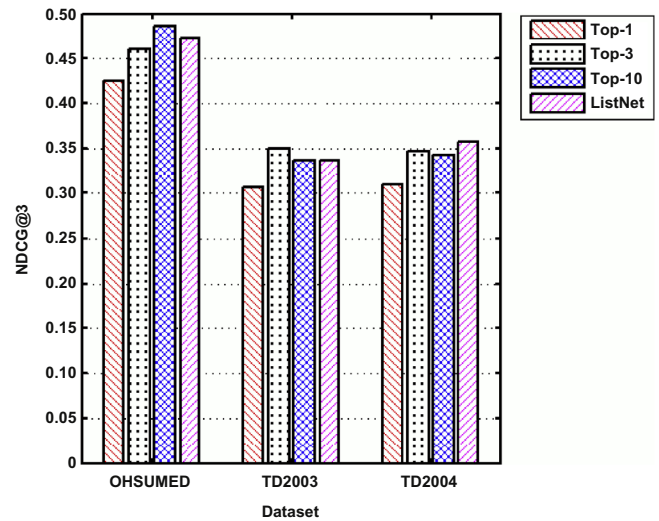


Fig. 1. Effectiveness of different values of k .

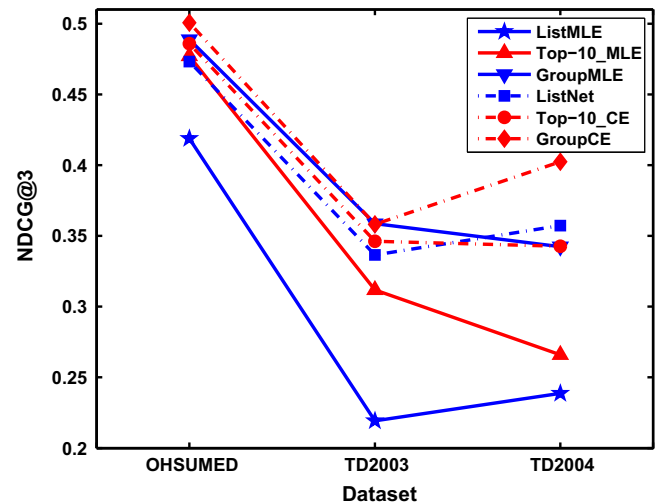


Fig. 2. Accuracies of considered methods.

Table 3
Accuracies of likelihood loss based methods.

Method	MAP	NDCG@1	NDCG@3	NDCG@10
ListMLE ^a	0.4326	0.4196	0.4188	0.4190
Top-10_MLE ^a	0.4441	0.5156	0.4772	0.4360
GroupMLE ^a	0.4517*	0.5589*#	0.4888*	0.4572*#
ListMLE ^b	0.1797	0.2222	0.2193	0.2362
Top-10_MLE ^b	0.2452	0.3022	0.3118	0.3077
GroupMLE ^b	0.2811*#	0.3267*#	0.3587*#	0.3568*#
ListMLE ^c	0.1600	0.2400	0.2387	0.2171
Top-10_MLE ^c	0.1774	0.2400	0.2659	0.2455
GroupMLE ^c	0.2386*#	0.4267*#	0.3424*#	0.3175*#

^a Evaluating on OHSUMED dataset.

^b Evaluating on TD2003 dataset.

^c Evaluating on TD2004 dataset.

empirical results illustrate that it can improve the ranking performance by adopting the true loss based on relevance documents.

In a group-group sample, the documents with higher label are considered as relevant documents and the ones with lower label as irrelevant documents. Compared with top- k ranking methods, group-group methods have two advantages. The true loss function is based only on the relevant documents in the group samples. The

group sample is easy to distinguish multiple labels in the multiple labeled collections. The results illustrate that the group-group methods almost outperform the baseline of Top-10_MLE and ListMLE method.

5.4. Group ranking with cross entropy loss

The proposed group ranking method considers the cross entropy as the loss function in this subsection. The learning error performances are shown in Figs. 3 and 4. We compared the proposed method with the existing methods based on cross entropy. From Table 4, it is evident that although Top-10 methods on the Letor datasets aim to improve the ListNet, in fact the improvement is marginal. Some evaluations show that Top-10 is no better than ListNet. It may be caused by the relevance label confusion as we discuss in Section 3.2. However, GroupCE achieves the best performance among all the methods on the OHSUMED and TD2004 datasets in terms of almost all of the measures. Fig. 2 illustrates the accuracies of the considered method on the three datasets. Our method also performs fairly well as compared to the other methods on the TD2003 dataset. The main reason is that the loss function method is relevant documents sensitive and also secures the label classification clarity. The group ranking framework can eliminate the label confusion, because it annotates the document in the same group by the same value.

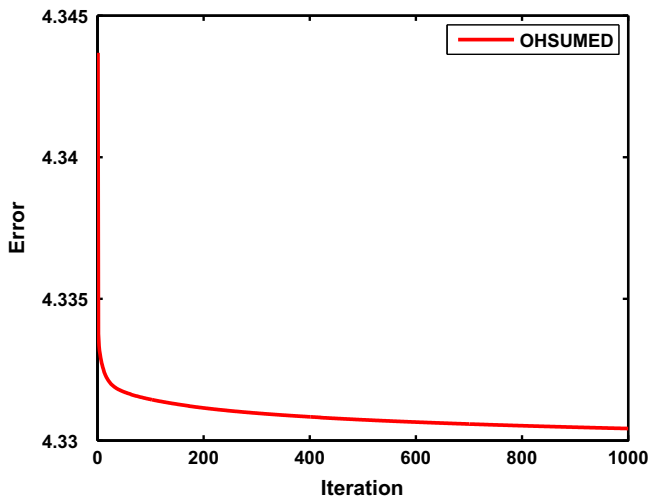


Fig. 3. The learning error process curve of GroupCE on OHSUMED.

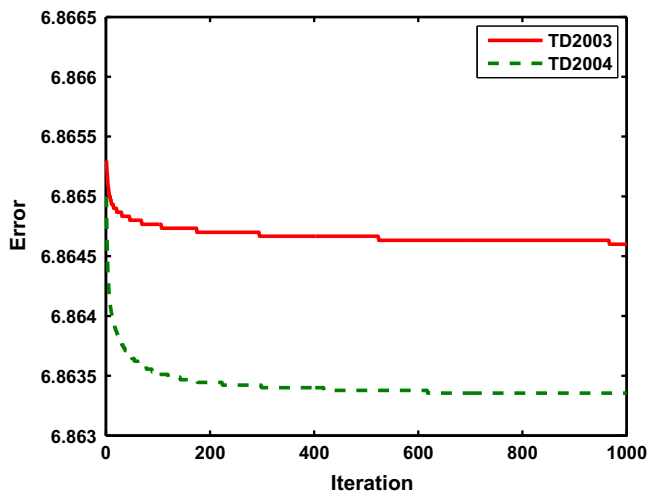


Fig. 4. The learning error curve of GroupCE on TD2003 and TD2004.

5.5. Group ranking with preference weighted loss

Varying from TD2003 and TD2004 datasets, there are three level labels: 2 (definitely relevance), 1 (possibly relevant) and 0 (irrelevant) in the OHSUMED dataset. According to Section 4.2.3, we use the preference group loss to improve the training process on OHSUMED dataset. The results are shown in Table 5. We can see that the preference group loss based approaches, i.e. p-GroupMLE, p-GroupCE, can significantly improve the ranking accuracies than the original versions of group ranking methods. p-GroupCE also achieves best performance among all the considered methods. The reason is that the loss rank model introduced the preference into each group samples, and the preference coefficient reflects the importance of the group sample effectively. It is obvious that our strategy improves the ranking performance.

We examine the top-k ranking performance of the proposed method when compared with the existing methods by NDCG@3. As illustrated in Fig. 5, GroupCE achieves the best performance

Table 4 Ranking accuracies of the considered methods with likelihood loss.

Method	MAP	NDCG@1	NDCG@3	NDCG@10
ListNet ^a	0.4457	0.5326	0.4732	0.4410
Top-10_CE ^a	0.4465	0.5126	0.4859	0.4411
GroupCE ^a	0.4564	0.5628 ^{##}	0.5007 ^{##}	0.4583 ^{##}
ListNet ^b	0.2753	0.4000	0.3365	0.3484
Top-10_CE ^b	0.2458	0.2667	0.3462	0.3371
GroupCE ^b	0.2818 [#]	0.3867 [#]	0.3582 ^{##}	0.3650 ^{##}
ListNet ^c	0.2231	0.3600	0.3573	0.3175
Top-10_CE ^c	0.2275	0.4400	0.3427	0.3157
GroupCE ^c	0.2485 ^{##}	0.4933 ^{##}	0.4024 ^{##}	0.3377 ^{##}

^a Evaluating on OHSUMED dataset.
^b Evaluating on TD2003 dataset.
^c Evaluating on TD2004 dataset.

Table 5 Group ranking accuracies on OHSUMED.

Methods	MAP	NDCG@1	NDCG@3	NDCG@10
GroupMLE	0.4517	0.5589	0.4888	0.4572
GroupCE	0.4564	0.5628	0.5007	0.4583
p-GroupMLE	0.4521	0.5621	0.4912	0.4583
p-GroupCE	0.4581	0.5711	0.4967	0.4768

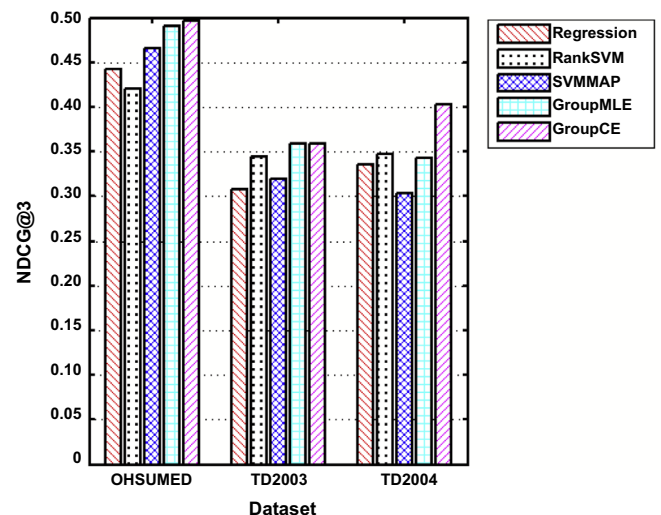


Fig. 5. Top-k performance of ranking methods.

comparing with existing methods, because the group ranking methods are sensitive to relevant documents.

6. Conclusions and future works

In this paper, we presented a group ranking framework through group pair samples and preference strategy. Before the documents are ranked, they were divided into several groups. Each label was organized together. The document pair was constructed by two groups, i.e. a group of documents with higher level label and a group of documents with lower level label. So the ranking task was reduced obviously. The group loss functions were defined by likelihood loss and cross entropy loss. And the preference coefficient enhanced the sensitiveness to relevant documents. Our group ranking approach was evaluated on different TREC collections. The experimental results illustrated that our group ranking approach, especially the one with cross entropy loss, could significantly outperform other considered ranking methods. For TD2003 and TD2003 collection, the group method obtained 12 percent improvement on average.

As future works, we plan to study them as follows:

- (1) It would be worthwhile to explore other loss functions to implement the group ranking methods.
- (2) We will extend the group ranking to some latest works about the top- k ranking [30].
- (3) For the effectiveness of group ranking method, we will also exploit the way to apply our method to the other research field such as the term-ranking task of query expansion.

Acknowledgments

The work described in this paper was supported by the grant from the Natural Science Foundation of China (No. 61173035, 61402075, 61472058), the Research Fund for the Doctoral Program of Higher Education (No. 20090041110002), the Program for New Century Excellent Talents in University (Grant no. NCET-11-0861), and the Fundamental Research Funds for the Central Universities.

References

- [1] H. Deng, I. King, M. Lyu, Enhanced models for expertise retrieval using community-aware strategies, *IEEE Trans. Syst. Man Cybern. Part B: Cybern.* 42 (1) (2012) 93–106.
- [2] S. Bashir, Estimating retrievability ranks of documents using document features, *Neurocomputing* 123 (2014) 216–232.
- [3] H. Zhang, D. Wang, L. Wang, Z. Bi, Y. Chen, A semantics-based method for clustering of Chinese web search results, *Enterp. Inf. Syst.* 8 (1) (2014) 147–165.
- [4] W. Fan, M. Gordon, P. Pathak, Discovery of context-specific ranking functions for effective information retrieval using genetic programming, *IEEE Trans. Knowl. Data Eng.* 16 (4) (2004) 523–527.
- [5] A. Salton, G. Wong, C.S. Yang, A vector space model for automatic indexing, *Commun. ACM* 18 (11) (1975) 609–613.
- [6] S.E. Robertson, Overview of the Okapi projects, *J. Doc.* 53 (1) (1997) 3–7.
- [7] C.X. Zhai, J. Lafferty, A study of smoothing methods for language models applied to information retrieval, *ACM Trans. Inf. Syst.* 22 (2) (2004) 179–214.
- [8] L. Page, R. Brin, S. Motwani, T. Winograd, The PageRank Citation Ranking: Bringing Order to the Web, Technical Report 1999-66, Stanford InfoLab, 1999.
- [9] J. Kleinberg, Authoritative sources in a hyperlinked environment, in: Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998, pp. 668–677.
- [10] M. du Plessis, G. de Kock, Incorporating structured text retrieval into the extended boolean model, *Comput. Inform.* 28 (5) (2009) 581–597.
- [11] M. Vajtersic, Special section on information retrieval by matrix methods on supercomputer systems, *Comput. Inform.* 30 (2) (2011) 201–204.
- [12] T. Liu, Learning to rank for information retrieval, *Found. Trends Inf. Retr.* 3 (3) (2009) 225–331.
- [13] P. Gursky, R. Pázman, P. Vojtáš, On supporting wide range of attribute types for top- k search, *Comput. Inform.* 28 (4) (2009) 483–513.
- [14] S. Niu, J. Lan, Y. Guo, X. Cheng, A new probabilistic model for top- k ranking problem, in: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, ACM, New York, NY, USA, 2012, pp. 2519–2522.
- [15] N. Fuhr, Optimum polynomial retrieval functions based on the probability ranking principle, *ACM Trans. Inf. Syst.* 7 (3) (1989) 183–204.
- [16] R. Herbrich, T. Graepel, K. Obermayer, Large margin rank boundaries for ordinal regression, *Adv. Large Margin Classif.* 88 (2) (2000) 115–132.
- [17] Z. Cao, T. Qin, T.Y. Liu, M.F. Tsai, H. Li, Learning to rank: from pairwise approach to listwise approach, in: Proceedings of the 24th International Conference on Machine Learning, ACM, New York, NY, USA, 2007, pp. 129–136.
- [18] P. Li, C. Burges, Q. Wu, Mcrank: learning to rank using multiple classification and gradient boosting, in: Proceedings of NIPS, 2007, pp. 897–904.
- [19] Y. Freund, R. Iyer, R.E. Schapire, Y. Singer, An efficient boosting algorithm for combining preferences, *J. Mach. Learn. Res.* 4 (2003) 933–969.
- [20] T. Qin, X. Zhang, M. Tsai, D. Wang, T. Liu, H. Li, Query-level loss functions for information retrieval, *Inf. Process. Manag.* 44 (2) (2008) 838–855.
- [21] F. Xia, T. Liu, Z. Wang, H. Li, Listwise approach to learning to rank – theory and algorithm, in: Proceedings of the 25th International Conference on Machine Learning, ACM, New York, NY, USA, 2008, pp. 1192–1199.
- [22] R.D. Luce, *Individual Choice Behavior: A Theoretical Analysis*, Wiley, New York, 1959.
- [23] T. Liu, J. Xu, T. Qin, W. Xiong, H. Li, Letor: benchmark dataset for research on learning to rank for information retrieval, in: Proceedings of the Learning to Rank Workshop in SIGIR, 2007, pp. 3–10.
- [24] F. Xia, T. Liu, H. Li, Statistical consistency of top- k ranking, in: Proceedings of the Conference on Neural Information Processing Systems, 2009, pp. 2098–2106.
- [25] W. Mahoney, P. Hospodka, W. Sousan, R. Nickell, Q. Zhu, A coherent measurement of web-search relevance, *IEEE Trans. Syst. Man Cybern. Part A: Syst. Hum.* 39 (6) (2009) 1176–1187.
- [26] Y. Lin, H. Lin, Z. Ye, S. Jin, X. Sun, Learning to rank with groups, in: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, ACM, New York, NY, USA, 2010, pp. 1589–1592.
- [27] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, Addison Wesley, Boston, USA, 1999.
- [28] A. Singhal, Modern information retrieval: a brief overview, *IEEE Data Eng. Bull.* 24 (4) (2001) 35–43.
- [29] K. Jarvelin, J. Kekalainen, IR evaluation methods for retrieving highly relevant documents, *ACM Trans. Inf. Syst.* 20 (4) (2002) 41–48.
- [30] Y. Lan, S. Niu, J. Guo, X. Cheng, Is top- k sufficient for ranking? in: Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, ACM, New York, NY, USA, 2013, pp. 1261–1270.



Yuan Lin received his Ph.D. at the Dalian University of Technology, China. Currently he works at School of Computer Science and Technology, Dalian University of Technology. His research interests include natural language processing, information retrieval, and learning to rank.



Hongfei Lin received his Ph.D. at the Northeastern University, China. Currently he is a Professor at School of Computer Science and Technology, Dalian University of Technology. His research interests are in information retrieval, text mining, natural language processing and sentimental analysis.

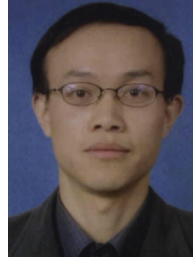


Kan Xu is a Ph.D. candidate at School of Computer Science and Technology, Dalian University of Technology. His research interests include text mining and Chinese information processing.



Ajith Abraham is the Director of Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, WA, USA, with a joint appointment in the IT For Innovations – Center of excellence at VSB – Technical University of Ostrava, Czech Republic. He also holds an Adjunct Professor appointment in Dalian Maritime University, China. He received the Ph.D. degree in Computer Science from Monash University, Melbourne, Australia. His research and development experience includes more than 23 years in the industry and academia. He works in a multidisciplinary environment involving machine intelligence, network security, various aspects of net-

works, e-commerce, Web intelligence, Web services, computational grids, data mining and their applications to various real-world problems. He has authored/co-authored more than 900 publications (h-index=50+), and some of the works have also won best paper awards at international conferences. He has given more than 60 plenary lectures and conference tutorials in these areas. He serves/has served the editorial board of over 50 International journals and has also guest edited over 40 special issues on various topics. Since 2008, he is the Chair of IEEE Systems, Man and Cybernetics Society Technical Committee on Soft Computing and a Distinguished Lecturer of IEEE Computer Society representing Europe (since 2011). More information at: <http://www.softcomputing.net>.



Hongbo Liu received his three level educations (B.Sc., M.Sc., Ph.D.) at the Dalian University of Technology, China. Currently he is a Professor at School of Information Science and Technology, Dalian Maritime University. His research interests are in system modeling and optimization involving soft computing, probabilistic modeling, cognitive computing, machine learning, data mining, etc. He participates and organizes actively international conference and workshop and international journals/publications.