Learning Maximal Marginal Relevance Model via Directly Optimizing Diversity Evaluation Measures

Conference Paper · July 2015

DOI: 10.1145/2766462.2767710

5 authors, including:

Long Xia
Chinese Academy of Sciences
9 PUBLICATIONS **230** CITATIONS

SEE PROFILE

Yanyan Lan
Chinese Academy of Sciences
152 PUBLICATIONS **3,737** CITATIONS

SEE PROFILE

Jiafeng Guo
Chinese Academy of Sciences
211 PUBLICATIONS **5,913** CITATIONS

SEE PROFILE

Learning Maximal Marginal Relevance Model via Directly Optimizing Diversity Evaluation Measures

Long Xia Jun Xu Yanyan Lan Jiafeng Guo Xueqi Cheng
CAS Key Lab of Network Data Science and Technology,
Institute of Computing Technology, Chinese Academy of Sciences
xialong@software.ict.ac.cn, {junxu, lanyanyan, guojiafeng, cxq}@ict.ac.cn

ABSTRACT

In this paper we address the issue of learning a ranking model for search result diversification. In the task, a model concerns with both query-document relevance and document diversity is automatically created with training data. Ideally a diverse ranking model would be designed to meet the criterion of maximal marginal relevance, for selecting documents that have the least similarity to previously selected documents. Also, an ideal learning algorithm for diverse ranking would train a ranking model that could directly optimize the diversity evaluation measures with respect to the training data. Existing methods, however, either fail to model the marginal relevance, or train ranking models by minimizing loss functions that loosely related to the evaluation measures. To deal with the problem, we propose a novel learning algorithm under the framework of Perceptron, which adopts the ranking model that *maximizes marginal relevance at ranking and can optimize any diversity evaluation measure in training*. The algorithm, referred to as PAMM (Perceptron Algorithm using Measures as Margins), first constructs positive and negative diverse rankings for each training query, and then repeatedly adjusts the model parameters so that the margins between the positive and negative rankings are maximized. Experimental results on three benchmark datasets show that PAMM significantly outperforms the state-of-the-art baseline methods.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information Search and Retrieval – *Retrieval Models*

General Terms

Algorithms

Keywords

search result diversification; maximal marginal relevance; directly optimizing evaluation measures

SIGIR'15, August 09 - 13, 2015, Santiago, Chile.

DOI: http://dx.doi.org/10.1145/2766462.2767710.

1. INTRODUCTION

It has been widely observed that users' information needs, described by keyword based queries, are often ambiguous or multi-faceted. It is important for commercial search engines to provide search results which balance query-document relevance and document diversity, called search result diversification [1, 30]. One of the key problems in search result diversification is ranking, specifically, how to develop a ranking model that can sort documents based on their relevance to the given query as well as the novelty of the information in the documents.

Methods for search result diversification can be categorized into heuristic approaches and learning approaches. The heuristic approaches construct diverse rankings with handcrafted ranking rules. As a representative method in the category, Carbonell and Goldstein [2] propose the maximal marginal relevance (MMR) criterion for guiding the construction ranking models. In MMR, constructing of a diverse ranking is formulated as a process of sequential document selection. At each iteration, the document with the highest marginal relevance is selected. The marginal relevance can be defined as, for example, a linear combination of the query-document relevance and the maximum distance of the document to the selected document set. A number of approaches have been proposed [8, 23, 24, 25] on the basis of the criterion and promising results have been achieved. User studies also shows that the user browsing behavior matches very well with the maximal marginal relevance criterion: usually users browse the web search results in a top-down manner, and perceive diverse information from each individual document based on what they have obtained in the preceding results [5]. Therefore, in a certain sense, we can say that maximal marginal relevance has been widely accepted as a criterion for guiding the construction of diverse ranking models.

Recently, machine learning approaches have been proposed for the task of search result diversification [14, 20, 22, 29, 31], especially the methods that can directly optimize evaluation measures on training data [16, 28]. Yue and Joachims [28] propose SVM-DIV which formulates the task as a problem of structured output prediction. In the model, the measure of subtopic diversity is directly optimized under the structural SVM framework. Liang et al. [16] propose to conduct personalized search result diversification via directly optimizing the measure of α-NDCG, also under the structural SVM framework. All of these methods try to resolve the mismatch between the objective function used in training and the final evaluation measure used in testing. Experimen-

tal results also showed that directly optimizing the diversity evaluation measures can indeed improve the diverse ranking performances [16, 28]. One problem with the direct optimization approaches is that it is hard, if not impossible, to define a ranking model that can meet the maximal marginal relevance criterion under the direct optimization framework.

In this paper, we aim to develop a new learning algorithm that utilizes the maximal marginal relevance model for ranking as well as can directly optimize any diversity evaluation measure in training. Inspired by the work of R-LTR [31] and Perceptron variations [7, 15], we propose a new algorithm for search result diversification, referred to as PAMM (Perceptron Algorithm using Measures as Margins). PAMM utilizes a sequential document selection process as its ranking model. In learning, it first generates positive rankings (ground truth rankings) and negative rankings for the training queries. It then repeats the process of estimating the probabilities for the rankings, calculating the margins between the positive rankings and negative rankings in terms of the ranking probabilities, and updating the model parameters so that the margins are maximized. We show that PAMM algorithm minimizes an upper bound of the loss function that directly defined over the diversity evaluation measures.

PAMM offers several advantages: 1) adopting the ranking model that meets the maximal marginal relevance criterion; 2) ability to directly optimize any diversity evaluation measure in training; 3) ability to use both positive rankings and negative rankings in training.

To evaluate the effectiveness of PAMM, we conducted extensive experiments on three public TREC benchmark datasets. The experimental results showed that our methods significantly outperform the state-of-the-art diverse ranking approaches including MMR, SVM-DIV, and R-LTR. We analyzed the results and showed that PAMM makes a good balance between the relevance and diversity via maximizing marginal relevance in ranking. We also showed that by directly optimizing a measure in training, PAMM can indeed enhance the ranking performances in terms of the measure.

The rest of the paper is organized as follows. After a summary of related work in Section 2, we describe the general framework of learning maximal marginal relevance model in Section 3. In Section 4 we discuss the proposed PAMM algorithm. Experimental results and discussions are given in Section 5. Section 6 concludes this paper and gives future work.

2. RELATED WORK

Methods of search result diversification can be categorized into heuristic approaches and learning approaches.

2.1 Heuristic approaches

It is a common practice to use heuristic rules to construct a diverse ranking list in search. Usually, the rules are created based on the observation that in diverse ranking a document's novelty depends on not only the document itself but also the documents ranked in previous positions. Carbonell and Goldstein [2] propose the maximal marginal relevance criterion to guide the design of diverse ranking models. The criterion is implemented with a process of iteratively selecting the documents from the candidate document set. At each iteration, the document with the highest marginal relevance score is selected, where the score is a linear combination of the query-document relevance and the maximum distance of the document to the documents in current result set. The marginal relevance score is then updated in the next iteration as the number of documents in the result set increases by one. More methods have been developed under the criterion. PM-2 [8] treats the problem of finding a diverse search result as finding a proportional representation for the document ranking. xQuAD [25] directly models different aspects underlying the original query in the form of sub-queries, and estimates the relevance of the retrieved documents to each identified sub-query. See also [3, 9, 10, 11, 21]

Heuristic approaches rely on the utility functions that can only use a limited number of ranking signals. Also, the parameter tuning cost is high, especially in complex search settings. In this paper, we propose a learning approach to construct diverse ranking models that can meet the maximal marginal relevance criterion.

2.2 Learning approaches

Methods of machine learning have been applied to search result diversification. In the approaches, rich features can be utilized and the parameters are automatically estimated from the training data. Some promising results have been obtained. For example, Zhu et al. [31] proposed the relational learning to rank model (R-LTR) in which the diverse ranking is constructed with a process of sequential document selection. The training of R-LTR amounts to optimizing the likelihood of ground truth rankings. More work please refer to [14, 20, 22, 29]. All these methods, however, formulate the learning problem as optimizing loss function that loosely related to diversity evaluation measures.

Recently methods that can directly optimize evaluation measures have been proposed and applied to search result diversification. Yue and Joachims [28] formulate the task of constructing a diverse ranking as a problem of predicting diverse subsets. Structural SVM framework is adopted to perform the training. Liang et al. [16] propose to conduct personalized search result diversification, also under the structural SVM framework. In the model, the loss function is defined based on the diversity evaluation measure of α-NDCG. Thus, the algorithm can be considered as directly optimizing α-NDCG in training. One issue with the approach is that it is hard to learn a maximal marginal relevance model under the structural SVM framework.

In this paper, we propose a Perceptron algorithm that can learn a maximal marginal relevance model, at the same time directly optimizing diversity evaluation measures.

3. LEARNING MAXIMAL MARGINAL RELEVANCE MODEL

We first describe the general framework of learning maximal marginal relevance model for search result diversification.

3.1 Maximal marginal relevance model

Suppose that we are given a query q, which is associated with a set of retrieved documents $X = \{\mathbf{x}_1, \cdots, \mathbf{x}_M\}$, where each document $\mathbf{x}_i$ is represented as a D-dimensional relevance feature vector. Let $R = \mathcal{R}^{M \times M \times K}$ denotes a 3-way tensor representing relationship among the M documents, where R_{ijk} stands for the k-th relationship feature of document $\mathbf{x}_i$ and document $\mathbf{x}_j$.

Algorithm 1 Ranking via maximizing marginal relevance

Input: documents X, document relation R, and ranking model parameters ω_r and ω_d
Output: ranking $\mathbf{y}$

1: $S_0 \leftarrow \phi$
2: **for** $r = 1, \cdots, M$ **do**
3: $\quad y(r) \leftarrow \arg\max_{j:\mathbf{x}_j \in X \setminus S_{r-1}} f_{S_{r-1}}(\mathbf{x}_j, R_j)$
4: $\quad S_r \leftarrow S_{r-1} \cup \{\mathbf{x}_{y(r)}\}$
5: **end for**
6: **return** $\mathbf{y}$

The maximal marginal relevance model creates a diverse ranking over X with a process of sequential document selection. At each step, the document with the highest marginal relevance is selected and added to the tail of the list [31]. Specifically, let $S \subseteq X$ be the set of documents have been selected for query q at one of the document selection step. Given S, the marginal relevance score of each document $\mathbf{x}_i \in X \setminus S$ at current step is defined as a linear combination of the query-document relevance and diversity of the document to the documents in S:

$$f_S(\mathbf{x}_i, R_i) = \omega_r^T \mathbf{x}_i + \omega_d^T h_S(R_i), \tag{1}$$

where $\mathbf{x}_i$ denotes the relevance feature vector of the document, $R_i \in \mathcal{R}^{M \times K}$ is the matrix representation of the relationship between document $\mathbf{x}_i$ and the other documents (note that $R_{ij} \in \mathcal{R}^K$ denotes the relationship feature vector of document pair $(\mathbf{x}_i, \mathbf{x}_j)$), and ω_r and ω_d are the weights for the relevance features and diversity features, respectively. The first term in Equation (1) represents the relevance of document $\mathbf{x}_i$ to the query and the second term represents the diversity of $\mathbf{x}_i$ w.r.t. documents in S. Following the practice in [31], the relational function $h_S(R_i)$ is defined as the minimal distance:

$$h_S(R_i) = \left(\min_{\mathbf{x}_j \in S} R_{ij1}, \cdots, \min_{\mathbf{x}_j \in S} R_{ijK} \right).$$

According to the maximal marginal relevance criterion, sequential document selection process can be used to create a diverse ranking, as shown in Algorithm 1. Specifically, given a query q, the retrieved documents X, and document relationship R, the algorithm initializes S_0 as an empty set. It then iteratively selects the documents from the candidate set. At iteration r ($r = 1, \cdots, M$), the document with the maximal marginal relevance score $f_{S_{r-1}}$ is selected and ranked at position r. At the same time, the selected document is inserted to S_{r-1}.

3.2 Learning the ranking model

Machine learning approaches can be used to learn the maximal marginal relevance model. Suppose we are given N labeled training queries $\{(X^{(n)}, R^{(n)}, J^{(n)})\}_{n=1}^N$, where $J^{(n)}$ denotes the human labels on the documents, in the form of a binary matrix. $J^{(n)}(i,s) = 1$ if document $\mathbf{x}_i^{(n)}$ contains the s-th subtopic of q_n and 0 otherwise[1]. The learning process, thus, amounts to minimize the loss over all of the training queries:

$$\min_{\omega_r, \omega_d} \sum_{n=1}^{N} L\left(\hat{\mathbf{y}}^{(n)}, J^{(n)}\right), \tag{2}$$

[1]Some datasets also use graded judgements. In this paper, we assume that all labels are binary.

Table 1: Summary of notations.

Notations	Explanations
q	query
$X = \{\mathbf{x}_1, \cdots, \mathbf{x}_M\}$	list of documents for q
$\mathbf{x}_i \in \mathcal{R}^D$	document relevant feature vector
$R \in \mathcal{R}^{M \times M \times K}$	relationship tensor among M documents
$\mathcal{Y}$	set of rankings over documents
$\mathbf{y} \in \mathcal{Y}$	the ranking of documents
$y(t) \in \{1, \cdots, M\}$	index of the document ranked at t
$S_r \subseteq X$	selected documents before iteration r
$f_S(\mathbf{x}_i, R_i)$	the scoring function at each step
$h_S(R_i)$	the relational function on R_i
ω_d	weights for relevance features
ω_r	weights for diversity features
J	human labels on document subtopics
$E(X, \mathbf{y}, J) \in [0, 1]$	diversity evaluation measure

where $\hat{\mathbf{y}}^{(n)}$ is the ranking constructed by the maximal marginal relevance model (Algorithm 1) for documents $X^{(n)}$, and $L(\hat{\mathbf{y}}^{(n)}, J^{(n)})$ is the function for judging the 'loss' of the predicted ranking $\mathbf{y}^{(n)}$ compared with the human labels $J^{(n)}$.

3.3 Diversity evaluation measures

In search result diversification, query level evaluation measures are used to evaluate the 'goodness' of a ranking model. These measures include α-NDCG [5], ERR-IA [4], and NRBP [6] etc. We utilize a general function $E(X, \mathbf{y}, J) \in [0, 1]$ to represent the evaluation measures. The first argument of E is the set of candidate documents, the second argument is a ranking $\mathbf{y}$ over documents in X, and the third argument is the human judgements. E measures the agreement between $\mathbf{y}$ and J.

As an example of diversity evaluation measures, α-NDCG [5] is a variation of NDCG [13] in which the newly found subtopics are rewarded and redundant subtopics are penalized. The α-NDCG score at rank k can be defined by replacing the raw gain values in standard NDCG@k with novelty-baised gains:

$$\alpha\text{-NDCG@}k = \frac{\sum_{r=1}^{k} NG(r)/\log(r+1)}{\sum_{r=1}^{k} NG^*(r)/\log(r+1)}, \tag{3}$$

where $NG(r) = \sum_s J(y(r), s)(1-\alpha)^{C_s(r-1)}$ is the novelty-biased gain at rank r in ranking $\mathbf{y}$, $C_s(r-1) = \sum_{k=1}^{r-1} J(y(k), s)$ denotes the number of documents observed within top $r-1$ that contain the s-th subtopic, $NG^*(r)$ is the novelty-biased gain at rank r in a positive ranking, and $y(k)$ denotes the index of the document ranked at k. Usually the parameter α is set to 0.5.

ERR-IA [4] is another popular used diversity evaluation measure. Given a query with several different subtopics s, the probability of each intent $\Pr(s|q)$ can be estimated, where $\sum_s \Pr(s|q) = 1$. The intent-aware ERR at rank k can be computed as:

$$\text{ERR-IA@}k = \sum_s \Pr(s|q)\text{ERR@}k(s), \tag{4}$$

where ERR@$k(s)$ is the expected reciprocal rank score at k in terms of subtopic s.

Table 1 gives a summary of the notations described above.

4. OUR APPROACH: PAMM

4.1 Evaluation measure as loss function

We aim to maximize the diverse ranking accuracy in terms of a diversity evaluation measure on the training data. Thus, the loss function in Equation (2) becomes

$$\sum_{n=1}^{N}\Big(1-E(X^{(n)},\hat{\mathbf{y}}^{(n)},J^{(n)})\Big). \quad (5)$$

It is difficult to directly optimize the loss as E is a non-convex function.

We resort to optimize the upper bound of the loss function under the framework of structured output prediction. According to Theorem (2) in [27], we know that the loss function defined in Equation (5) can be upper bounded by the function defined over the ranking pairs:

$$\sum_{n=1}^{N}\max_{\substack{\mathbf{y}^+\in\mathcal{Y}^{+(n)},\\ \mathbf{y}^-\in\mathcal{Y}^{-(n)}}}\Big(E(X^{(n)},\mathbf{y}^+,J^{(n)})-E(X^{(n)},\mathbf{y}^-,J^{(n)})\Big)\cdot \Big[\!\Big[F(\mathbf{y}^+,X^{(n)},R^{(n)})\le F(\mathbf{y}^-,X^{(n)},R^{(n)})\Big]\!\Big], \quad (6)$$

where $\mathcal{Y}^{+(n)}$ is the set of all possible 'positive' rankings (rankings whose α-NDCG/ERR-IA equals to one) for the n-th query, $\mathcal{Y}^{-(n)}$ is the set of all possible 'negative' rankings (rankings whose α-NDCG/ERR-IA is less than one) for the n-the query, $[\![\cdot]\!]$ is one if the condition is satisfied otherwise zero, and $F(X,R,\mathbf{y})$ is the query level ranking model. F takes the document set X, document relationship R, and ranking over the document $\mathbf{y}$ as inputs. The output of F is the confidence score of the ranking $\mathbf{y}$. The predicted $\hat{\mathbf{y}}^{(n)}$ in Equation (5) can be considered as the ranking that maximizes F:

$$\hat{\mathbf{y}}^{(n)}=\arg\max_{\mathbf{y}\in\mathcal{Y}^{(n)}}F(X^{(n)},R^{(n)},\mathbf{y}), \quad (7)$$

where $\mathcal{Y}^{(n)}$ is the set of all possible rankings over $X^{(n)}$. Here F is defined as the probability of generating the ranking list $\mathbf{y}$ with a process of iteratively selecting the top ranked documents from the remaining documents, and using the marginal relevance function f_S in Equation (1) as the selection criterion:

$$\begin{aligned}F(X,R,\mathbf{y}) =&\Pr(\mathbf{y}|X,R)\\ =&\Pr(\mathbf{x}_{y(1)}\cdots\mathbf{x}_{y(M)}|X,R)\\ =&\prod_{r=1}^{M-1}\Pr(\mathbf{x}_{y(r)}|X,S_{r-1},R)\\ =&\prod_{r=1}^{M-1}\frac{\exp\{f_{S_{r-1}}(\mathbf{x}_i,R_{y(r)})\}}{\sum_{k=r}^{M}\exp\{f_{S_{r-1}}(\mathbf{x}_i,R_{y(k)})\}}\end{aligned} \quad (8)$$

where $y(r)$ denotes the index of the document ranked at the r-th position in $\mathbf{y}$, $S_{r-1}=\{\mathbf{x}_{y(k)}\}_{k=1}^{r-1}$ is the documents ranked at the top $r-1$ positions in $\mathbf{y}$, $f_{S_{r-1}}(\mathbf{x}_i,R_i)$ is the marginal relevance score of document $\mathbf{x}_i$ w.r.t. the selected documents in S_{r-1}, and $S_0=\phi$ is an empty set. With the definition of F, it is obvious that the maximal marginal relevance process of Algorithm 1 actually greedily searches the solution for optimizing the problem of Equation (7).

To conduct the optimization under the Perceptron framework, the upper bound of Equation (6) is further relaxed, by replacing the max with sum and moving the term $(E(X^{(n)},\mathbf{y}^+,J^{(n)})-E(X^{(n)},\mathbf{y}^-,J^{(n)}))$ into $[\![\cdot]\!]$ as margin. The upper bound of Equation (6) becomes:

$$\sum_{n=1}^{N}\sum_{\mathbf{y}^+;\mathbf{y}^-}\Big[\!\Big[F(X^{(n)},R^{(n)},\mathbf{y}^+)-F(X^{(n)},R^{(n)},\mathbf{y}^-)\le E(X^{(n)},\mathbf{y}^+,J^{(n)})-E(X^{(n)},\mathbf{y}^-,J^{(n)})\Big]\!\Big]. \quad (9)$$

This is because $\sum_i x_i\ge\max_i x_i$ if $x_i\ge 0$ holds for all i, and $[\![x-y\le z]\!]\ge z\cdot[\![x\le y]\!]$ holds if $z\in[0,1]$. Please note that we assume $E(X,\mathbf{y}^+,J)\in[0,1]$ and thus we have $(E(X^{(n)},\mathbf{y}^+,J^{(n)})-E(X^{(n)},\mathbf{y}^-,J^{(n)}))\in[0,1]$ because $E(X^{(n)},\mathbf{y}^+,J^{(n)})>E(X^{(n)},\mathbf{y}^-,J^{(n)})$.

4.2 Direct optimization with Perceptron

The loss function in Equation (9) can be optimized under the framework of Perceptron. In this paper, inspired by the work of structured Perceptron [7] and Perceptron algorithm with uneven margins [15], we have developed a novel learning algorithm to optimize the loss function in Equation (9). The algorithm is referred to as PAMM and is shown in Algorithm 2.

PAMM takes a training set $\{(X^{(n)},R^{(n)},J^{(n)})\}_{n=1}^{N}$ as input and takes the diversity evaluation measure E, learning rate η, number of positive rankings per query τ^+, and number of negative rankings per query τ^- as parameters. For each query q_n, PAMM first generates τ^+ positive rankings $PR^{(n)}$ and τ^- negative rankings $NR^{(n)}$ (line (2) and line (3)). $PR^{(n)}$ and $NR^{(n)}$ play as the random samples of $\mathcal{Y}^{+(n)}$ and $\mathcal{Y}^{-(n)}$, respectively. PAMM then optimizes the model parameters ω_r and ω_d iteratively in a stochastic manner over the ranking pairs: at each round, for each pair between a positive ranking and a negative ranking $(\mathbf{y}^+,\mathbf{y}^-)$, the gap of these two rankings in terms of the query level ranking model $\Delta F=F(X,R,\mathbf{y}^+)-F(X,R,\mathbf{y}^-)$ is calculated based on current parameters ω_r and ω_d (line (9)). If ΔF is smaller than the margin in terms of evaluation measure $\Delta E=E(X,\mathbf{y}^+,J)-E(X,\mathbf{y}^-,J)$ (line (10)), the model parameters will be updated so that ΔF will be enlarged (line (11) and line (12)). The iteration continues until convergence. Finally, PAMM outputs the optimized model parameters (ω_r,ω_d).

Next, we will explain the key steps of PAMM in detail.

4.2.1 Generating positive and negative rankings

In PAMM, it is hard to directly conduct the optimization over the sets of positive rankings $\mathcal{Y}^{+(n)}$ and negative rankings $\mathcal{Y}^{-(n)}$, because in total these two sets have $M!$ rankings if the candidate set contains M documents. Thus, PAMM samples the rankings to reduce the training time.

For each training query, PAMM first samples a set of positive rankings. Algorithm 3 illustrates the procedure. Similar to the online ranking algorithm shown in Algorithm 1, the positive rankings are generated with a sequential document selection process and the selection criteria is the diversity evaluation measure E. After generating the first positive ranking $\mathbf{y}^{(1)}$, the algorithm constructs other positive rankings based on $\mathbf{y}^{(1)}$, by randomly swapping the positions of two documents whose subtopic coverage are identical.

For each training query, PAMM also samples a set of negative rankings. Algorithm 4 shows the procedure. The algorithm simply generates random rankings iteratively. If the generated ranking is not a positive ranking and satisfies the

Algorithm 2 The PAMM Algorithm

Input: training data $\{(X^{(n)}, R^{(n)}, J^{(n)})\}_{n=1}^{N}$, learning rate η, diversity evaluation measure E, number of positive rankings per query τ^+, number of negative rankings per query τ^-.
Output: model parameters (ω_r, ω_d)

1: **for** $n = 1$ **to** N **do**
2: $PR^{(n)} \leftarrow$PositiveRankings$(X^{(n)}, J^{(n)}, E, \tau^+)$ {Algorithm 3}
3: $NR^{(n)} \leftarrow$NegativeRankings$(X^{(n)}, J^{(n)}, E, \tau^-)$ {Algorithm 4}
4: **end for**
5: initialize $\{\omega_r, \omega_d\} \leftarrow$ random values in $[0, 1]$
6: **repeat**
7: **for** $n = 1$ **to** N **do**
8: **for all** $\{\mathbf{y}^+, \mathbf{y}^-\} \in PR^{(n)} \times NR^{(n)}$ **do**
9: $\Delta F \leftarrow F(X^{(n)}, R^{(n)}, \mathbf{y}^+) - F(X^{(n)}, R^{(n)}, \mathbf{y}^-)$ {$F(X, R, \mathbf{y})$ is defined in Equation (8)}
10: **if** $\Delta F \leq E(X^{(n)}, \mathbf{y}^+, J^{(n)}) - E(X^{(n)}, \mathbf{y}^-, J^{(n)})$ **then**
11: calculate $\nabla\omega_r^{(n)}$ and $\nabla\omega_d^{(n)}$ {Equation (10) and Equation (11)}
12: $(\omega_r, \omega_d) \leftarrow (\omega_r, \omega_d) + \eta \times (\nabla\omega_r^{(n)}, \nabla\omega_d^{(n)})$
13: **end if**
14: **end for**
15: **end for**
16: **until** convergence
17: **return** (ω_r, ω_d)

user predefined constraints (e.g, α-NDCG@20 ≤ 0.8), the ranking will be added into the ranking set NR.

Please note that in some extreme cases Algorithm 3 and Algorithm 4 cannot create enough rankings. In our implementations, the algorithms are forced to return after running enough iterations.

4.2.2 Updating ω_r and ω_d

Given a ranking pair $(\mathbf{y}^+, \mathbf{y}^-) \in PR^{(n)} \times NR^{(n)}$, PAMM updates ω_r and ω_d as

$$\omega_r \leftarrow \omega_r + \eta \times \nabla\omega_r \text{ and } \omega_d \leftarrow \omega_d + \eta \times \nabla\omega_d,$$

if $F(X, R, \mathbf{y}^+) - F(X, R, \mathbf{y}^-) \leq E(X, \mathbf{y}^+, J) - E(X, \mathbf{y}^-, J)$. The goal of the update is to enlarge the margin between $\mathbf{y}^+$ and $\mathbf{y}^-$ in terms of query level model: $\Delta F = F(X, R, \mathbf{y}^+) - F(X, R, \mathbf{y}^-)$. For convenience of calculation, we resort to the problem of

$$\max_{\omega_r, \omega_d} \log \frac{F(X, R, \mathbf{y}^+)}{F(X, R, \mathbf{y}^-)},$$

because $F(X, R, \mathbf{y}) > 0$ and $\log(\cdot)$ is a monotonous increasing function. Thus, $\nabla\omega_r$ can be calculated as the gradient:

$$\begin{aligned}\nabla\omega_r =& \frac{\partial \log \frac{F(X,R,\mathbf{y}^+)}{F(X,R,\mathbf{y}^-)}}{\partial\omega_r} \\ =& \frac{\partial \log F(X, R, \mathbf{y}^+)}{\partial\omega_r} - \frac{\partial \log F(X, R, \mathbf{y}^-)}{\partial\omega_r},\end{aligned} \tag{10}$$

Algorithm 3 PositiveRankings

Input: documents X, diversity labels J, evaluation measure E, and the number of positive rankings τ^+
Output: positive rankings PR

1: **for** $r = 1$ **to** $|X|$ **do**
2: $y^{(1)}(r) \leftarrow \arg\max_{j:\mathbf{x}_j \in X \setminus S_{r-1}}$ $E\left(S_{r-1} \cup \{\mathbf{x}_j\}, \left(y^{(1)}(1), \cdots, y^{(1)}(r-1), j\right), J\right)$
3: $S_r \leftarrow S_{r-1} \cup \{\mathbf{x}_{y^{(1)}(r)}\}$
4: **end for**
5: $PR \leftarrow \{\mathbf{y}^{(1)}\}$
6: **while** $|PR| < \tau^+$ **do**
7: $\mathbf{y} \leftarrow \mathbf{y}^{(1)}$
8: $(k, l) \leftarrow$ randomly choose two documents whose human labels are identical, i.e., $J(y(k)) = J(y^{(1)}(l))$
9: $y(k) \leftrightarrow y(l)$ {swap documents at rank k and l}
10: **if** $\mathbf{y} \notin PR$ **then**
11: $PR \leftarrow PR \cup \{\mathbf{y}\}$
12: **end if**
13: **end while**
14: **return** PR

Algorithm 4 NegativeRankings

Input: documents X, diversity labels J, evaluation measure E, and number of negative rankings τ^-
Output: NR

1: $NR = \phi$
2: **while** $|NR| < \tau^-$ **do**
3: $\mathbf{y} \leftarrow$ random shuffle $(1, \cdots, |X|)$
4: **if** $\mathbf{y} \notin NR$ **and** $E(X, \mathbf{y}, J)$ is as expected **then**
5: $NR \leftarrow NR \cup \{\mathbf{y}\}$
6: **end if**
7: **end while**
8: **return** NR

where

$$\begin{aligned}\frac{\partial \log F(X, R, \mathbf{y})}{\partial\omega_r} &= \frac{\partial \sum_{j=1}^{|X|-1} \log \Pr(\mathbf{x}_{y(j)} | X \setminus S_{j-1}, R)}{\partial\omega_r} \\ &= \sum_{j=1}^{|X|-1} \left\{ \mathbf{x}_{y(j)} - \frac{\sum_{k=j}^{|X|} \mathbf{x}_{y(k)} \exp\{f_{S_{j-1}}(\mathbf{x}_{y(k)}, R_{y(k)})\}}{\sum_{k=j}^{|X|} \exp\{f_{S_{j-1}}(\mathbf{x}_{y(k)}, R_{y(k)})\}} \right\}.\end{aligned}$$

Similarly, $\nabla\omega_d$ can be calculated as

$$\nabla\omega_d = \frac{\partial \log F(X, R, \mathbf{y}^+)}{\partial\omega_d} - \frac{\partial \log F(X, R, \mathbf{y}^-)}{\partial\omega_d}, \tag{11}$$

where

$$\begin{aligned}\frac{\partial \log F(X, R, \mathbf{y})}{\partial\omega_d} = \sum_{j=1}^{|X|-1} \Bigg\{ & h_{S_{j-1}}(R_{y(j)}) - \\ & \frac{\sum_{k=j}^{|X|} h_{S_{j-1}}(R_{y(k)}) \exp\{f_{S_{j-1}}(\mathbf{x}_{y(k)}, R_{y(k)})\}}{\sum_{k=j}^{|X|} \exp\{f_{S_{j-1}}(\mathbf{x}_{y(k)}, R_{y(k)})\}} \Bigg\}.\end{aligned}$$

Intuitively, the gradients $\nabla\omega_r$ and $\nabla\omega_d$ are calculated so that the line 12 of Algorithm 2 will increase $F(X, R, \mathbf{y}^+)$ and decrease $F(X, R, \mathbf{y}^-)$.

4.3 Analysis

We analyzed time complexity of PAMM. The learning process of PAMM (Algorithm 2) is of order $\mathcal{O}(T \cdot N \cdot \tau^+ \cdot \tau^- \cdot$

Table 2: Statistics on WT2009, WT2010 and WT2011.

Dataset	#queries	#labeled docs	#subtopics per query
WT2009	50	5149	$3 \sim 8$
WT2010	48	6554	$3 \sim 7$
WT2011	50	5000	$2 \sim 6$

$M^2 \cdot (D + K)$), where T denotes the number of iterations, N the number of queries in training data, τ^+ the number of positive rankings per query, τ^- the number of negative rankings per query, M the maximum number of documents for queries in training data, D the number of relevance features, and K the number of diversity features. The time complexity of online ranking prediction (Algorithm 1) is of order $\mathcal{O}(M^2(D + K))$.

PAMM is a simple yet powerful learning algorithm for search result diversification. It has several advantages compared with the existing learning methods such as R-LTR [31], SVM-DIV [28], and structural SVM [26].

First, PAMM employs a more reasonable ranking model. The model follows the maximal marginal relevance criterion and can be implemented with a process of sequential document selection. In contrast, structural SVM approaches [26] calculate all of the ranking scores within a single step, as that of in relevance ranking. The marginal relevance of each document cannot be taken into consideration at ranking time.

Second, PAMM can incorporate any diversity evaluation measure in training, which makes the algorithm focus on the specified measure when updating the model parameters. In contrast, R-LTR only minimizes loss function that is loosely related to diversity evaluation measures and SVM-DIV is trained to optimize the subtopic coverage.

Third, PAMM utilizes the pairs between the positive rankings and the negative rankings in training, which makes it possible to leverage more information in training. Specifically, it enables PAMM algorithm to enlarge the margins between the positive rankings and negative rankings when updating the parameters. In contrast, R-LTR only uses the information in the positive rankings and the training is aimed to maximizing the likelihood.

5. EXPERIMENTAL RESULTS

5.1 Experiment setting

We conducted experiments to test the performances of PAMM using three TREC benchmark datasets for diversity tasks: TREC 2009 Web Track (WT2009), TREC 2010 Web Track (WT2010), and TREC 2011 Web Track (WT2011). Each dataset consists of queries, corresponding retrieved documents, and human judged labels. Each query includes several subtopics identified by TREC assessors. The document relevance labels were made at the subtopic level and the labels are binary[2]. Statistics on the datasets are given in Table 2.

All the experiments were carried out on the ClueWeb09 Category B data collection[3], which comprises of 50 million English web documents. Porter stemming, tokenization, and stop-words removal (using the INQUERY list) were applied to the documents as preprocessing. We conducted 5-fold cross-validation experiments on the three datasets. For each dataset, we randomly split the queries into five even subsets. At each fold three subsets were used for training, one was used for validation, and one was used for testing. The results reported were the average over the five trials.

As for evaluation measures, α-NDCG@k (Equation (3)) with $\alpha = 0.5$ and $k = 20$ is used. We also used ERR-IA@k (Equation (4)) with $k = 20$ to evaluate the performances.

We compared PAMM with several types of baselines. The baselines include the conventional relevance ranking models in which document diversity is not taken into consideration.

Query likelihood (QL) [18] language models for information retrieval.

ListMLE [17] a representative learning-to-rank model for information retrieval.

We also compared PAMM with three heuristic approaches to search result diversification in the experiments.

MMR [2] a heuristic approach to search result diversification in which the document ranking is constructed via iteratively selecting the document with the maximal marginal relevance.

xQuAD [25] a representative heuristic approach to search result diversification.

PM-2 [8] another widely used heuristic approach to search result diversification.

Please note that these three baselines require a prior relevance function to implement their diversification steps. In our experiments, ListMLE was chosen as the relevance function.

Learning approaches to search result diversification are also used as baselines in the experiments.

SVM-DIV [28] a representative learning approach to search result diversification. It utilizes structural SVMs to optimize the subtopic coverage. SVM-DIV does not consider relevance. For fair performance comparison, in the baseline, we first apply ListMLE to capture relevance, and then apply SVM-DIV to re-rank the top-K retrieved documents.

Structural SVM [26] Structural SVM can be configured to directly optimize diversity evaluation measures, as shown in [16]. In the paper, we used structural SVM to optimize α-NDCG@20 and ERR-IA@20, denoted as StructSVM(α-NDCG) and StructSVM(ERR-IA), respectively.

R-LTR [31] a state-of-the-art learning approach to search result diversification. The ranking function is a linear combination of the relevance score and diversity score between the current document and those previously selected. Following the practice in [31], in our experiments we used the results of R-LTR$_{min}$ which defines the relation function $h_S(R)$ as the minimal distance.

5.2 Features

As for features, we adopted the features used in the work of R-LTR [31]. There are two types of features: the relevance features which capture the relevance information of a query with respect to a document, and the diversity features which represent the relation information among documents. Table 3 and Table 4 list the relevance features and diversity features used in the experiments, respectively.

[2]WT2011 has graded judgements. In this paper we treat them as binary.

[3]http://boston.lti.cs.cmu.edu/data/clueweb09

[4]http://www.dmoz.org

Table 3: Relevance features used in the experiments. The first 4 lines are query-document matching features, each applied to the fields of body, anchor, title, URL, and the whole documents. The latter 3 lines are document quality features. [31]

Name	Description	# Features
TF-IDF	The tf-idf model	5
BM25	BM25 with default parameters	5
LMIR	LMIR with Dirichlet smoothing	5
MRF[19]	MRF with ordered/unordered phrase	10
PageRank	PageRank score	1
#inlinks	number of inlinks	1
#outlinks	number of outlinks	1

Table 4: The seven diversity features used in the experiments. Each feature is extracted over two documents. [31]

Name	Description
Subtopic Diversity	Euclidean distance based on PLSA[12]
Text Diversity	Cosine-based distance on term vectors
Title Diversity	Text diversity on title
Anchor Text Diversity	Text diversity on anchor
ODP-Based Diversity	ODP[4] taxonomy-based distance
Link-Based Diversity	Link similarity of document pair
URL-Based Diversity	URL similarity of document pair

5.3 Experiments with TREC datasets

In the experiments, we made use of the benchmark datasets of WT2009, WT2010, and WT2011 from the TREC Web Track, to test the performances of PAMM.

PAMM has to tune some parameters. The learning rate parameter η was tuned based on the validation set during each experiment. In all of the experiments in this subsection, we set the number of positive rankings per query $\tau^+ = 5$, and number of negative rankings per query $\tau^- = 20$. As for the parameter E of PAMM, α-NDCG@20 and ERR-IA@20 were utilized. The results for PAMM using α-NDCG@20 in training are denoted as PAMM(α-NDCG). The PAMM results using ERR-IA@20 as measures are denoted as PAMM(ERR-IA).

The experimental results on WT2009, WT2010, and WT2011 are reported in Table 5, Table 6, and Table 7, respectively. Numbers in parentheses are the relative improvements compared with the baseline method of query likelihood (QL). Boldface indicates the highest score among all runs. From the results, we can see that PAMM(α-NDCG) and PAMM(ERR-IA) outperform all of the baselines on all of the three datasets in terms of both α-NDCG@20 and ERR-IA@20. We conducted significant testing (t-test) on the improvements of PAMM(α-NDCG) over the baselines in terms of α-NDCG@20 and ERR-IA@20. The results indicate that all of the improvements are statistically significant (p-value < 0.05). We also conducted t-test on the improvements of PAMM(ERR-IA) over the baselines in terms of α-NDCG@20 and ERR-IA@20. The improvements are also statistically significant. All of the results show that PAMM is effective for the task of search result diversification.

We observed that on all of the three datasets, PAMM(α-NDCG) trained with α-NDCG@20 performed best in terms of α-NDCG@20 while PAMM(ERR-IA) trained with ERR-IA@20 performed best in terms of ERR-IA@20. The results indicate that PAMM can enhance diverse ranking performances in terms of a measure by using the measure in training. We will further discuss the phenomenon in next section.

Table 5: Performance comparison of all methods in official TREC diversity measures for WT2009.

Method	ERR-IA@20	α-NDCG@20
QL	0.164	0.269
ListMLE	0.191(+16.46%)	0.307(+14.13%)
MMR	0.202(+23.17%)	0.308(+14.50%)
xQuAD	0.232(+41.46%)	0.344(+27.88%)
PM-2	0.229(+39.63%)	0.337(+25.28%)
SVM-DIV	0.241(+46.95%)	0.353(+31.23%)
StructSVM(α-NDCG)	0.260(+58.54%)	0.377(+40.15%)
StructSVM(ERR-IA)	0.261(+59.15%)	0.373(+38.66%)
R-LTR	0.271(+65.24%)	0.396(+47.21%)
PAMM(α-NDCG)	0.284(+73.17%)	**0.427**(+58.74%)
PAMM(ERR-IA)	**0.294**(+79.26%)	0.422(+56.88%)

Table 6: Performance comparison of all methods in official TREC diversity measures for WT2010.

Method	ERR-IA@20	α-NDCG@20
QL	0.198	0.302
ListMLE	0.244(+23.23%)	0.376(+24.50%)
MMR	0.274(+38.38%)	0.404(+33.77%)
xQuAD	0.328(+65.66%)	0.445(+47.35%)
PM-2	0.330(+66.67%)	0.448(+48.34%)
SVM-DIV	0.333(+68.18%)	0.459(+51.99%)
StructSVM(α-NDCG)	0.352(+77.78%)	0.476(+57.62%)
StructSVM(ERR-IA)	0.355(+79.29%)	0.472(+56.29%)
R-LTR	0.365(+84.34%)	0.492(+62.91%)
PAMM(α-NDCG)	0.380(+91.92%)	**0.524**(+73.51%)
PAMM(ERR-IA)	**0.387**(+95.45%)	0.511(+69.21%)

Table 7: Performance comparison of all methods in official TREC diversity measures for WT2011.

Method	ERR-IA@20	α-NDCG@20
QL	0.352	0.453
ListMLE	0.417(+18.47%)	0.517(+14.13%)
MMR	0.428(+21.59%)	0.530(+17.00%)
xQuAD	0.475(+34.94%)	0.565(+24.72%)
PM-2	0.487(+38.35%)	0.579(+27.81%)
SVM-DIV	0.490(+39.20%)	0.591(+30.46%)
StructSVM(α-NDCG)	0.512(+45.45%)	0.617(+36.20%)
StructSVM(ERR-IA)	0.513(+45.74%)	0.613(+35.32%)
R-LTR	0.539(+53.13%)	0.630(+39.07%)
PAMM(α-NDCG)	0.541(+53.70%)	**0.643**(+41.94%)
PAMM(ERR-IA)	**0.548**(+55.68%)	0.637(+40.62%)

5.4 Discussions

We conducted experiments to show the reasons that PAMM outperforms the baselines, using the results of the WT2009 dataset as examples.

5.4.1 *Effect of maximizing marginal relevance*

We found that PAMM makes a good tradeoff between the query-document relevance and document diversity via maximizing marginal relevance. Here we use the result with regard to query number 24 ("diversity" which contains 4 subtopics), to illustrate why our method is superior to the baseline method of Structural SVM trained with α-NDCG@20 (denoted as StructSVM(α-NDCG)). Note that structural

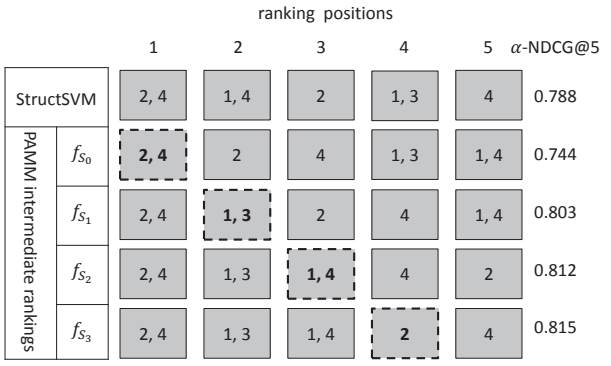

Figure 1: Example rankings from WT2009. Each shaded block represents a document and the number(s) in the block represent the subtopic(s) covered by the document.

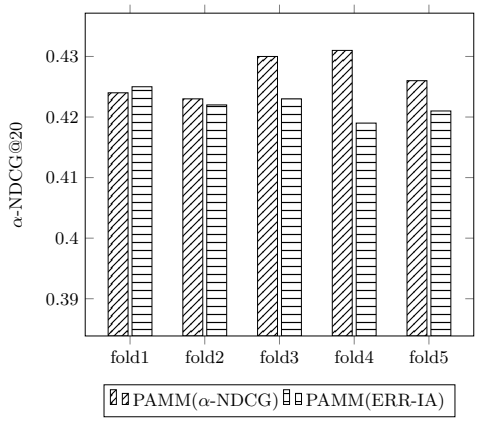

Figure 2: Performance in terms of α-NDCG@20 when model is trained with α-NDCG@20 or ERR-IA@20.

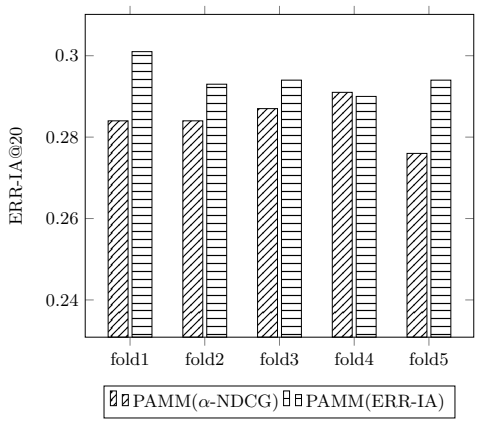

Figure 3: Performance in terms of ERR-IA@20 when model is trained with α-NDCG@20 or ERR-IA@20.

SVM cannot leverage the marginal relevance in its ranking model. Figure 1 shows the top five ranked documents by StructSVM(α-NDCG), as well as four intermediate rankings generated by PAMM(α-NDCG) (denoted as $f_{S_0}, f_{S_1}, f_{S_2}$, and f_{S_3}). The ranking denoted as f_{S_r} is generated as: first sequentially selecting the documents for ranking positions of $1, 2, \cdots, r-1$ with models $f_{S_0}, f_{S_1}, \cdots, f_{S_{r-2}}$, respectively; then ranking the remaining documents with $f_{S_{r-1}}$. For example, the intermediate ranking denoted as f_{S_2} is generated as: selecting one document with f_{S_0} and setting it to rank 1, then selecting one document with f_{S_1} and set it to rank 2, and finally ranking the remaining documents with f_{S_2} and putting them to the tail of the list. Each of the shaded block indicates a document and the number(s) in the block indicates the subtopic(s) assigned to the document by the human annotators. The performances in terms of α-NDCG@5 are also shown in the last column. Here we used α-NDCG@5 because only the top 5 documents are shown.

The results in Figure 1 indicate the effectiveness of the maximal marginal relevance criterion. We can see that the α-NDCG@5 increases steadily with the increasing rounds of document selection iterations. In the first iteration, f_{S_0} selects the most relevant document and puts it to the first position, without considering the diversity. Thus, the α-NDCG@5 of the ranking generated by f_{S_0} is lower than that of by StructSVM(α-NDCG). In the second iteration, the ranking function f_{S_1} selects the document associated with subtopics 1 and 3 and ranks it to the second position, according to the maximal marginal relevance criterion. From the view point of diverse ranking, this is obviously a better choice than StructSVM(α-NDCG) made, which selects the document with subtopics 1 and 4. (Note that both Structural SVM and PAMM select the document with subtopics 2 and 4 for the first position.) In the following steps, f_{S_2} and f_{S_3} select documents for ranking positions of 3 and 4, also following the maximal marginal relevance criterion. As a result, f_{S_1}, f_{S_2}, and f_{S_3} outperforms StructSVM(α-NDCG).

5.4.2 Ability to improve the evaluation measures

We conducted experiments to see whether PAMM has the ability to improve the diverse ranking quality in terms of a measure by using the measure in training. Specifically, we trained models using α-NDCG@20 and ERR-IA@20 and evaluated their accuracies on the test dataset in terms of both α-NDCG@20 and ERR-IA@20. The experiments were conducted for each fold of the cross validation and performances on each fold are reported. Figure 2 and Figure 3 show the results in terms of α-NDCG@20 and ERR-IA@20, respectively. From Figure 2, we can see that on all of the 5 folds (except fold 1), PAMM(α-NDCG) trained with α-NDCG@20 performs better in terms of α-NDCG@20. Similarly, from Figure 3, we can see that on all of the 5 folds (except fold 4), PAMM(ERR-IA) trained with ERR-IA@20 performs better in terms of ERR-IA@20. Similar results have also been observed in experiments on other datasets (see the results in Table 5, Table 6, and Table 7). All of the results indicate that PAMM can indeed enhance the diverse ranking quality in terms of a measure by using the measure in training.

5.4.3 Effects of positive and negative rankings

We examined the effects of the number of positive rankings generated per query (parameter τ^+). Specifically, we compared the performances of PAMM(α-NDCG) w.r.t. different τ^+ values. Figure 4 shows the performance curve in terms of α-NDCG@20. The performance of R-LTR base-

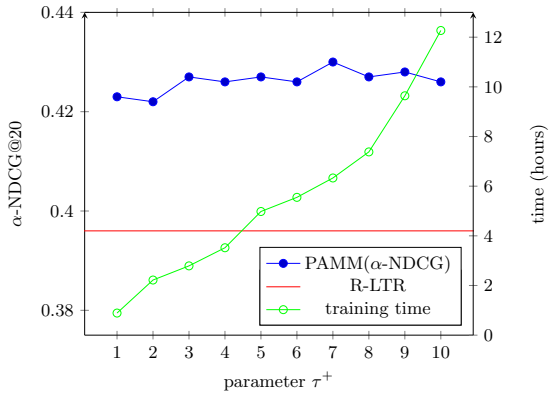

Figure 4: Ranking accuracies and training time w.r.t. τ^+.

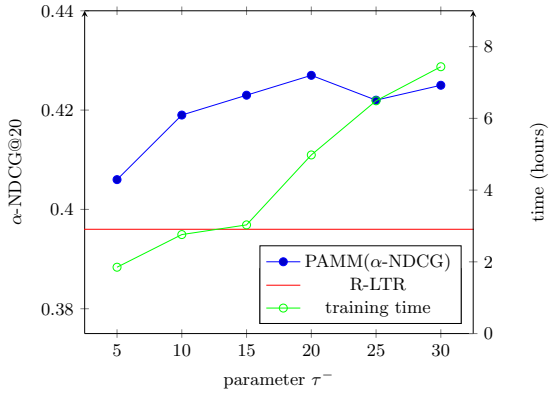

Figure 5: Ranking accuracies and training time w.r.t τ^-.

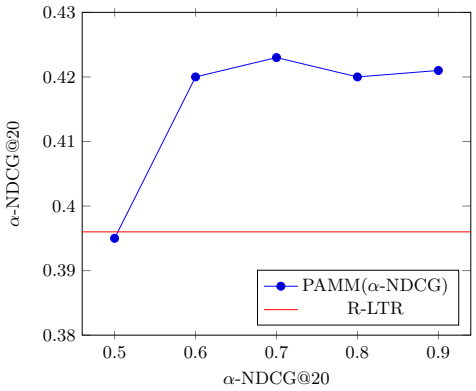

Figure 6: Ranking accuracies w.r.t. different α-NDCG@20 values of the negative rankings.

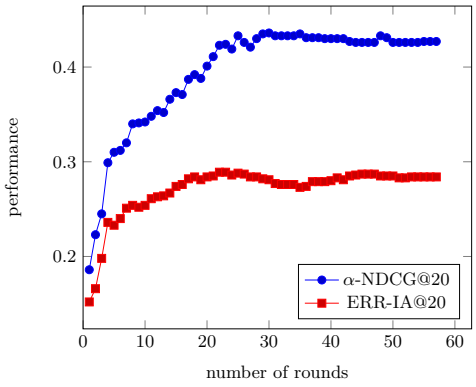

Figure 7: Learning curve of PAMM(α-NDCG).

line is also shown for reference. From the result, we can see that the curve does not change much with different τ^+ values, which indicates the robustness of PAMM. Figure 4 also shows training time (in hours) w.r.t. different τ^+ values. The training time increased dramatically with large τ^+, because more ranking pairs are generated for training. In our experiments τ^+ was set to 5.

We further examined the effect of the number of negative rankings per query (parameter τ^-). Specifically, we compared the performances of PAMM(α-NDCG) w.r.t. different τ^- and the results are shown in Figure 5. From the results, we can see that the performance of PAMM increasing steadily with the increasing τ^- values until $\tau^- = 20$, which indicates that PAMM can achieve better ranking performance with more information from the negative rankings. As the cost, the training time increased dramatically, because more training instances are involved in training. In our experiments, τ^- was set to 20.

We also conducted experiments to show the effect of sampling the negative rankings with different α-NDCG values. Specifically, in each of the experiment, we configured the Algorithm 4 to choose the negative rankings whose α-NDCG@20 values are 0.5, 0.6, 0.7, 0.8, and 0.9, respectively. Figure 6 shows the performances of PAMM(α-NDCG) w.r.t. different α-NDCG@20 values of the sampled negative rankings. From the results, we can see that PAMM performs best when the α-NDCG@20 of the sampled negative rankings ranges from 0.6 to 0.9. The results also indicate that PAMM is robust and not very sensitive to different methods of sampling the negative rankings.

5.4.4 *Convergence*

Finally we conducted experiments to show whether PAMM can converge in terms of the diversity evaluation measures. Specifically, we showed the learning curve of PAMM(α-NDCG) in terms of α-NDCG@20 and ERR-IA@20 during the training phase. At each training iteration the model parameters are outputted and evaluated on the test data. Figure 7 shows the performance curves w.r.t. the number of training iterations. From the results, we can see that the ranking accuracy of that PAMM(α-NDCG) steadily improves in terms of both α-NDCG@20 and ERR-IA@20, as the training goes on. PAMM converges and returns after running about 60 iterations. We also observed that in all of our experiments, PAMM usually converges and returns after running 50∼100 iterations. Similar phenomenon was also observed from the learning curve of PAMM(ERR-IA). The results indicates that PAMM converges fast and conducts the training efficiently.

6. CONCLUSION AND FUTURE WORK

In this paper we have proposed a novel algorithm for learning ranking models in search result diversification, referred to as PAMM. PAMM makes use of the maximal marginal relevance model for constructing the diverse rankings. In training, PAMM directly optimizes the diversity evaluation measures on training queries under the framework of Perceptron. PAMM offers several advantages: employs a ranking model that follows the maximal marginal relevance criterion, ability to directly optimize any diversity evaluation measure, and ability to utilize both positive rankings and negative rankings in training. Experimental results based on three benchmark datasets show that PAMM significantly outperforms the state-of-the-art baseline methods including SVM-DIV, structural SVM, and R-LTR.

Future work includes theoretical analysis on the convergence, generalization error, and other properties of the PAMM algorithm, and improving the efficiency of PAMM in both offline training and online prediction.

7. ACKNOWLEDGMENTS

This research work was funded by the 973 Program of China under Grants No. 2014CB340401, No. 2012CB316303, the 863 Program of China under Grants No. 2014AA015204, the National Natural Science Foundation of China under Grant No. 61232010, No. 61425016, No. 61173064, No. 61472401, No. 61203298, and No. 61202214.

We would like to express our gratitude to Prof. Chengxiang Zhai who has offered us valuable suggestions in the academic studies.

8. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *Proceedings of the 2th ACM WSDM*, pages 5–14, 2009.

[2] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st ACM SIGIR*, pages 335–336, 1998.

[3] B. Carterette and P. Chandar. Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proceedings of the 18th ACM CIKM*, pages 1287–1296, 2009.

[4] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM CIKM*, pages 621–630, 2009.

[5] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st ACM SIGIR*, pages 659–666, 2008.

[6] C. L. Clarke, M. Kolla, and O. Vechtomova. An effectiveness measure for ambiguous and underspecified queries. In *Proceedings of the 2nd ICTIR*, pages 188–199, 2009.

[7] M. Collins. Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In *EMNLP '02*, pages 1–8, 2002.

[8] V. Dang and W. B. Croft. Diversity by proportionality: an election-based approach to search result diversification. In *Proceedings of the 35th ACM SIGIR*, pages 65–74, 2012.

[9] S. Gollapudi and A. Sharma. An axiomatic approach for result diversification. In *Proceedings of the 18th WWW*, pages 381–390, 2009.

[10] S. Guo and S. Sanner. Probabilistic latent maximal marginal relevance. In *Proceedings of the 33rd ACM SIGIR*, pages 833–834, 2010.

[11] J. He, V. Hollink, and A. de Vries. Combining implicit and explicit topic representations for result diversification. In *Proceedings of the 35th ACM SIGIR*, pages 851–860, 2012.

[12] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd ACM SIGIR*, pages 50–57, 1999.

[13] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, Oct. 2002.

[14] L. Li, K. Zhou, G.-R. Xue, H. Zha, and Y. Yu. Enhancing diversity, coverage and balance for summarization through structure learning. In *Proceedings of the 18th WWW*, pages 71–80, 2009.

[15] Y. Li, H. Zaragoza, R. Herbrich, J. Shawe-Taylor, and J. Kandola. The perceptron algorithm with uneven margins. In *Proceedings of the 19th ICML*, pages 379–386, 2002.

[16] S.-S. Liang, Z.-C. Ren, and M. de Rijke. Personalized search result diversification via structured learning. In *Proceedings of the 20th ACM SIGKDD*, pages 751–760, 2014.

[17] T.-Y. Liu. *Learning to Rank for Information Retrieval*. Springer, 2011.

[18] C. D. Manning, P. Raghavan, and H. SchÜtze. *An Introduction to Information Retrieval*. Cambridge University Press, 2009.

[19] D. Metzler and W. B. Croft. A markov random field model for term dependencies. In *Proceedings of the 28th ACM SIGIR*, pages 472–479, 2005.

[20] L. Mihalkova and R. Mooney. Learning to disambiguate search queries from short sessions. In *Machine Learning and Knowledge Discovery in Databases*, volume 5782, pages 111–127, 2009.

[21] F. Radlinski and S. Dumais. Improving personalized web search using result diversification. In *Proceedings of the 29th ACM SIGIR*, pages 691–692, 2006.

[22] F. Radlinski, R. Kleinberg, and T. Joachims. Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th ICML*, pages 784–791, 2008.

[23] D. Rafiei, K. Bharat, and A. Shukla. Diversifying web search results. In *Proceedings of the 19th WWW*, pages 781–790, 2010.

[24] K. Raman, P. Shivaswamy, and T. Joachims. Online learning to diversify from implicit feedback. In *Proceedings of the 18th ACM SIGKDD*, pages 705–713, 2012.

[25] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th WWW*, pages 881–890, 2010.

[26] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *J. Mach. Learn. Res.*, 6:1453–1484, Dec. 2005.

[27] J. Xu, T.-Y. Liu, M. Lu, H. Li, and M. Wei-Ying. Directly optimizing evaluation measures in learning to rank. In *Proceedings of the 31th ACM SIGIR*, pages 107–114, 2008.

[28] Y. Yue and T. Joachims. Predicting diverse subsets using structural svms. In *Proceedings of the 25th ICML*, pages 1224–1231, 2008.

[29] Y. Yue and T. Joachims. Interactively optimizing information retrieval systems as a dueling bandits problem. In *Proceedings of the 26th ICML*, pages 1201–1208, 2009.

[30] C. X. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th ACM SIGIR*, pages 10–17, 2003.

[31] Y. Zhu, Y. Lan, J. Guo, X. Cheng, and S. Niu. Learning for search result diversification. In *Proceedings of the 37th ACM SIGIR*, pages 293–302, 2014.