Elsevier Editorial System(tm) for Expert Systems With Applications

Manuscript Draft

Manuscript Number: ESWA-D-17-02634

Title: A Learnable Search Result Diversification Method

Article Type: Full length article

Keywords: Search Result Diversification;
Explicit Diversification Model;
Learning Model;
Markov Random Fields

Corresponding Author: Mr. Hai-Tao Zheng, Ph.D

Corresponding Author's Institution: Graduate school at Shenzhen, Tsinghua university

First Author: Hai-Tao Zheng, Ph.D

Order of Authors: Hai-Tao Zheng, Ph.D; Jinxin Han; Zhuren Wang; Xi Xiao

A Learnable Search Result Diversification Method

Hai-Tao Zheng[a,b,*], Jinxin Han[a,c], Zhuren Wang[a,d], Xi Xiao[a,e]

[a] *Tsinghua-Southampton Web Science Laboratory, Graduate School at Shenzhen, Tsinghua University, China*
[b] *zheng.haitao@sz.tsinghua.edu.cn*
[c] *hanjx16@mails.tsinghua.edu.cn*
[d] *wang-zr14@mails.tsinghua.edu.cn*
[e] *xiaox@sz.tsinghua.edu.cn*

Abstract

Search result diversification is to tackle the ambiguous queries and multi-faced information needs. The search result diversification problem can be formalized as a balance between the relevance score and the diversity score. Most previous diversification models utilize a predefined function to calculate the diversity score. The values of parameters need to be tuned by manual experiments. It is time-consuming and hard to reach optimal result in diversity evaluation. Proposing a learnable approach to solve the above problem is a pressing task. Therefore we introduce a Learnable Search Result Diversification model called L-SRD. On this basis, we redefine the diversity function and derive our loss function as the likelihood loss of ground truth generation. Stochastic gradient descent algorithms are employed to optimize the values of weightiness. Finally we derive our ranking function to generate the diverse list sequentially. Benefit from the learning model, the values of parameter are determined automatically and get optimal. The experiments on TREC web tracks show that our approach outperforms several existing diversification models significantly.

Keywords: Search Result Diverification, Explicit Diversification Model, Learning Model, Markov Random Fields

2017 MSC: 00-01, 99-00

*Corresponding author

1. Introduction

There are many ambiguous queries in search system. The keyword **apple** may refer to the Apple, the one of the most famous companies in the world, or the electronics Apple manufactures. It may be the most familiar fruit also. There are many aspects of information needs underlying a simple query. How to produce a good quality diverse result is our main concern.

The existing diversification approaches have been categorized as either implicit approaches or explicit approaches. The implicit approaches assume each document representing its own aspect and promote diversity by selecting documents for different aspect based on the difference of their vocabulary (Carbonell & Goldstein, 1998). It is a less effective model for the reason that it cannot express the inherent meaning of a document good enough only on the literal level (Agrawal et al., 2009; Santos et al., 2010; Dang & Croft, 2012). The explicit approaches are proposed to overcome the weakness of the implicit approaches. They explicitly formalize the aspects underlying a query and select documents that cover different aspects. The xQuAD and PM2 (Santos et al., 2010; Dang & Croft, 2012) are classic explicit models.

Most diversification approaches just utilize a predefined function to calculate the diversity score based on query aspects. It is subjective and hard to reach optimal result. The value of parameters need to be tuned by the experiment repeatedly without intention, causing a time-consuming optimizing problem to find the most suitable parameters.

In this paper, we treat the Query Aspect Diversification as a learning problem and propose Learnable Search Result Diversification (L-SRD for short). Different from the previous models (Santos et al., 2010; Agrawal et al., 2009), we incorporate various features into diversity measurement. We extract the features based on the Markov Random Field (MRF for short), which enables the integration of various types of features. The value of parameters can be determined automatically, which saves the manual labour, and the parameters are more optimal. Firstly we redefine the diversity function and derive our loss

function as the likelihood loss of ground truth generation. Then Stochastic gradient descent algorithms are employed to optimize the value of weight. Finally we derive our ranking function to generate the diverse list sequentially.

We conduct a series of experiments to demonstrate L-SRD is more effective than other diversification models in terms of the official evaluation metrics including α-NDCG (Clarke et al., 2008), ERR-IA (Chapelle et al., 2009), NRBP (Clarke et al., 2009b) and the classical diversification metrics such as Precision-IA and Aspect Recall. Additionally, we get a remarkable performance in robust evaluation.

The main contributions of our work are listed as follows:

1. L-SRD is the first method to introduce the learning mechanism to the Query Aspect Diversification model. We conduct inference for the loss function based on its sequential selection model, which solves the parameter tuning problem automatically at the same time.
2. We are the first one utilizing the Markov Random Field to integrate different types of features to address the diversity measurement problem for Query Aspect Search Result Diversification.
3. We conduct extensive experiments to verify that L-SRD achieve better performance comparing with the existing diversification models.

The remainder of this paper is organised as follows. Section 2 introduces the current research situation on the search result diversification. Section 3 describes the definition of the loss function and the estimation of parameters. Sections 4 details the experiments setup on the TREC web track and their evaluations. In Section 5, we summarize our achievements and give future works.

2. Related Work

The existing approaches differ from the usage of the aspects underlying the query. We categorize them into two groups: implicit approach and explicit approach.

Implicit approaches iteratively select a document that differs from the selected documents in terms of literal meaning, such as vocabulary. The implicit approaches include maximal marginal relevance (MMR) method and its probabilistic variants (Zhai et al., 2003). The special process of MMR proposed by Carbonell and Goldstein (Carbonell & Goldstein, 1998) is selecting the document iteratively, and meanwhile, both content-based relevance and diversity relation between current selected document and the previously selected documents are considered. It is a low effective approach (Santos et al., 2010; Drosou & Pitoura, 2010).

Explicit approaches make use of the aspects underlying the query to select documents that cover different aspects as far as possible. The algorithms such as IA-select (Agrawal et al., 2009), xQuAD (Santos et al., 2010) and RxQuAD (Vargas et al., 2012) are proposed to reduce redundancy on the aspect levels. These methods select the document that covers more novel aspects. The PM-1 and PM-2 (Dang & Croft, 2012) models pay more attention to maintain the proportionality of aspects. They produce the ranked result according to the proportionality of aspects. Intrinsic diversity products a series of successor queries to figure out the appropriate content to cover (Raman et al., 2013). Yu & Ren (2014) formulized the problem as a 0-1 multiple subtopic knapsack problem. These approaches lay a good foundation for diversification model nowadays. By utilizing the existing diversification model, we transfer our attention to how to measure the relevance between the aspects and the candidate documents better. Just like Dou et al. (2011), proposed a multi-dimensional aspect model based on IA-select model and xQuAD model. Capannini et al. (2011) made use of query logs to improve the xQuAD (Santos et al., 2010) model. Liang et al. (2014) proposed a data fusion model based on PM-2 model. From the viewpoint of term level, Dang & Croft (2013) set up his model by referring PM model. He et al. (2012) combined the implicit and the explicit model by dint of the IA-select model, which get considerable results. Besides the research on the model itself, Radlinski et al. (2008) promoted the diversity by leveraging the external resources such as query logs.

In addition, Zhu et al. (2014) proposed a learning model without considering the aspects underlying the query. Yue & Joachims (2008) proposed Structural SVMs to model the diversity but discarded the relevance. In this paper, we focus on the diversity measurement in aspect level and the combination of relevance and diversity, which is different from existing learning approaches and shows promising experimental performance.

3. Learning Approach for Search Result Diversification

3.1. Mining aspects underlying the query

The key step for Query Aspects Diversification model is mining the aspects underlying the query. With the help of query aspects we can generate the diverse ranking list by minimizing the redundancy on the basis of the aspects. We mine the query aspects like Santos et al. (Santos et al., 2010), issuing the query to the commercial search engine (we use Yahoo) and get back the query suggestion result list as the aspects. Nextly, we can use these aspects as a new query to search the candidate document set D and we can get the relevance score between the aspect q_i and each document d in D, which can be formalized as $P(q_i|d)$.

3.2. Topic diversity model

Traditional topic diversity model (Santos et al., 2010) is a greedy approximation. It sequentially selects the "local-best" document from the candidate document set. The ranking function in topic diversity model is formalized as follows:

$$f(d, \bar{S}) = (1 - \lambda)P(d|q) + \lambda \sum_{q_i \in Q} P(q_i|q)P(d|q_i)P(\bar{S}|q_i). \quad (1)$$

where d denotes for the current document to be considered in the sequential process, $\bar{S}$ denotes for the unselected document set (equal to the $D \backslash S$ in Fig. (1)), q denotes for the query, λ is a balance parameter for a trade-off between relevance and diversity, q_i denotes for the aspects underlying the query q. In each

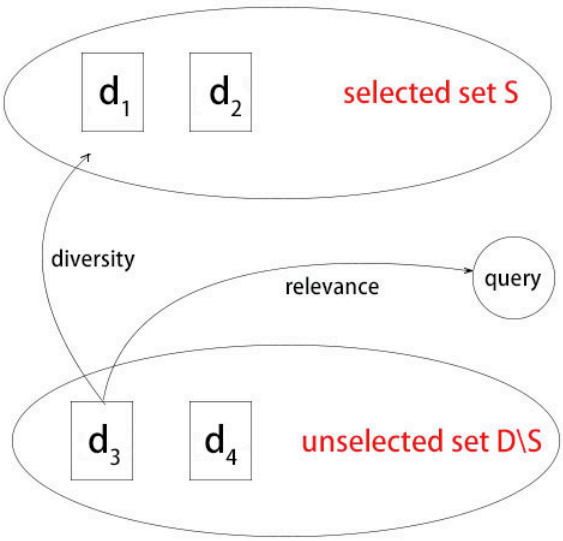


Figure 1: An illustration for sequential selection in topic diversity model

step, we add the "local-best" document with the highest score to the selected set.

As for Eq. (1), the left part corresponds to the relevance score and the right part corresponds to the diversity score. We are looking forward to redefine the estimation of diversity score $P(\bar{S}|q_i)$. According to the conditional probabilistic formula, the task can be formalized as follows:

$$P(\bar{S}|q_i) = \frac{P(\bar{S}, q_i)}{P(q_i)} \stackrel{rank}{=} P(\bar{S}, q_i) \tag{2}$$

where $P(q_i)$ denotes the occurrence rate of aspects q_i corresponding to query q, which is usually regard to be normalized as $1/n$ (Santos et al., 2010) (n denotes the number of aspects). Because the value of $P(q_i)$ are equal and do not impact on the result of ranking, we neglect $P(q_i)$.

The main concern is how to define feature function for $P(\bar{S}, q_i)$. There are many ways to integrate different features, just like linear regression, logistic regression and some other ways. Under our situation, we use Markov Random Field (MRF for short) to model $P(\bar{S}, q_i)$. We can benefit from its convenient combination of different types of features and we can get its derivation easily.

3.3. Feature extraction via MRF

A MRF is a probabilistic model defined on a undirected graph G. In the MRF model, the nodes represent the random variables and the edges represent dependencies between these variables. In our study, the nodes represent the aspect q_i and the unselected set $\bar{S}$. Consequently, we compute the joint probability defined over the graph G as follows:

$$P(\bar{S}, q_i) = \frac{\prod_{l \in L(G)} \phi_l(l)}{Z}, \tag{3}$$

where $L(G)$ is the cliques over the graph G, $\phi_l(l)$ is a potential function defined over the clique l, and $Z = \sum_{\bar{S}, q_i} \prod_l(l)$ is the normalization factor to ensure that $P(\bar{S}, q_i)$ satisfies a probability distribution.

The potential function is usually defined like:

$$\phi_l(l) = exp(\lambda_l f_l(l)), \tag{4}$$

where $f_l(l)$ is a feature function defined over clique l, and λ_l is the corresponding weightiness factor. By applying log function and neglecting normalization factor, the final feature function is formalized as follows:

$$P(\bar{S}, q_i) \overset{rank}{=} \sum_{l \in L(G)} \lambda_l f_l(l). \tag{5}$$

Note that Eq. (5) is derived from Eq. (3) by neglecting the log function because its form is more simple for derivation and the simplifying does not impact the learning and ranking. Nextly, we specify the structure of graph G and its clique set $L(G)$ to derive our final feature functions.

Fig. (2) shows the three types of cliques in the MRF for our model. The formal description about feature extraction based on three cliques is given as follows:

1. Based on l_{sd}. The high occupancy of aspects q_i reflects the high potential relevance for q_i with respect to $\bar{S}$.

   - We define $f_{ave-topic} = ave_{d \in \bar{S}} P(q_i|d)$ for feature function on this clique. $P(q_i|d)$ is the aspects distribution measurement which we have mentioned in section 3.1.

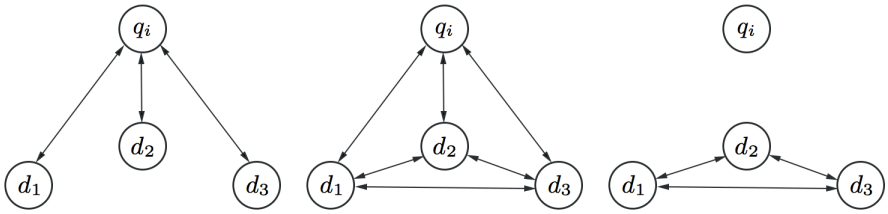


Figure 2: An illustration for three types of cliques. The graph G contains a aspects node q_i and three document nodes (just for example) that correspond to the documents in the unselected set $\bar{S}$. (a) l_{sd} contains q_i and a single document node; (b) l_{sD} includes q_i and the whole $\bar{S}$; (c) l_D only contains $\bar{S}$.

2. Based on l_{sD}. The clique involves the inter-relationships in the candidate set $\bar{S}$.

   - We use the maximal, minimal, standard deviation of the $P(q_i|d)_{d\in\bar{S}}$ as feature functions defined upon clique l_{sD}.
   - For the sake of minimizing redundancy, we use the $num_{q_i}(\bar{S}) - num_{q_i}(S)$ as a feature function defined on this clique too. $num_{q_i}(x)$ represents the number of documents in set x with respect to aspects q_i.

3. Based on l_D. Both l_{sd} and l_{sD} consider relations of documents with respect to the above two aspects. The clique l_D only takes into account the relations among documents excluding aspects q_i. Previous research has shown that the aspect-independent property can indicate the relevance of documents for q_i (Kurland & Domshlak, 2008).

   - We use the entropy of all the documents d in $\bar{S}$: $P_{entropy}(d) \stackrel{def}{=} -\sum_{w\in d} P(w|d)\log p(w|d)$ as feature function, where w is a term and $p(w|d)$ is the probability that w appears in d (given by the language model).
   - Spam ratio (inspired by the Web Spam Classification (Fetterly et al., 2004)) is used for feature function, too.

To conclude, by replacing the feature function into equation (1) and putting the parameter λ into the learning process, the ranking function is given as follows:

$$f(d, \bar{S}) = \lambda_r P(d|q) + \lambda_n \sum_{q_i \in Q} [P(q_i|q)P(d|q_i) \sum_{l \in L(G(q_i))} \lambda_l f_l(l)] \tag{6}$$

where $L(G(q_i))$ represents the clique set L from the graph G which is built around the aspects q_i, $f_l(l)$ stands for the feature function defined on the clique l. There exists a parameter λ in equation (1), it is a balance parameter between relevance and diversity. In our learning method, we use λ_r and λ_n to replace it and we can infer its value by learning process.

3.4. Loss function

We define the loss function as a likelihood loss of generation probability:

$$L(rank(D, C), Y) = -\log P(Y|D), \tag{7}$$

where *rank* denotes the ranking function in our model, C is the feature function defined in unselected set $\bar{S}$, D denotes all the candidate documents, and Y is the final result of search result diversification. Because our L-SRD model is a sequential selection model, it can be viewed as maximizing probability of correctly choosing the top-n document from unselected set:

$$\begin{aligned} P(Y|D) &= P(y_1, y_2, ..., y_n|D) \\ &= P(y_1|D)P(y_2|D\backslash S_1) \cdots P(y_n|D\backslash S_{n-1}), \end{aligned} \tag{8}$$

where $y_1, ..., y_n$ is the ground truth for search result diversification task with respect to query q, n represents the top n result generated by the sequential selection process, the index i denotes its ranking position, S_{i-1} denotes the selected set after $i-1$ iterations, the probability $P(y_i|D\backslash S_{i-1})$ represents the probability that select the document y_i under the condition of $D\backslash S_{i-1}$.

On the basis of the Plackett-Luce Model (Marden, 1996), we derive the steps in our generation process shown as follows:

$$P(Y|D) = \prod_{i=1}^{n} P(y_i|D\backslash S_{i-1}) = \prod_{i=1}^{n} \frac{exp(f(y_i, D\backslash S_{i-1}))}{\sum_{k=i}^{n} exp(f(y_k, D\backslash S_{i-1}))}, \tag{9}$$

where S_0 means empty set $\emptyset$, function f corresponds to Eq. (6). Incorporating Eq.(9) into Eq.(7), we get the definition of the loss function as follows:

$$L(f(D,C),Y) = -\sum_{i=1}^{n} \log(\frac{exp(f(y_i, D\backslash S_{i-1}))}{\sum_{k=i}^{n} exp(f(y_k, D\backslash S_{i-1}))}). \tag{10}$$

To get the final loss function, we simplify Eq. (6) by uniting the parameter λ_n and λ_l (because the parameter in our model all can be decided by the learning process):

$$f(d,\bar{S}) = \lambda_r P(d|q) + \vec{\mu_d} \cdot N_{1\ldots L}(d, q_i, \bar{S}) \tag{11}$$

$$N_l(d, q_i, \bar{S}) = \sum_{q_i \in Q} P(q_i|q) P(d|q_i) f_l(l) \quad (l \in L(G(q_i))) \tag{12}$$

where $\vec{\mu_d}$ represents a L-dimensional weight vector, L stands for the number of features, $N_{1\ldots L}$ denotes a series of function vectors, l is the cliques defined on $\bar{S}$ and q_i.

The total loss function is formulized as follows:

$$-\sum_{i=1}^{T_r}\sum_{j=1}^{n} \log(\frac{exp(\lambda_r P(y_j|q) + \vec{\mu_d} \cdot N_{1\ldots L}(y_j, q_i, \bar{S}))}{\sum_{k=j}^{n} exp(\lambda_r P(y_k|q) + \vec{\mu_d} \cdot N_{1\ldots L}(y_k, q_i, \bar{S}))}) \tag{13}$$

where T_r denotes the number of training examples.

3.5. Learning and prediction

Given the precise definition of loss function, the next step is minimizing the loss function to get the best performance. Firstly, we generate the training data and apply the optimization method. Nextly, we use our ranking function to predict the final diverse ranking result.

In this study, we use the data in TREC dataset in a format of quadruples: $(q^{(i)}, RD^{(i)}, T_i, J(s_j^{(i)}|t_k))$, where $q^{(i)}$ means the i-th query q, $RD^{(i)}$ is the corresponding related documents set, T_i represent the aspect underlying the query $q^{(i)}$ which are provided by official labeler, and $J(d_j^{(i)}|t_k)$ represents the judgement factor whether the j-th document $d_j^{(i)}$ in $RD^{(i)}$ covers the aspect t_k. Note that the last two elements in quadruples are used to calculate the score

of evaluation metrics (e.g. α-nDCG), we cannot make use of it directly in our model.

At first, we should generate a approximate ground truth for training set. So we construct a list y_i which maximize the diversity metrics, such as α-nDCG, ERR-IA, etc. In our study, we use ERR-IA to measure the results which is described as function f_{ERR-IA}. In algorithm 1, at the every i-th step in loop structure, we select the document d from $D \backslash S_{i-1}$ to maximize the function f_{ERR-IA} and update the $S \backslash D_{i-1}$ by adding the document d. By recording the best document in every step, we get our final ideal rankling list as our training data.

ALGORITHM 1: Ideal ranking list construction algorithm

Input: $(q^{(i)}, RD^{(i)}, T_i, J(d_j^{(i)}|t_k)), f_{ERR-IA}$

Output: $y_1, y_2, ..., y_n$

1: initialize $S_0 = \emptyset$
2: **for** $k = 1$ to n **do**
3: $\quad bestDoc \leftarrow \arg\max_{d \in RD^{(i)} \backslash S_{k-1}} f_{ERR-IA}(d \cup S_{k-1})$
4: $\quad S_k = S_{k-1} \cup bestDoc$
5: $\quad y_k = bestDoc$
6: **end for**

Nextly, we use the stochastic gradient descent method to optimize the loss function as shown in Algorithm 2. At every step in algorithm 2, we calculate the gradient according to Eq. (14)-(15) and update the value of weight. The gradient in step i at training set D_{init} is computed as follows:

$$\Delta\lambda_r^{(i)} = \sum_{j=1}^{n} \Big(\frac{\sum_{k=j}^{n} P(y_j|q) exp(\lambda_r^{(i-1)} P(y_j|q) + \vec{\mu_d}^{(i-1)} \cdot N_{1...L}(y_j, q_i, \bar{S}))}{\sum_{k=j}^{n} exp(\lambda_r^{(i-1)} P(y_k|q) + \vec{\mu_d}^{(i-1)} \cdot N_{1...L}(y_k, q_i, \bar{S}))} - \frac{P(y_j|q) exp(\lambda_r^{(i-1)} P(y_j|q) + \vec{\mu_d}^{(i-1)} \cdot N_{1...L}(y_j, q_i, \bar{S}))}{exp(\lambda_r^{(i-1)} P(y_k|q) + \vec{\mu_d}^{(i-1)} \cdot N_{1...L}(y_k, q_i, \bar{S}))} \Big) \quad (14)$$

$$\Delta\vec{\mu_d}^{(i)} = \sum_{j=1}^{n}\Big(\frac{\sum_{k=j}^{n} N_l(y_j,q_i,\bar{S})exp(\lambda_r^{(i-1)}P(y_j|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_j,q_i,\bar{S}))}{\sum_{k=j}^{n} exp(\lambda_r^{(i-1)}P(y_k|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_k,q_i,\bar{S}))} - \frac{N_l(y_j,q_i,\bar{S})exp(\lambda_r^{(i-1)}P(y_j|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_j,q_i,\bar{S}))}{exp(\lambda_r^{(i-1)}P(y_k|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_k,q_i,\bar{S}))}\Big) \tag{15}$$

ALGORITHM 2: Parameter learning algorithm

Input: training data: $D_{init}^{Tr}, (y_1...y_n)^{Tr}$

parameter: learning rate η, tolerate ϵ

Output: $\lambda_r, \vec{\mu_d}$

Initialize $\lambda_r, \vec{\mu_d}$
repeat
  $\lambda_r^{(0)} = \lambda_r$, $\vec{\mu_d}^{(0)} = \vec{\mu_d}$
  Randomly choose one of the training data
  for $i = 1, ..., n$ **do**
    Compute the gradient $\Delta\lambda_r^{(i)}$ and $\Delta\vec{\mu_d}^{(i)}$
    Update: $\lambda_r^{(i)} = \lambda_r^{(i-1)} - \eta\Delta\lambda_r^{(i)}$, $\vec{\mu_d}^{(i)} = \vec{\mu_d}^{(i-1)} - \eta\Delta\vec{\mu_d}^{(i)}$
  end for
  $\lambda_r = \lambda_r^{(n)}$, $\vec{\mu_d} = \vec{\mu_d}^{(n)}$
until change for value of loss function below the tolerate ϵ

Finally, we propose a sequential prediction method as described in Algorithm 3. At the i-th step in algorithm 3, we select the best document d from $D\backslash S_{i-1}$ to maximize our ranking score and update the candidate set $D\backslash S_{i-1}$ by adding document d. By recording the best document in every step, we predict the final

diverse ranking list as result.

ALGORITHM 3: The prediction process

Input: query q and its retrieval result D_{init}, weight λ_r and λ_{nl}

Output: $y_1, y_2, ..., y_n$

1: $S_0 = \emptyset$

2: **for** $i = 1, ..., n$ **do**

3: $\quad best = \arg\max_{d \in D \setminus S_{i-1}} f(d, D \setminus S_{i-1})$

4: $\quad S_i = S_{i-1} \cup best$

5: $\quad y_i = best$

6: **end for**

4. Experiment

4.1. Dataset and evaluation metrics

Our query set contains of 150 queries, from TREC web track 2009 (WT2009) (Clarke et al., 2009a), WT2010, WT2011(50 for each). Evaluation is done on the ClueWeb09 Category B retrieval collection[1]. The collection consists of nearly 50 millions web pages in English. There is a list of aspects for each query with binary relevance judgements, which are provided by TREC assessors.

There are three mainly evaluation metrics we use to evaluate the performance of our method: α-NDCG, ERR-IA, NRBP (computed at cutoff 50). These metrics penalize redundancy in a different degree for the document at each sorted position to maximize the aspects coverage. Additionally, we report our result using Precision-IA (Agrawal et al., 2009) and Aspect Recall too. To measure the robustness, we use Win/Loss ratio metrics.

The evaluation metrics are reported at different cutoffs. We use {5, 10, 20} as our cutoffs to set up our experiments. These cutoffs focus on the evaluation at early ranks, which are particularly important in a Web search context (Jansen et al., 1998). The α are set to 0.5 in our experiments.

[1] http://www.lemurproject.org/clueweb09.php/

4.2. Baseline methods

We use the Indri[2] to conduct our retrieval run with its default parameter configuration. We generate three initial documents D_{init} by three weighting models: language model (LM) (Hiemstra, 2001), BM25 (Robertson et al., 1995) and TFIDF. We employ these model with its default suggestion parameter settings: b=0.75 in BM25, $\lambda_{LM} = 0.15$ in LM. All of the search result diversification methods are applied based on the top-K retrieved documents. As for the parameter K, we conduct a series of tests to finding the appropriate K maximizing the performance, which found 50 achieves the best result.

Besides the baseline retrieval model, we compare L-SRD with some advanced diversification models as follows:

- **QL**. The Query-likelihood language model is used for indri search engine as an initial retrieval method. We use it to provide the initial top 1000 documents for our diversification method. We also use it as a baseline method.

- **MMR**. A classical implicit diversification model. It is a representation of comparison between implicit and explicit models.

- **xQuAD**. xQuAD is a popular explicit diversification model which focus on the redundancy of aspects.

- **PM2**. PM2 is a popular explicit diversification model. PM2 generates the result set according to the aspects proportionality.

- **SVMDIV**. SVMDIV is also a learning model for search result diversification (Yue & Joachims, 2008). It use the structural SVMs to optimize the aspects coverage problem. But it only models the diversity without consideration of relevance. We get the source code from the svmdiv homepage[3] provided by the author.

[2]http://www.lemurproject.org/indri.php
[3]http://projects.yisongyue.com/svmdiv/

These baseline 2-4 (corresponding to MMR, xQuAD, PM2) possess a single parameter λ to tune, we perform a 5-fold cross validation to train λ through optimizing ERR-IA. In our model, we also use a 5-cross validation with a ratio of 3:1:1 for training, validation and prediction for the test query on each year. The final result are calculated over all the folds.

4.3. Experimental results

In particularly, our experiments aim to answer two mainly questions:

1. Can we utilize learning mechanism to promote the performance of search result diversification?
2. Does L-SRD achieve better performance in terms of robustness comparing other search result diversified models?

4.3.1. Diversification analysis

To answer question 1, we compare our L-SRD with other diversification models in terms of diversification metrics.

Table 1 shows the result of the evaluation in terms of α-nDCG, ERR-IA, and NRBP. The best result per baseline is highlighted in bold.

The classical MMR method (Carbonell & Goldstein, 1998) is used as a representative of implicit diversification model. As for explicit model, we consider xQuAD (Santos et al., 2010) and PM2 (Dang & Croft, 2012). The SVMDIV is selected for the representative of a learning method.

The result shows that L-SRD always performs best in terms of all the metrics. It consistently improves the initial retrieval ranking method with gains up to 17.78%, 27.44%, 12.55%, in terms of α-nDCG on WT2009, WT2010, WT2011 respectively. It indicates that our introducing learning approach tackle the diversity measurement problem more effectively with the consideration of integrate different features. The reason is that features such as query-aspects relevance and information richness conform to the property of diversity. Furthermore, comparing with the explicit diversification models in terms of the evaluation of α-nDCG, the improvement of L-SRD over the xQuAD is up to 18.80%,

13.47%, 12.35% on WT2009, WT2010, WT2011 respectively, and the improvement of L-SRD over the PM2 is up to 9.13%, 19.61%, 12.10% on WT2009, WT2010, WT2011 respectively. Previous explicit diversification use a predefined function to calculate the diversity score, which cannot reach a optimal result from the overall situation. Our proposing a learnable approach to solve the diversity measurement and parameter tuning problem is significative. Besides the non-learning model, the improvement of L-SRD over the SVMDIV is up to 10.18%, 14.70%, 11.01% on WT2009, WT2010, WT2011 respectively. It shows that considering relevance and different types of features in diversity measurement is helpful in learning approach. That is the reason why our model wins. Therefore, our L-SRD shows better understanding on the diverse ranking and leads to a better result. So our utilizing learning mechanism indeed promotes the performance of search result diversification.

We consider not only the advanced diversity metrics, but also traditional diversity metrics, such as Precision-IA and Aspect Recall. The former indicates how many relevant documents for each aspects we have for reranking, the latter indicates how many of the aspects for which we have relevant documents. The result is shown in Fig. 3. MMR still underperform all of them, as for Precision-IA, xQuAD wins on WT2010 casually, while our L-SRD performs more stable, even on WT2010, the gap is small. This proves that our L-SRD model outperform others from different perspective. Our learnable model solves the diverse ranking problem in a global perspective and always reach prominent result.

4.3.2. Robustness analysis

For the robustness analysis which we raise question 2, we set up series of experiments on robustness research to study the Win/Loss behaviour and the usage of different retrieval algorithms respectively.

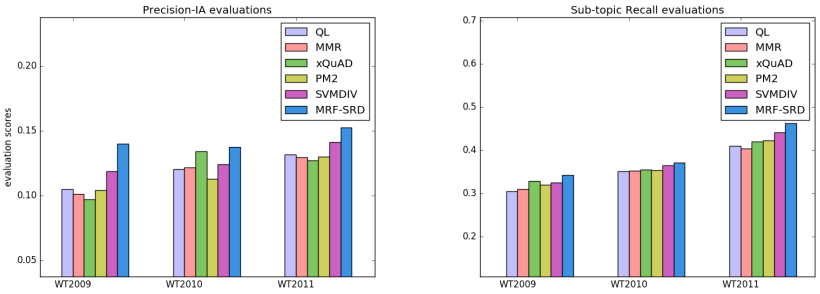
Yue et al. (Yue & Joachims, 2008) and Dang et al. (Dang & Croft, 2012) mention an indicator for entirety robustness measurement, which is the Win/Loss ratio. The Win/Loss ratio denotes whether the model improve or hurt the result when comparing with the basic relevant baseline QL (Yue & Joachims,

Year	experiment	ERR-IA@20	α-nDCG@20	NRBP
2009	QL	0.1376	0.2548	0.1008
	MMR	0.1405	0.2526	0.1070
	xQuAD	0.1411	0.2444	0.1113
	PM2	0.1482	0.2750	0.1101
	SVMDIV	0.1531	0.2849	0.1219
	L-SRD	**0.1862**	**0.3139**	**0.1615**
2010	QL	0.1484	0.2445	0.1092
	MMR	0.1494	0.2450	0.1129
	xQuAD	0.1732	0.2746	0.1326
	PM2	0.1599	0.2605	0.1175
	SVMDIV	0.1698	0.2796	0.1158
	L-SRD	**0.2193**	**0.3207**	**0.1826**
2011	QL	0.3288	0.4454	0.2802
	MMR	0.3253	0.4337	0.2834
	xQuAD	0.3235	0.4462	0.2812
	PM2	0.3316	0.4472	0.2831
	SVMDIV	0.3429	0.4615	0.2923
	L-SRD	**0.4078**	**0.5127**	**0.3374**

Table 1: Diversification performance using the official evaluation metrics for WT2009, WT2010, WT2011

2008; Dang & Croft, 2012) in terms of evaluation metrics. Particularly in our experiment, we use ERR-IA to calculate the Win/Loss ratio.

From table 2, we found that L-SRD model performs best with its ratio of 2.65. While the Win/Loss ratios of MMR, xQuAD, PM2 and SVMDIV are 1.31, 1.45, 1.52, 1.77 respectively. It reflects the remarkable robustness of L-SRD model comparing with other outstanding diversification models. The promotion of robutsness over the MMR, xQuAD, PM2 and SVMDIV is up to 101.49%, 81.51%, 74.05%, 49.43% respectively. And it confirms the overall



(a) Precicison-IA evaluation

(b) Aspect Recall evaluation

Figure 3: Performance comparison in WT2009, WT2010, WT2011 with Precision-IA and Aspect Recall

experiment	WT2009	WT2010	WT2011
MMR	20/18	24/17	23/16
xQuAD	23/18	23/16	24/14
PM2	22/20	26/14	25/14
SVMDIV	24/18	27/13	27/13
L-SRD	**28/14**	**30/10**	**32/10**

Table 2: Win/Loss ratio

performance of our model is not restricted to a small subset, it still works in the whole dataset for three years data. Our different types of features and learning approach addresses this problem well enough.

Many retrieval algorithm can be used for providing search results as original input. As a concern for robustness, we are looking forward to investigate whether different retrieval algorithms affect the final result of our model. In addition, we compare L-SRD with other search result diversification models by using different retrieval algorithms. We evaluate the effectiveness of L-SRD on diversifying the search results produced by three effective probabilistic document weighting models: BM25, TF-IDF, and Hiemstra's language modelling (LM). In particular, we employ the parameter setting as discussed in Section

	α-nDCG			ERR-IA		
	@5	@10	@20	@5	@10	@20
BM25	0.160	0.211	0.255	0.107	0.127	0.138
+MMR	0.188	0.216	0.256	0.124	0.136	0.145
+xQuAD	0.183	0.221	0.263	0.115	0.132	0.142
+PM2	0.174	0.209	0.252	0.108	0.124	0.135
+SVMDIV	0.193	0.246	0.285	0.128	0.141	0.153
+L-SRD	**0.242**	**0.268**	**0.314**	**0.161**	**0.174**	**0.186**
TFIDF	0.158	0.207	0.251	0.102	0.121	0.131
+MMR	0.173	0.215	0.259	0.118	0.126	0.135
+xQuAD	0.181	0.217	0.257	0.113	0.130	0.137
+PM2	0.189	0.223	0.263	0.117	0.133	0.142
+SVMDIV	0.187	0.238	0.279	0.119	0.136	0.148
+L-SRD	**0.237**	**0.255**	**0.302**	**0.151**	**0.169**	**0.180**
LM	0.095	0.147	0.195	0.054	0.074	0.085
+MMR	0.108	0.145	0.187	0.060	0.074	0.084
+xQuAD	0.099	0.159	0.199	0.056	0.078	0.088
+PM2	0.094	0.149	0.196	0.054	0.076	0.087
+SVMDIV	0.100	0.161	0.207	0.060	0.081	0.091
+L-SRD	**0.137**	**0.182**	**0.223**	**0.084**	**0.102**	**0.111**

Table 3: Results based on different retrieval models (as for WT2009)

4.2. These weighting models are used to produce both a baseline ranking as well as the sub-rankings for different sub-queries. Additionally, we compare its performance to that of three diversification baselines.

Table 3 shows the results of this robustness evaluation in terms of α-NDCG and ERR-IA. Specially, L-SRD model is the only approach continuously improving the evaluation result with respect to the baseline ranking method. Our learnable model tackles the diversification problem with a global angle and adjust the weight of parameter million times, so it performs more stable.

5. Conclusion and future work

In this paper, we propose a Learnable Search Result Diversification model (L-SRD). We pay our attention to the explicit Query Aspect Diversification models and introduce the learning approach to regard the model as a learning problem. Unlike the traditional explicit diversification models utilizing a predefined novelty measure function (Santos et al., 2010; Dang & Croft, 2012), we integrate different types of diversity features and estimate the weight with the learning approach. We derive our loss function as the likelihood of ground truth generation by the virtue of the Plackett-Luce model(Marden, 1996). Stochastic gradient descent algorithms are used to estimate the value of parameter. Benefit from the learning approach, we can optimize the parameter in an automatic method. The prediction of our diversification model is provided by iteratively maximizing the learned ranking function.

We have demonstrated the improvement of L-SRD comparing with other diversification models. We find our model achieve considerable results in terms of official diversity metrics on three years in TREC web track dataset. To prove its robustness, we set the experiment about Win/Loss ratio and usage of different retrieval algorithms. We believe L-SRD will play an important role to improve the Query Aspect Search Result Diversification by using a learning method.

There exist a number of directions to be explored in the future. We are looking forward to take some considerable steps to make L-SRD achieve convergency as quick as possible. Meanwhile, we desire to employ the learning approach on the hierarchical aspects underlying the query.

Ackonwloedgements

This research is supported by National Natural Science Foundation of China (Grant No. 61375054), Natural Science Foundation of Guangdong Province Grant No. 2014A030313745, Basic Scientific Research Program of Shenzhen

City (Grant No. JCYJ20160331184440545), and Cross fund of Graduate School at Shenzhen, Tsinghua University (Grant No. JC20140001).

reference

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining* (pp. 5–14). ACM.

Capannini, G., Nardini, F. M., Perego, R., & Silvestri, F. (2011). Efficient diversification of search results using query logs. In *Proceedings of the 20th international conference companion on World wide web* (pp. 17–18). ACM.

Carbonell, J., & Goldstein, J. (1998). The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 335–336). ACM.

Chapelle, O., Metlzer, D., Zhang, Y., & Grinspan, P. (2009). Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 621–630). ACM.

Clarke, C. L., Craswell, N., & Soboroff, I. (2009a). Preliminary report on the trec 2009 web track. In *Proc. of TREC*.

Clarke, C. L., Kolla, M., Cormack, G. V., Vechtomova, O., Ashkan, A., Büttcher, S., & MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 659–666). ACM.

Clarke, C. L., Kolla, M., & Vechtomova, O. (2009b). *An effectiveness measure for ambiguous and underspecified queries*. Springer.

Dang, V., & Croft, B. W. (2013). Term level search result diversification. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* (pp. 603–612). ACM.

Dang, V., & Croft, W. B. (2012). Diversity by proportionality: an election-based approach to search result diversification. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 65–74). ACM.

Dou, Z., Hu, S., Chen, K., Song, R., & Wen, J.-R. (2011). Multi-dimensional search result diversification. In *Proceedings of the fourth ACM international conference on Web search and data mining* (pp. 475–484). ACM.

Drosou, M., & Pitoura, E. (2010). Search result diversification. *ACM SIGMOD Record*, *39*, 41–47.

Fetterly, D., Manasse, M., & Najork, M. (2004). Spam, damn spam, and statistics: Using statistical analysis to locate spam web pages. In *Proceedings of the 7th International Workshop on the Web and Databases: colocated with ACM SIGMOD/PODS 2004* (pp. 1–6). ACM.

He, J., Hollink, V., & de Vries, A. (2012). Combining implicit and explicit topic representations for result diversification. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 851–860). ACM.

Hiemstra, D. (2001). *Using language models for information retrieval*. Taaluitgeverij Neslia Paniculata.

Jansen, B. J., Spink, A., Bateman, J., & Saracevic, T. (1998). Real life information retrieval: A study of user queries on the web. In *ACM SIGIR Forum* (pp. 5–17). ACM volume 32.

Kurland, O., & Domshlak, C. (2008). A rank-aggregation approach to searching for optimal query-specific clusters. In *Proceedings of the 31st annual interna-*

tional ACM SIGIR conference on Research and development in information retrieval (pp. 547–554). ACM.

Liang, S., Ren, Z., & de Rijke, M. (2014). Fusion helps diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (pp. 303–312). ACM.

Marden, J. I. (1996). *Analyzing and modeling rank data*. CRC Press.

Radlinski, F., Kleinberg, R., & Joachims, T. (2008). Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th international conference on Machine learning* (pp. 784–791). ACM.

Raman, K., Bennett, P. N., & Collins-Thompson, K. (2013). Toward whole-session relevance: exploring intrinsic diversity in web search. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* (pp. 463–472). ACM.

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M. M., Gatford, M. et al. (1995). Okapi at trec-3. *NIST SPECIAL PUBLICATION SP*, (pp. 109–109).

Santos, R. L., Macdonald, C., & Ounis, I. (2010). Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th international conference on World wide web* (pp. 881–890). ACM.

Vargas, S., Castells, P., & Vallet, D. (2012). Explicit relevance models in intent-oriented information retrieval diversification. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 75–84). ACM.

Yu, H.-T., & Ren, F. (2014). Search result diversification via filling up multiple knapsacks. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management* (pp. 609–618). ACM.

Yue, Y., & Joachims, T. (2008). Predicting diverse subsets using structural svms. In *Proceedings of the 25th international conference on Machine learning* (pp. 1224–1231). ACM.

Zhai, C. X., Cohen, W. W., & Lafferty, J. (2003). Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval* (pp. 10–17). ACM.

Zhu, Y., Lan, Y., Guo, J., Cheng, X., & Niu, S. (2014). Learning for search result diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (pp. 293–302). ACM.

Research Highlights

- We Introduce the learning mechanism to the Query Aspect Diversification model.
- We Utilize the Markov Random Field to integrate different types of features.
- We conduct extensive experiments to verify that our model achieve better performance.

LaTeX Source Files

Click here to download LaTeX Source Files: Source Files.zip