# 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 problems 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 algorithm is employed to optimize the values of parameters. Finally we derive our ranking function to generate the diverse list sequentially. Due to the learning model, the values of parameters are determined automatically and get optimally. The experiments on TREC web tracks show that our approach outperforms several existing diversification models significantly.

*Keywords:* Explicit Search Result Diversification, 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, 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 aspects 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 well (Agrawal et al. (2009); Zhai et al. (2003)). The explicit approaches are proposed to overcome the weakness. They explicitly formalize the aspects underlying a query and select documents that cover different aspects. The xQuAD and PM2 are classic explicit models (Santos et al. (2010a); Dang & Croft (2012)). But they just utilize a predefined function to calculate the diversity score based on query aspects. It is subjective and hard to reach optimal result.

In this paper, we treat the Query Aspect Diversification as a learning problem and propose a Learnable Search Result Diversification (L-SRD) method. We incorporate various features into diversity measurement based on the Markov Random Field (MRF), which enables the integration of various types of features. The values 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 algorithm is employed to optimize the values of weights. Finally we derive our ranking function to generate the diverse list sequentially.

We conduct a series of experiments to demonstrate that L-SRD is more effective than other diversification models in terms of the official evaluation metrics including $\alpha$-nDCG, ERR-IA, NRBP and the classical diversification metrics such as Precision-IA and Aspect Recall (Clarke et al. (2008); Chapelle et al. (2009); Clarke et al. (2009b)). 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 parameters 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 L-SRD achieve better performance comparing with the existing diversification models.

The remainder of this paper is organized 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 and 4.3 detail 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

Search result diversification can be characterized as a bidirectional optimization problem, in which one seeks to maximize the overall relevance of a document ranking to multiple query aspects, while minimizing its redundancy (Santos et al. (2010b)). In particular, the existing approaches can be categorized as either implicit or explicit making a difference in how they account for the query aspects (Santos et al. (2010c)).

The basic assumption of implicit diversification approaches is that dissimilar documents are more likely to satisfy different information needs. The most representative approach in maximal marginal relevance (MMR) method and its probabilistic variants is shown as follows (Zhai et al. (2003)):

$$S_{MMR}(q,d,c) = (1-\lambda)S^{rel}(d,q) - \lambda \underset{d_j \in C}{max} S^{div}(d,d_j) \tag{1}$$

where $S^{rel}$and $S^{div}$ represents document $d's$ relevance to the query $q$ and its similarity to a selected document $d_j$ respectively. To gain high ranking score, a document should not only be relevant, but also be dissimilar from the selected documents. The special process of MMR proposed by Garbonell & Goldstein is selecting the document iteratively Carbonell & Goldstein (1998), and meanwhile, both content-based relevance and diversity relation between current selected document and the previously selected documents are considered. Yu et al. formulate this as a process of selecting and ranking $k$ exemplar documents and utilize linear programming to solve this problem (Yu et al. (2017)). In summary, they are all implicit approaches without using aspects to mine the underlying aspects, besides, they are a low effective approaches (Santos et al. (2010a); 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. (2010a) and RxQuAD Vargas et al. (2012) are proposed to reduce redundancy on the aspect levels. These methods select documents that cover more novel aspects. The PM-1 and PM-2 models pay more attention to maintain the proportionality of aspects Dang & Croft (2012). 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)). Wang et al. (2016); Hu & et al. (2015) think the aspects underlying the query should be hierarchical, and propose some hierarchical measures to find the relationships among aspects. To conclude, all existing explicit approaches are unsupervised, and the values of parameters need to be tuned by the experiment repeatedly without intention, causing a time-consuming optimizing problem to find the most suitable parameters.

Our approach also accounts for the aspects of a query in an explicit way. However, differently from the existing approaches, we use a learnable process to identify features from documents using Markov Random Field. It is other than using structural SVM to learn to identify a document subset with maximum word coverage (Zhu et al. (2014)). They just learn the maximum word coverage and do not mine the aspects underlying the query. Besides, we redefine the diversity function and derive our loss function as the likelihood loss of ground truth generation to resolve this bicriterion optimization problem.

## 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. (2010a)), 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 is a greedy approximation. It sequentially selects the "local-best" document from the candidate document set (Santos et al. (2010a)). The original function is formalized as follows:

$$f(d, \bar{S}) = (1-\lambda)P(d|q) + \lambda \sum_{q_i \in Q} P(q_i|q)P(d|q_i)P(\bar{S}|q_i). \tag{2}$$

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, $\lambda$ is a balance parameter for a trade-off between relevance and diversity, $q_i$ denotes for the aspects underlying the query $q$.

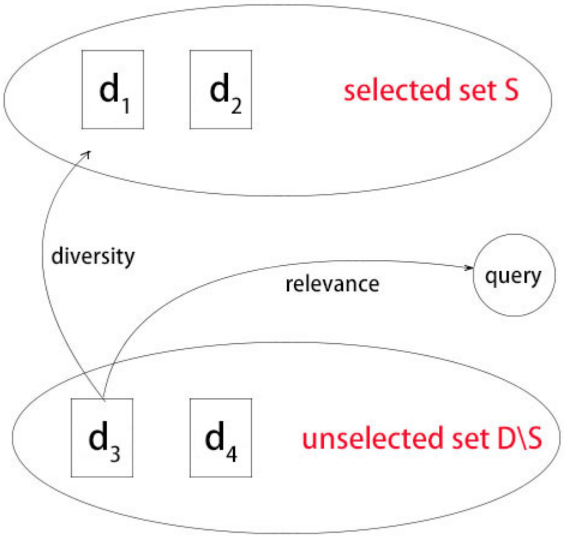


Figure 1: An illustration for sequential selection in topic diversity model

As for Eq. (2), the left part corresponds to the relevance score and the right part corresponds to the diversity score. We look 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:

$$P(\bar{S}|q_i) = \frac{P(\bar{S}, q_i)}{P(q_i)} \overset{rank}{=} P(\bar{S}, q_i) \tag{3}$$

where $P(q_i)$ denotes the occurrence rate of aspects $q_i$ corresponding to query $q$, which is usually regard to be normalized as $1/n$ ($n$ denotes the number of aspects) (Santos et al. (2010a)). Because the values 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) to model $P(\bar{S}, q_i)$. We 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 an undirected graph $G$. In 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:

$$P(\bar{S}, q_i) = \frac{\prod_{l \in L(G)} \phi_l(l)}{Z}, \tag{4}$$

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:

$$\phi_l(l) = exp(\lambda_l f_l(l)), \tag{5}$$

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

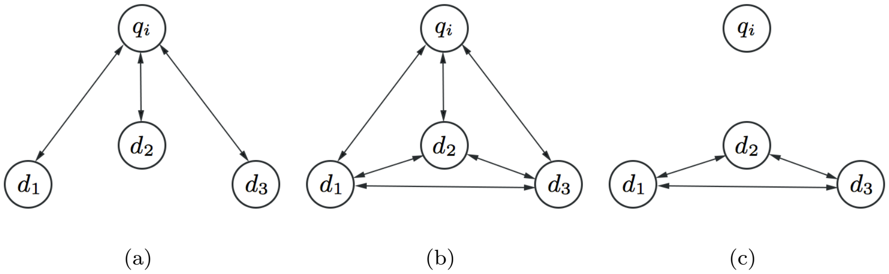
$$P(\bar{S}, q_i) \overset{rank}{=} \sum_{l \in L(G)} \lambda_l f_l(l). \tag{6}$$

Note that Eq. (6) is derived from Eq. (4) 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 MRF. 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.



(a) (b) (c)

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 $\bar{S}$: $P_{entropy}(d) \overset{def}{=} -\sum_{w \in d} P(w|d) \log p(w|d)$ 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 is used for feature function, too (Fetterly et al. (2004)).

To conclude, by replacing the feature function into equation (2) and putting the parameters $\lambda$ into the learning process, the ranking function is given as follows:

$$f(d, \bar{S}) = \lambda_r P(d|q) + \lambda_n \sum_{q_i \in Q} [P(q_i|q)P(d|q_i) \sum_{l \in L(G(q_i))} \lambda_l f_l(l)] \tag{7}$$

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 $\lambda$ in equation (2), it is a balance parameter between relevance and diversity. In our learning method, we use $\lambda_r$ and $\lambda_n$ 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:

$$L(rank(D, C), Y) = -\log P(Y|D), \tag{8}$$

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:

$$\begin{aligned} P(Y|D) &= P(y_1, y_2, ..., y_n|D) \\ &= P(y_1|D)P(y_2|D \backslash S_1) \cdots P(y_n|D \backslash S_{n-1}), \end{aligned} \tag{9}$$

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:

$$P(Y|D) = \prod_{i=1}^{n} P(y_i|D \backslash S_{i-1}) = \prod_{i=1}^{n} \frac{exp(f(y_i, D \backslash S_{i-1}))}{\sum_{k=i}^{n} exp(f(y_k, D \backslash S_{i-1}))}, \tag{10}$$

where $S_0$ means empty set $\emptyset$, function $f$ corresponds to Eq. (7). Incorporating Eq.(10) into Eq.(8), we get the definition of the loss function as follows:

$$L(f(D,C),Y) = -\sum_{i=1}^{n} \log(\frac{exp(f(y_i, D\backslash S_{i-1}))}{\sum_{k=i}^{n} exp(f(y_k, D\backslash S_{i-1}))}). \tag{11}$$

To get the final loss function, we simplify Eq. (7) by uniting the parameter $\lambda_n$ and $\lambda_l$ (because the parameter in our model all can be decided by the learning process):

$$f(d,\bar{S}) = \lambda_r P(d|q) + \vec{\mu_d} \cdot N_{1\ldots L}(d, q_i, \bar{S}) \tag{12}$$

$$N_l(d, q_i, \bar{S}) = \sum_{q_i \in Q} P(q_i|q)P(d|q_i)f_l(l) \quad (l \in L(G(q_i))) \tag{13}$$

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:

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

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. $\alpha$-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 $\alpha$-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: $y_1, y_2, ..., y_n$
initialize $S_0 = \emptyset$
for $k = 1$ to $n$ do
    $bestDoc \leftarrow \arg\max_{d \in RD^{(i)} \backslash S_{k-1}} f_{ERR-IA}(d \cup S_{k-1})$
    $S_k = S_{k-1} \cup bestDoc$
    $y_k = bestDoc$
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. (15)-(16) and update the value of weight. The gradient in step $i$ at training set $D_{init}$ is computed as follows:

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

$$\Delta\vec{\mu_d}^{(i)} = \sum_{j=1}^{n}(\frac{\sum_{k=j}^{n} N_l(y_j,q_i,\bar{S})exp(\lambda_r^{(i-1)}P(y_j|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_j,q_i,\bar{S}))}{\sum_{k=j}^{n} exp(\lambda_r^{(i-1)}P(y_k|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_k,q_i,\bar{S}))} - \frac{N_l(y_j,q_i,\bar{S})exp(\lambda_r^{(i-1)}P(y_j|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_j,q_i,\bar{S}))}{exp(\lambda_r^{(i-1)}P(y_k|q)+\vec{\mu_d}^{(i-1)}\cdot N_{1...L}(y_k,q_i,\bar{S}))}) \tag{16}$$

**ALGORITHM 2:** Parameter learning algorithm

```
Input: training data: $D_{init}^{Tr}, (y_1...y_n)^{Tr}$
    parameter: learning rate $\eta$, tolerate $\epsilon$
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 $\epsilon$
```

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 $\lambda_r$ and $\lambda_{nl}$
Output: $y_1, y_2, ..., y_n$
$S_0 = \emptyset$
for $i = 1, ..., n$ do
    $best = \arg\max_{d \in D \backslash S_{i-1}} f(d, D \backslash S_{i-1})$
    $S_i = S_{i-1} \cup best$
    $y_i = best$
end for
```

## 4. Experiment

### *4.1. Dataset and evaluation metrics*

There are 150 queries in our query set, 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: $\alpha$-nDCG (Clarke et al. (2008)), ERR-IA (Chapelle et al. (2009)), NRBP (Clarke et al. (2009b)). $\alpha$-nDCG is used to balance both relevance and diversity of candidate documents. ERR-IA measures the expected effort required for a user to satisfy their information needs. And NRBP is a feasible metric to evaluate the balance between the complexity of needs and the query. 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 and Aspect Recall too. To measure the robustness, we use Win/Loss ratio metrics (Yue & Joachims (2008); Dang & Croft (2012)). The Win/Loss ratio denotes whether the model improve or hurt the result when

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

comparing with the basic relevant baseline QL in terms of evaluation metrics (Dang & Croft (2012)). Particularly in our experiment, we use ERR-IA to calculate the Win/Loss ratio.

The evaluation metrics are reported at different cutoffs. We use {5, 10, 20} as our cutoffs to set up experiments. These cutoffs focus on the evaluation at early ranking, which are particularly important in a Web search context (Jansen et al. (1998)). The $\alpha$ is set to 0.5 in our experiments for the reason it gives equal weight to both relevance and diversity.

### *4.2. Baseline methods*

We use the Indri[2] to conduct our retrieval experiments and run with its default parameters configuration. 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 find the appropriate $K$ maximizing the performance, which found 50 achieves the best result.

Besides the baseline retrieval models, 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 (Dang & Croft (2012)).

- **MMR**. A classical implicit diversification model. It is a representation of comparison between implicit and explicit models(Carbonell & Goldstein (1998)).

- **xQuAD**. xQuAD is a popular explicit diversification model which focuses on the redundancy of aspects (Santos et al. (2010a)).

- **PM2**. PM2 is a popular explicit diversification model. PM2 generates the result set according to the aspects proportionality (Dang & Croft (2012)).

[2] http://www.lemurproject.org/indri.php

- **SVMDIV**. SVMDIV is also a learning model for search result diversification (Yue & Joachims (2008)). It uses 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.
- **HxQuAD** is a hierarchical diversification model based on xQuAD (Hu & et al. (2015)).

These baselines 2-4 (corresponding to MMR, xQuAD, PM2) possess a single parameter $\lambda$ to tune, we perform a 5-fold cross validation to train $\lambda$ through optimizing ERR-IA. In our model, we 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 results are calculated over all the folds.

### *4.3. Experimental results*

In particularly, our experiments aim to answer two main 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 to 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 $\alpha$-nDCG, ERR-IA, and NRBP. The best result per line is highlighted in bold. The classical MMR method is used as a representative of implicit diversification model (Carbonell & Goldstein (1998)). As for explicit model, we consider xQuAD and PM2 (Santos et al. (2010a); Dang & Croft (2012)). SVMDIV is selected for the representative of learning methods, HxQuAD is selected for the hierarchical model.

[3] http://projects.yisongyue.com/svmdiv/

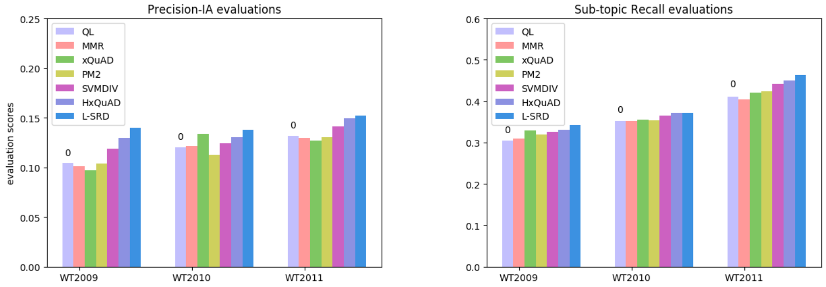
The result shows that L-SRD always performs best in terms of all metrics. It consistently improves the initial retrieval ranking method with gains up to 23.19%, 31.17%, 15.11% in terms of $\alpha$-nDCG on WT2009, WT2010, WT2011 respectively. It indicates that our learning approach tackles the diversity 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 $\alpha$-nDCG, the improvement of L-SRD over the xQuAD is up to 28.44%, 16.87%, 14.90% on WT2009, WT2010, WT2011 respectively, and the improvement of L-SRD over the PM2 is up to 14.15%, 23.11%, 14.65% on WT2009, WT2010, WT2011 respectively. Previous explicit diversifications use a predefined function to calculate the diversity score, which cannot reach an optimal result from the overall situation. A learnable approach to solve the diversity measurement and parameter tuning problem is significative. In addition, comparing with the hierarchical diversification model in terms of the evaluation of $\alpha$-nDCG, the improvement of L-SRD over the HxQuAD is up to 3.77%, 9.68%, 5.49% on WT2009, WT2010, WT2011 respectively. HxQuAD only use a predefined function to measure the diversity score, and the parameters may not be optimal because it needs to be tuned manually. Our learning model tackles the parameters tuning problem in an automatic fashion and reaches optimal result. Besides the non-learning model, the improvement of L-SRD over the SVMDIV is up to 10.18%, 14.70%, 11.09% on WT2009, WT2010, WT2011 respectively. It shows that considering relevance and different types of features in diversity measurement is helpful in the learning approach. That is the reason why our model wins. Therefore, L-SRD shows better understanding on the diverse ranking and leads to a better result. So 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 aspect we have for reranking, the latter

| Year | experiment | ERR-IA@20 | $\alpha$-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 |
| | HxQuAD | 0.1653 | 0.3025 | 0.1372 |
| | 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 |
| | HxQuAD | 0.1807 | 0.2924 | 0.1303 |
| | 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 |
| | HxQuAD | 0.3606 | 0.4860 | 0.3107 |
| | L-SRD | **0.4078** | **0.5127** | **0.3374** |

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

indicates how many of the aspects for which we have relevant documents. The result is shown in Fig. 3. MMR still underperforms all of them, as for Precision-IA, xQuAD wins on WT2010 casually, while L-SRD performs more stable, even on WT2010, the gap is small. It proves that L-SRD outperforms others from different perspectives. Our learnable model solves the diverse ranking problem in a global perspective and always reaches prominent results.



(a) Precicison-IA evaluation

(b) Aspect Recall evaluation

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

### *4.3.2. Robustness analysis*

An effective search result diversification method should not only outperform other models in terms of diversity metrics, but also maintain a high level of robustness, which we have raised the 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.

| 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 |
| HxQuAD | 27/15 | 30/10 | 31/10 |
| L-SRD | **28/14** | **30/10** | **32/10** |

Table 2: Win/Loss ratio

From table 2, we find that L-SRD model performs best with its ratio of 2.65. While the Win/Loss ratio of MMR, xQuAD, PM2, SVMDIV and HxQuAD is

1.31, 1.46, 1.52, 1.77, 2.51 respectively. It reflects the remarkable robustness of L-SRD model comparing with other outstanding diversification models. The promotion of robustness over the MMR, xQuAD, PM2 , SVMDIV and HxQuAD is up to 101.51%, 81.51%, 74.34%, 49.72%, 42.06 % respectively. And it confirms the overall 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 address this problem well.

## 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, we integrate different types of diversity features and estimate the weight with a learning approach. We derive our loss function as the likelihood of ground truth generation. Stochastic gradient descent algorithm is used to estimate the values of parameters. Benefiting from the learning approach, we can optimize the parameters in an automatic method. The prediction of our diversification model is provided by iterative maximizing the learned ranking function.

We have demonstrated the improvement of L-SRD comparing with other diversification models. We find L-SRD 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 look forward to take some considerable steps to make L-SRD achieve convergency as quick as possible.

## 6. Acknowledgments

This research is supported by National Natural Science Foundation of China (Grant No. 61773229), Shenzhen Science and Technology Project (Grant No. CYZZ20150408152315667), Basic Scientific Research Program of Shenzhen City (Grant No. JCYJ20160331184440545), and Cross fund of Graduate School at Shenzhen, Tsinghua University (Grant No. JC20140001).

## References

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.

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, 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.

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.

Hu, S., & et al. (2015). Search result diversification based on hierarchical intents.

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

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

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.

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

Santos, R. L., Macdonald, C., & Ounis, I. (2010b). Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th International Conference on World Wide Web* WWW '10 (pp. 881–890). New York, NY, USA: ACM. URL: `http://doi.acm.org/10.1145/1772690.1772780`. doi:`10.1145/1772690.1772780`.

Santos, R. L. T., Peng, J., Macdonald, C., & Ounis, I. (2010c). Explicit search result diversification through sub-queries. In C. Gurrin, Y. He, G. Kazai, U. Kruschwitz, S. Little, T. Roelleke, S. Rüger, & K. van Rijsbergen (Eds.), *Advances in Information Retrieval* (pp. 87–99). Berlin, Heidelberg: Springer Berlin Heidelberg.

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.

Wang, X., Dou, Z., Sakai, T., & Wen, J.-R. (2016). Evaluating search result diversity using intent hierarchies. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval* SIGIR '16 (pp. 415–424). New York, NY, USA: ACM. URL: `http://doi.acm.org/10.1145/2911451.2911497`. doi:`10.1145/2911451.2911497`.

Yu, H.-T., Jatowt, A., Blanco, R., Joho, H., Jose, J., Chen, L., & Yuan, F. (2017). A concise integer linear programming formulation for implicit search result diversification. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining* WSDM '17 (pp. 191–200). New York, NY, USA: ACM. URL: `http://doi.acm.org/10.1145/3018661.3018710`. doi:`10.1145/3018661.3018710`.

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.