A Learning Approach to Hierarchical Search Result Diversification

Hai-Tao Zheng [⋆], Zhuren Wang, and Xi Xiao

Tsinghua-Southampton Web Science Laboratory,
Graduate School at Shenzhen, Tsinghua University, China
zheng.haitao@sz.tsinghua.edu.cn
wang-zr14@mails.tsinghua.edu.cn
xiaox@sz.tsinghua.edu.cn

Abstract. The queries in search engine that issued by users are often ambiguous. By returning diverse ranking results we can satisfy different information needs as far as possible. Recently, a hierarchical structure are proposed to represent user intents instead of a flat list of subtopics. Although the hierarchical diversification model performs better than previous models, it utilizes a predefined function to calculate the diversity score, which may not reach the optimal result. The model's parameters need to be tuned manually and repeatedly without intention, which cause a time-consuming problem. In this paper, we introduce a learning based hierarchical diversification model. Benefit from the learning model, the parameter values are determined automatically and more optimal. Experiments show that our approach outperform several existing diversification models significantly.

1 Introduction

Search result diversification [2, 1, 11, 10] is an effective way to solve the query ambiguation problem. The diversification model regards the problem as a combination of relevance score and diversity score. Diversity score are often calculated based on the user intents. Previous models regard the user intents as a flat list of subtopics. However, the flat list of subtopics cannot match the actual intents in evaluation tasks [4, 8] good enough. The hierarchical structure of subtopics are proposed to solve this problem.

The hierarchical diversification methods perform better than previous diversification methods, but it just utilizes a predefined function to calculate the diversity score. There exists some parameters to tune. Usually researchers have to set a series of repeatedly experiments to find the suitable parameter. It is hard to reach optimal value by manual tuning, and it causes a time-consuming problem.

In this paper, we introduce a learning approach to hierarchical search result diversification called L-HSRD. Firstly, we redefine the loss function as the generation probability of sequential selection for a ground truth list. Then Stochastic

[⋆] corresponding author

gradient descent are employed to optimize the value of weight. Finally we derive our ranking function to generate the diverse list sequentially.

We demonstrate L-HSRD is more excellent than other diversification models in terms of official evaluation metrics including α-NDCG [6], ERR-IA [3] and NRBP [7]. Additionally, we conduct a series of experiments to illustrate the robustness of our method, which get a outstanding performance.

The main contributions of our work are listed as follows:

1. L-HSRD is the first method introducing the learning mechanism for the hierarchical search result diversification. We conduct inference for the loss function based on its sequential selection model, which solves the parameter tuning problem at the same time.
2. We put forward a series of instructive different features based on the hierarchical structure of subtopics.
3. We conduct extensive experiments to verify that L-HSRD achieves excellent performance comparing with the existing diversification models.

2 Related Work

Search result diversification model can be categorized as implicit approaches and explicit approaches. The implicit approaches includes MMR [2]. MMR selecs the document iteratively, and meanwhile, both content-based relevance and diversity should be considered. It is considered as a low effective model [14, 11].

Explicit approaches extract the aspects explicitly and make use of them to calculate the diversity score. The algorithms such as IA-select [1], xQuAD [14] and RxQuAD [15] are proposed to reduce redundancy. These methods select the document that covering more novel aspects. The PM-1 and PM-2 [9] models mainly consider the proportionality of aspects and produce the diverse result by virtue of the proportionality of aspects.

In addition, as for learning model, Zhu et al. [17] proposed a learning model without considering the aspects underlying the query. Yue et al. [16] proposed Structural SVMs to model the diversity but discarded the relevance.

3 Learning approach to hierarchical search result diversification

3.1 Definition of ranking function

At first, We generate the subtopics like Hu[12] and get the relevance score $P(q_i|d)$ between the subtopic q_i and each document d. In our model, the ranking function we select the "local-best" document in the round i is given as follows:

$$f_i(d, D\backslash S_{i-1}) = \lambda_r P(d|q) + \lambda_d^T F(S_{i-1} \cup d), \tag{1}$$

where d means current document to be considered in the sequential selection process, D denotes the candidate document set, S_{i-1} denotes the documents already selected in previous $i-1$ round, q denotes the query, $F(S_{i-1} \cup d)$ stands for the feature function, represented by the feature vector of $(f_1,\ f_2,\ ...,\ f_T)$, T stands for the number of features.

Feature definition We define our features on the document set $S_{i-1} \cup d$. In our work, we provide several representative features for the learning process, which are shown as follow:

- **Features defined between the query and first-level subtopics**. Average, minimum, maximum, standard deviation of relevance score $P(q_i|d)$ for a document d to a subtopic q_i from level 1.
- **Features defined between the first-level subtopics and second-level subtopics**. Average, minimum, maximum, standard deviation of relevance score $P(q_i|d)$ for a document d to a subtopic q_i from level 2.
- **Features defined only on the first-level subtopics**. Firstly we define the set of documents which possess a highest score of relevance for first-level subtopics as St_1. (all the documents possess a relevance score for all the subtopics in level 1 and level 2). The features are defined as the entropy of all the documents d in St_1: $P_{entropy}(d) \overset{def}{=} -\sum_{w \in d} P(w|d) \log P(w|d)$, where w is a term and $p(w|d)$ is the probability that w appears in d (given by the language model).
- **Features defined only on the second-level subtopics**. The features are defined as the entropy of all the documents d in St_2 like above.

3.2 Definition of loss function

The ranking process is a sequential selection, we define the loss function as the likelihood loss of the generation probability:

$$L(f(X,C),Y) = -\log P(Y|D) = -\log[P(y(1)|D)P(y(2)|D\backslash S_1)...P(y(n)|D\backslash S_{n-1})], \tag{2}$$

where X stands for the feature, C represents the weight, the Y is the final result, $y(1),...,y(n)$ is the ground truth, n represents the top n result, the index i denotes its ranking position, D is the initial retrieved documents D_{init}, S_{i-1} denotes the result set we had iteratively selected in the last $i-1$ round, the probability $P(y(i)|D\backslash S_{i-1})$ represents the probability that selecting the document $y(i)$ under the condition of $D\backslash S_{i-1}$.

The above sequential definition approach can be well captured by the Plackett-Luce Model [13]. We can derive every step in our generation process, which is shown as:

$$P(Y|D) = \prod_{i=1}^{n} P(y(i)|D\backslash S_{i-1}) = \prod_{i=1}^{n} \frac{exp(f_i(y(i), D\backslash S_{i-1}))}{\sum_{k=i}^{n} exp(f_i(y(k), D\backslash S_{i-1}))}, \tag{3}$$

Given the training data $\{(X, C, Y)^{(Tr)}\}$ (Tr denotes for the number of training samples), the total loss function is formulized as follows:

$$L(f(X,C),Y) = -\sum_{i=1}^{T_r}\sum_{j=1}^{n}\log(\frac{exp(\lambda_r P(y(j)|q) + \lambda_d^T F(S_{j-1} \cup y(j)))}{\sum_{k=j}^{n} exp(\lambda_r P(y(k)|q) + \lambda_d^T F(S_{j-1} \cup y(k)))}) \tag{4}$$

3.3 Learning and prediction

As for the training, we generate the training data and optimize the loss function. Then, we use the trained ranking function to re-rank and predict the result.

To generate the ground truth training data, we construct a list $y(i)$ which maximize the ERR-IA metrics. In the algorithm, at the i-th step in loop structure, we select the document d from $D\backslash S_{i-1}$ to maximize ERR-IA score and update the $D\backslash S_{i-1}$ by adding the document d. We get the final training data by recording the best document int every step.

Nextly, stochastic gradient descent method are applied to optimize the loss function. At every step, we calculate the gradient and update the value. The gradient for step i in loop structure at training set D_{init} is computed as follows:

$$\begin{aligned}\Delta\lambda_r^{(i)} = \sum_{j=1}^{n}(&\frac{\sum_{k=j}^{n} P(y(k)|q)exp(\lambda_r P(y(k)|q) + \lambda_d^T F(S_{i-1} \cup y(k)))}{\sum_{k=j}^{n} exp(\lambda_r P(y(k)|q) + \lambda_d^T F(S_{j-1} \cup y(k)))} \\ &- \frac{P(y(j)|q)exp(\lambda_r P(y(j)|q) + \lambda_d^T F(S_{j-1} \cup y(j)))}{exp(\lambda_r P(y(j)|q) + \lambda_d^T F(S_{j-1} \cup y(j))})\end{aligned} \tag{5}$$

$$\begin{aligned}\Delta\lambda_d^{(i)} = \sum_{j=1}^{n}(&\frac{\sum_{k=j}^{n} F(S_{j-1} \cup y(k))exp(\lambda_r P(y(k)|q) + \lambda_d^T F(S_{j-1} \cup y(k)))}{\sum_{k=j}^{n} exp(\lambda_r P(y(k)|q) + \lambda_d^T F(S_{j-1} \cup y(k)))} \\ &- \frac{F(S_{j-1} \cup y(j))exp(\lambda_r P(y(j)|q) + \lambda_d^T F(S_{j-1} \cup y(j)))}{exp(\lambda_r P(y(j)|q) + \lambda_d^T F(S_{j-1} \cup y(j)))})\end{aligned} \tag{6}$$

Finally, the sequential prediction method is used to predict the result. In algorithm, at the i-th step in loop structure, we select the best document d from $D\backslash S_{i-1}$ to maximize our ranking function and update the candidate set $D\backslash S_{i-1}$ by adding document d. Then we predict the final diverse ranking list by recording the best document in every step.

4 Experiments

4.1 Experimental setup

Dataset We use TREC web track (WT2009 for short), WT2010, WT2011 as our dataset. We do our evaluation on the ClueWeb09 Category B retrieval collection[1]. Our query set contains of 150 queries, from TREC web track 2009

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

(WT2009) [5], TREC web track 2010 (WT2010), TREC web track 2011 (WT2011) (50 for each).

Evaluation metrics Three mainly evaluation metrics are used to evaluate the performance of our method: α-NDCG, ERR-IA, NRBP (computed at cutoff 50). To measure the robustness, we use Win/Loss ratio metrics. The evaluation metrics reported at different cutoffs. We use $\{5, 10, 20\}$ as our cutoffs to set up our experiments. The α are set to 0.5 in our experiments.

Baseline methods We use the Indri[2] to conduct our retrieval run with its default parameter configuration. The krovetz stemmer and stopword removal are applied both in the index and retrieval time. All of the search result diversification methods are applied based on the top-50 retrieved documents.

We compare L-HSRD with some baseline 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.
- **MMR**. A classical implicit diversification model.
- **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.
- **HxQuAD**. HxQuAD is a hierarchical diversification model based on xQuAD [12].
- **SVMDIV**. SVMDIV is a learning model for search result diversification [16]. We get the source code from the svmdiv homepage[3] provided by the author.

There exists a single parameter λ to tune in baselines 2-5 (corresponding to MMR, xQuAD, PM2, HxQuAD), we divide the data into 5 parts randomly and perform a 5-fold cross validation to train λ through optimizing ERR-IA. In our model, we also perform 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.

4.2 Experimental results

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

It is noted that L-HSRD always performs best in α-nDCG and ERR-IA. It improves the initial retrieval ranking method with gains up to 34.34%, 48.43%,

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

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
	HxQuAD	0.1653	0.3025	0.1372
	L-HSRD	**0.2084**	**0.3423**	**0.1894**
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-HSRD	**0.2248**	**0.3629**	**0.2011**
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-HSRD	**0.4216**	**0.5374**	**0.3501**

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

20.66%, in terms of α-nDCG on WT2009, WT2010, WT2011 respectively. The improvement of L-HSRD over the MMR in terms of α-nDCG is up to 35.51%, 48.12%, 23.91% on WT2009, WT2010, WT2011 respectively. The improvement of L-HSRD over the xQuAD in terms of α-nDCG is up to 40.06%, 32.16%, 20.44% on WT2009, WT2010, WT2011 respectively, and the improvement of L-HSRD over the PM2 is up to 24.47%, 39.31%, 20.17% on WT2009, WT2010, WT2011 respectively. It indicates that our learning approach tackles the diversity measurement problem more effectively with the consideration of integrate different level of diversity features based on the hierarchical subtopics. Besides the non-learning model, the improvement of L-HSRD over the SVMDIV in terms of α-nDCG is up to 20.15%, 29.79%, 16.45% on WT2009, WT2010, WT2011 respectively. It shows that considering relevance and different type of features in diversity measurement is helpful in diversification. As for the traditional hierarchical diversification method, the improvement of L-HSRD over the HxQuAD in terms of α-nDCG is up to 13.16%, 24.11%, 10.58% on WT2009, WT2010, WT2011, respectively. HxQuAD only use a predefined function to measure the diversity score, and the parameter may not be optimal because it needs to be tuned manually. Our learning model tackles the parameter tuning problem in an automatic fasion and reach optimal result.

Robustness analysis An effective search result diversification method should not only outperforms other models in terms of diversity metrics, but also maintains a high level of robustness. We set up series of experiments on robustness research to study the Win/Loss behaviour.

The Win/Loss ratio are proposed by Yue et al. [16] and Dang et al. [9] for entirety robustness measurement. It denotes whether the model improve or hurt the result when comparing with the basic relevant baseline QL [16, 9] in terms of evaluation metrics. In our experiment, ERR-IA is used to calculate the Win/Loss ratio.

experiment	WT2009	WT2010	WT2011	Total
MMR	20/18	24/17	23/16	67/51
xQuAD	23/18	23/16	24/14	70/48
PM2	22/20	26/14	25/14	73/48
HxQuAD	27/15	30/10	31/10	88/35
L-HSRD	**28/14**	**30/9**	**32/9**	**90/32**

Table 2. Win/Loss ratio

It can be inferred that L-HSRD model performs best with its ratio of 2.65 from the table 2. It reflects the remarkable robustness of L-HSRD model comparing with other diversification models. This confirms the overall performance of our model is not restricted to a small subset, it still work in the whole dataset for three years data.

5 Conclusion and future work

In this paper, we propose a learning based approach to hierarchical search result diversification. We pay our attention to the hierarchical diversification models and introduce the learning approach to address this as a learning problem. We have demonstrated the effectiveness of L-HSRD 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. We believe L-HSRD will play an important role to improve the search result diversification method.

There exists a number of directions to be explored in the future. We are looking forward to take some considerable steps to make L-HSRD achieve convergency as quick as possible. Meanwhile, we are looking forward to make use of the deep learning technology to improve the hierarchical search result diversification.

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).

References

1. Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: Proceedings of the Second ACM International Conference on Web Search and Data Mining. pp. 5–14. ACM (2009)
2. Carbonell, J., Goldstein, J.: 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 (1998)
3. Chapelle, O., Metlzer, D., Zhang, Y., Grinspan, P.: Expected reciprocal rank for graded relevance. In: Proceedings of the 18th ACM conference on Information and knowledge management. pp. 621–630. ACM (2009)
4. Clarke, C.L., Craswell, N., Soboroff, I.: Overview of the trec 2009 web track. Tech. rep., DTIC Document (2009)
5. Clarke, C.L., Craswell, N., Soboroff, I.: Preliminary report on the trec 2009 web track. In: Proc. of TREC (2009)
6. Clarke, C.L., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S., MacKinnon, I.: 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 (2008)
7. Clarke, C.L., Kolla, M., Vechtomova, O.: An effectiveness measure for ambiguous and underspecified queries. Springer (2009)
8. Collins-Thompson, K., Macdonald, C., Bennett, P., Diaz, F., Voorhees, E.M.: Trec 2014 web track overview. Tech. rep., DTIC Document (2015)
9. Dang, V., Croft, W.B.: 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 (2012)
10. Dou, Z., Hu, S., Chen, K., Song, R., Wen, J.R.: Multi-dimensional search result diversification. In: Proceedings of the fourth ACM international conference on Web search and data mining. pp. 475–484. ACM (2011)
11. Drosou, M., Pitoura, E.: Search result diversification. ACM SIGMOD Record 39(1), 41–47 (2010)
12. Hu, S., Dou, Z., Wang, X., Sakai, T., Wen, J.R.: Search result diversification based on hierarchical intents. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 63–72. ACM (2015)
13. Marden, J.I.: Analyzing and modeling rank data. CRC Press (1996)
14. Santos, R.L., Macdonald, C., Ounis, I.: Exploiting query reformulations for web search result diversification. In: Proceedings of the 19th international conference on World wide web. pp. 881–890. ACM (2010)
15. Vargas, S., Castells, P., Vallet, D.: 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 (2012)
16. Yue, Y., Joachims, T.: Predicting diverse subsets using structural svms. In: Proceedings of the 25th international conference on Machine learning. pp. 1224–1231. ACM (2008)
17. Zhu, Y., Lan, Y., Guo, J., Cheng, X., Niu, S.: Learning for search result diversification. In: Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. pp. 293–302. ACM (2014)