Context-aware Ranking Refinement with Attentive Semi-supervised Autoencoders

Bo Xu
xubo@dlut.edu.cn
Dalian University of Technology
Dalian, China

Hongfei Lin
hflin@dlut.edu.cn
Dalian University of Technology
Dalian, China

Dongyu Zhang
zhangdongyu@dlut.edu.cn
Dalian University of Technology
Dalian, China

ABSTRACT

In this work, we propose an attentive semi-supervised autoencoder to refine the ranked results based on Bregman divergence. Hybrid listwise query constraints are investigated in our method to capture the characteristics of relevant documents for different queries. We evaluate the effectiveness of our model on LETOR, and show that our model significantly outperforms other competing methods in improving retrieval performance.

CCS CONCEPTS

• **Information systems → Learning to rank**.

KEYWORDS

Ranking Refinement, learning to rank, pseudo relevance feedback, information retrieval

1 INTRODUCTION

Ranking is one of the most important technical issues in information retrieval (IR). To achieve the desired ranking performance, supervised machine learning methods have been integrated in ranking process and exhibited satisfactory performance, called learning to rank [4, 5]. Fix-length document feature vectors are treated as inputs of learning to rank for constructing ranking models. The quality of the learned ranking models can be affected by several factors, particularly by the usefulness of the document features. [3, 6, 7, 10].

In this work, we adopt autoencoder-based neural networks to generate highly effective and compact query-specific document features via pseudo relevance feedback. Autoencoders [2] have been applied for automatically generating effective features in different tasks [11]. Xu et al. [8] has incorporated ranking information into autoencoders to improve the ranking performance. In their work, two important factors are considered: the feature importance and the query constraints. Different query constraints have been investigated for further improving the ranking performance [9].

Inspired by their work, we propose to incorporate context information of each query via pseudo relevance feedback for generating effective query-specific document features. We adopt the attention mechanism to accurately measuring the feature importance in reconstructing the inputs of autoencoders and investigate different query constraints in our method as hybrid listwise constraints to encode query-level information. Experimental results demonstrate the effectiveness of our method in generating effective document features and improving retrieval performance.

2 CONTEXT-AWARE RANKING REFINEMENT

An autoencoder encodes inputs as low-dimensional representations in its hidden layer, and decodes the hidden representations as outputs. Loss function measures the differences between the inputs and the outputs of an autoencoder for effective feature representations. To learn the tailored autoencoders for learning to rank, we propose to integrate the attention mechanism and hybrid query constraints into the loss function of autoencoder.

First, we introduce the attention mechanism to fully capture the feature importance in training the ranking refinement model based on Bregman divergence [1]. The modified loss function can be formalized as follows.

$$loss = \sum_{i=1}^{n} \Theta^T (x_i - \hat{x_i})^2 \tag{1}$$

$$\Theta^T = softmax(\theta^T (x_i - \hat{x_i})^2) \tag{2}$$

Second, we incorporate another item into the loss function of autoencoders by considering hybrid query constraints. The pretrained ListNet ranker yields two ranking lists of documents based on the inputs and outputs of an autoencoder, respectively. The difference between these two lists of documents can be used to reflect the reconstruction capability of the autoencoder at the query level. We therefore incorporate the difference into the loss function of the autoencoders to guide the learning process for more effective features.

$$loss = \sum_{q \in Q} \eta(l_{in}^q, l_{out}^q)(\sum_{i=1}^{n(q)} \theta^T (\hat{x_i} - x_i)^2) \tag{3}$$

where $\eta(l_{in}^q, l_{out}^q)$ measures the differences between two ranking lists based on the inputs and the outputs of an autoencoder. $n(q)$ is the number of documents corresponding to the query q. The problem is then transformed to the computation of $\eta(l_{in}^q, l_{out}^q)$ for measuring the difference between these two ranking lists of documents at the list level. We adopt two methods to measure the difference. One is to directly measure the divergence of two ranking lists, and the other is to measure the difference by directly comparing the performance using evaluation measures. The first method adopts the cross entropy of two ranking lists of documents to directly compare their difference as follows.

$$\eta_{ce}(l_{in}^q, l_{out}^q) = \sum_{j=1}^{n(q)} P_{l_{in}^q}(j) log(P_{l_{out}^q}(j)) \tag{4}$$

The second method measures the difference of two document lists in terms of retrieval performance. Retrieval performance is evaluated

Table 1: Retrieval performance of retrieval models based on different models. Significant improvement of the proposed models with respect to the baseline models (QSA-listCE) (two-tailed paired t test, $p \leq 0.05$) is indicated with a dagger † .

OHSUMED	P@3	P@5	p@10	NDCG@3	NDCG@5	NDCG@10	MAP
original	0.6016	0.5502	0.4975	0.4732	0.4432	0.4410	0.4457
denoising-AD	0.5063	0.5283	0.4925	0.4312	0.4355	0.4294	0.4381
QSA-listOE	0.6132	0.5774	0.5142	0.4961	0.4779	0.4574	0.4537
QSA-listCE	0.6124	0.5752	0.5150	0.4928	0.4755	0.4601	0.4542
QSA-hybrid	0.6157	0.5801	0.5168	0.4982	0.4779	0.4612	0.4550
QSA-listOE+context	0.6141†	0.5789†	0.5153	0.4958†	0.4770†	0.4591	0.4544
QSA-listCE+context	0.6135†	0.5761	0.5146	0.4940†	0.4764	0.4613†	0.4550†
QSA-hybrid+context	**0.6162**†	**0.5815**†	**0.5177**†	**0.4993**†	**0.4781**†	**0.4622**†	**0.4561**†

based on any existing evaluation metric used in IR tasks, which can be formalized as follows.

$$\eta_{oe}(l_{in}^{q}, l_{out}^{q}) = \frac{|Eval_{in} - Eval_{out}|}{Eval_{in}} \tag{5}$$

To comprehensively consider the query constraints, we combine these two types of listwise constraints for a hybrid one, which can fully consider the query constraints in constructing ranking refinement models.

$$\eta = \lambda\eta_{ce} + (1 - \lambda)\eta_{oe} \tag{6}$$

Based on the equation, we incorporate query constraints of learning to rank into the loss function of the modified autoencoders for learning more effective document features. The modified autoencoder can capture the latent information of the inputs through the enhanced reconstruction capability at the query level, and produce more compact and effective feature representations of documents in the hidden layer.

3 EXPERIMENTS AND ANALYSIS

We evaluated the proposed method on the LETOR dataset [5] released by Microsoft Research Asia. We report the overall ranking performance of different models. Experimental results are shown in Table 1, where ListNet is used both for learning the pre-trained rankers and for learning the final ranking models. In the table, *original* represents the ranking models solely based on the original features without extension. *denoising* represents the ranking models based on the extended features by denoising autoencoders. *QSA-listOE* [8] and *QSA-listCE* [9] represent the ranking model based on the extended features by semi-supervised autoencoders with the defined two types of query constraints, respectively. *QSA-hybrid* represents the model based on the hybrid query constraints. *+context* represents the models considers the context information from pseudo relevance feedback. The results indicate that our context-aware models generally outperforms other models without context information. The model with hybrid query constraints achieves the best performance. This finding demonstrates that context information and hybrid query constraints can jointly contribute to improving the ranking performance.

4 CONCLUSIONS AND FUTURE WORK

We propose a novel method for context-aware ranking refinement. We propose an attentive semi-supervised autoencoder based on Bregman divergence. The attention mechanism is used to measure the feature importance in the loss. Two types of listwise query constraints are investigated and combined in our method to capture the characteristics of relevant documents for different queries. The experimental results show that our model significantly outperforms other competing methods in improving retrieval performance.

ACKNOWLEDGMENTS

This work was partially supported by a grant from the Ministry of Education Humanities and Social Science Project (No.19YJCZH199), the China Postdoctoral Science Foundation (No. 2018M641691), the Natural Science Foundation of China (No. 61632011, 61602078).

REFERENCES

[1] Arindam Banerjee, Srujana Merugu, Inderjit S Dhillon, and Joydeep Ghosh. 2004. Clustering with Bregman Divergences. *Journal of Machine Learning Research* 6, 4 (2004), 1705–1749.

[2] Yoshua Bengio. 2009. Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning* 2, 1 (2009), 1–127.

[3] Victor Lavrenko and W Bruce Croft. 2001. Relevance-Based Language Models. *international acm sigir conference on research and development in information retrieval* 51, 2 (2001), 120–127.

[4] Tie-Yan Liu. 2009. Learning to Rank for Information Retrieval. *Foundations and Trends in Information Retrieval* 3, 3 (2009), 225–331.

[5] Tao Qin, Tie-Yan Liu, Jun Xu, and Hang Li. 2010. LETOR: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval Journal* 13, 4 (2010), 346–374.

[6] Stephen E Robertson and K Sparck Jones. 1976. Relevance weighting of search terms. *Journal of the Association for Information Science and Technology* 27, 3 (1976), 129–146.

[7] Gerard Salton and Chris Buckley. 1997. Improving retrieval performance by relevance feedback. *Journal of the Association for Information Science and Technology* 41, 4 (1997), 355–364.

[8] Bo Xu, Hongfei Lin, Yuan Lin, and Kan Xu. 2017. Learning to Rank with Query-level Semi-supervised Autoencoders. In *Proceedings of the 26th ACM on Conference on Information and Knowledge Management (CIKM)*. ACM, 2395–2398.

[9] Bo Xu, Hongfei Lin, Yuan Lin, and Kan Xu. 2019. Incorporating query constraints for autoencoder enhanced ranking. *Neurocomputing* 356 (2019), 142–150.

[10] Chengxiang Zhai and John D Lafferty. 2004. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems* 22, 2 (2004), 179–214.

[11] Shuangfei Zhai and Zhongfei Zhang. 2016. Semisupervised Autoencoder for Sentiment Analysis. In *The Association for the Advancement of Artificial Intelligence (AAAI)*. 1394–1400.