Learning to rank for Consumer Health Search

Hua Yang[1,2], Xiaoming Liu[1], Binbin Zheng[1] and Guan Yang[1]

[1]School of Computer Science, Zhongyuan University of Technology, Zhengzhou, China
[2]Department of Informatics, University of Évora. Portugal

Abstract

CLEF 2021 eHealth Consumer Health Search task aims to investigate the effectiveness of the information retrieval systems in providing health information to common health consumers. Compared to previous years, this year's task includes three sub-tasks and adopts a new data corpus and set of queries. This paper presents the work of the Zhongyuan University of Technology participating in Subtask 1. It explores the use of learning to rank techniques in consumer health search. A number of retrieval features are used, and eight different learning to rank algorithms are then applied to train the models. The best four models are used to re-rank the documents and four runs are submitted to the subtask.

Keywords

consumer health, information retrieval, learning to rank

1. Introduction

CLEF 2021 eHealth Consumer Health Search (CHS) task is a continuation of the previous CLEF eHealth information retrieval (IR) tasks that started in 2013 [1, 2, 3]. The consumer health search task follows a standard IR shared challenge paradigm from the perspective that it provides a test collection consisting of a set of documents and a set of topics. Participants must retrieve web pages that fulfill a given patient's personalized information need. This needs to fulfill the following criteria: information credibility, quality, and suitability. The 2021 eHealth IR Task includes 3 sub-tasks: ad-hoc information retrieval, weakly supervised information retrieval, and document credibility prediction [4].

This paper describes the Zhongyuan University of Technology (ZUT) approach to CLEF 2021 eHealth IR task Subtask 1. The purpose of Subtask 1 is centered on realistic use cases, and to evaluate IR systems abilities to provide users with relevant, understandable, and credible documents. In this paper, we mainly aim to investigate how a model learned on data from the previous CLEF eHealth IR task [5] performs on this year's new data collection and a new set of queries.

2. Methods

In the information retrieval area, machine learning techniques can be applied to build ranking models for the information retrieval systems, and this is known as Learning to Rank (LTR) [6].

CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania
huayangchn@gmail.com (H. Yang); ming616@zut.edu.cn (X. Liu); luckzbb@hotmail.com (B. Zheng); yangguan@zut.edu.cn (G. Yang)

Table 1
Learning to rank algorithms classification.

Approach	Examples
Pointwise	MART, Random Forest, PRank, McRank
Pairwise	RankNet, RankBoost, RankSVM, LambdaMART
Listwise	LambdaRank, AdaRank, ListNet, Coordinate ascent

Typically, the training data consists of three elements: training queries Q, the associated documents D, and the corresponding relevance judgments or the gold standard *qrel* file for the query and document pairs. The learning algorithms are then used to generate a learning to rank model. The creation of testing data for evaluation is very similar to the creation of the training data which includes the testing queries and the associated documents. To these testing queries, the learning to rank model is jointly used with a retrieval model and to sort the documents according to their relevance to the query, and return a corresponding ranked list of the documents as the response to the query.

Learning to rank methods has been proposed based on different machine learning algorithms. Typically, existing learning to rank can be categorized into three main groups: pointwise, pairwise, and listwise approaches. The pointwise approaches, for example, MART [7] and Random Forests [8], regard the relevance degrees as numerical or ordinal scores, and the learning to rank problem is formulated as a regression or a classification problem. The pairwise approaches, for example, RankBoost [9], LambdaMART [10], and RankNet [11] deal with the ranking problem by treating documents pairs as training instances, and trains models via the minimization of related risks. The listwise approaches, for example, ListNet [12] and AdaRank [13], regard an entire set of documents associated with a query as instances in the training, and trains a ranking function through the minimization of a listwise loss function. Table 1 summarizes a number of the widely used algorithms according to each LTR approach.

In this paper, the dataset and the assessment results from the 2018 CLEF eHealth IR task are used for training the learning to rank models. A number of retrieval features are explored.

2.1. Features Explored for Learning to Rank

In this work, only the regularly used information retrieval features are used to train learning to rank models. They are extracted from a group of 22 different retrieval models [14, 15], as presented in Table 2.

2.2. Training Learning to Rank Models

We build models using eight state-of-the-art learning to rank methods, including two point-wise algorithms, two pair-wise algorithms, and four list-wise algorithms. The point-wise algorithms are MART [7] utilizing gradient boosting regression trees, and Random Forests [8] using regression. The pair-wise algorithms are RankNet [11] employing relative entropy as a loss function and gradient descent to train a neural network model, and RankBoost [9] based on boosting. The list-wise algorithms include AdaRank [13] based on boosting, Coordinate

Table 2
Features used for learning to rank models.

No.	The retrieval model used for feature extracting
1	BB2
2	BM25
3	DFI0
4	DFR_BM25
5	DLH
6	DLH13
7	DPH
8	DFRee
9	Hiemstra_LM
10	DirichletLM
11	IFB2
12	In_expB2
13	In_expC2
14	InL2
15	LemurTF_IDF
16	LGD
17	PL2
18	TF_IDF
19	DFRWeightingModel
20	PL2
21	Tf
22	Dl

Ascent [16] where the ranking scores are calculated as weighted combinations of the feature values, LambdaMART [10] combining MART and LambdaRank and directly optimize NDCG in training, and ListNeT [12] based on neural networks.

The dataset and the topical relevance assessments of the 2018 CLEF eHealth IRtask [5] are used as the training data. In the assessment files, the corresponding documents are scored with 0, 1, or 2, which represent *not relevant*, *relevant*, or *highly relevant*, respectively.

3. Experiments and Results

This section first presents the experimental settings, the dataset and queries for the subtask, and the evaluation measures used for the assessments. Then we describe the experiments we performed and analyze the results.

3.1. Experimental Settings

Terrier[1] platform version 5.4 is chosen as the IR model of the system. The Okapi BM25 weighting model is used as the retrieval model, with all the parameters set to default values (k_1 = 1.2d,

[1] http://terrier.org/

```
<topic>
        <id>101</id>
        <query>heavy flares swelling lymph nodes
        </query>
</topic>

<topic>
        <id>1</id>
        <query> What are the most common chronic diseases? What effects
        do chronic diseases have for the society and the individual?
        </query>
</topic>
```

Figure 1: Example topics in the CLEF 2021 CHS Subtask 1.

k_3 = 8d, b = 0.75d). All developed learning to rank models are implemented with RankLib[2] version 2.15.

3.2. Dataset

The dataset of the CLEF 2021 CHS task is basically constructed using the collection introduced in CLEF 2018 IR task, and extended with additional webpages and social media content. Totally, the collection consists of over 5 million medical webpages from selected domains acquired from the CommonCrawl and other resources [4].

3.3. Topics

Totally 55 topics are used in the CLEF 2021 CHS task, and they are based on realistic search scenarios. These topics are divided into two sets. The reddit-topics set includes 25 topics that are based on use cases from discussion forums. These queries are extracted and manually selected from Google trends to best fit each use case. The patients-topics set includes 30 topics which are based on discussions with multiple sclerosis and diabetes patients. These queries are manually generated by experts from established search scenarios. Figure 1 presents the example topics used in the task.

3.4. Pre-processing

All queries are pre-processed with characters lower-casing, stop words removing and Porter Stemmer stemming. The default stop words list available in the IR platform Terrier 5.4 is used.

3.5. Evaluation Measures

The task takes into account 3 dimensions in the relevance evaluation: topical relevance, understandability, and credibility. The ability of systems to retrieve relevant, readable, and credible documents for the topics, and the ability of systems to retrieve all kinds of documents (web or

[2]https://sourceforge.net/p/lemur/wiki/RankLib/

Table 3
The best four learning to rank models.

LTR model	LTR algorithm	NDCG@10
m_lm	LambdaMART	0.9662
m_mr	MART	0.8869
m_rf	Random Forests	0.6744
m_rb	RankBoost	0.5821

social media) are both considered. Evaluation measures used are NDCG@10, BPref, and RBP, as well as other metrics adapted to other relevance dimensions such as uRBP.

3.6. Experiments

Using the data from the CLEF 2018 ehealth IR task, we totally train eight learning to rank models. The loss function used to train the learning to rank model is NDCG@10. We choose the best four performed LTR models and use them in this year's task. The evaluation of these top four LTR models is presented in Table 3.

The top 1,000 relevant documents for each query are retrieved using the BM25 retrieval model in Terrier. The selected four models are then used to re-rank the initial results obtained with the BM25 retrieval model, and four runs are generated for the final submission.

3.7. Results

For each topic, 250 documents have been assessed in three relevance dimensions. And we compare our four run results to the six baselines, as shown in Table 4.

We first compare the performance among our four implemented models. The best result was obtained by the model *m_rf* which used Random Forests learning to rank algorithm, then followed by the model *r_rb* with RankBoost algorithm and the model *m_lm* with LambdaMART algorithm. On average, the model *m_mr* with MART algorithm achieved the worst result, although it showed somewhat better results in MAP and two cRBP measures when compared to the model *m_lm*.

Then we compare the best model *m_rf* with the baselines. When compared in MAP, this model was able to surpass all baselines. In Bpref, the model showed better results than the *DirichletLM_qe* baseline, but failed with other baselines. In the rRBP measures, the model showed better results than the two *DirichletLM* baselines. In the cRBP and the RBP measures, the model surpassed the baseline *BM25* and the two *DirichletLM* baselines.

4. Conclusion and Future Work

This paper reports the ZUT team participation in the CLEF 2021 eHealth CHS Subtask 1. Using the data from the CLEF 2018 eHealth IR task, a number of retrieval features are explored and eight learning to rank algorithms are used to train the LTR models. The top performed LTR models are used in the CLEF 2021 eHealth IR task Subtask1. In the future work, the methods

Table 4
The results and comparison to the baselines.

Run	MAP	Bpref	NDCG @10	binary rRBP	graded rRBP	binary cRBP	graded cRBP	binary RBP	graded RBP
m_rf	4.090	4.686	6.148	7.035	4.943	6.227	4.138	6.028	7.426
m_rb	3.733	4.472	5.651	6.572	4.499	6.036	4.088	5.599	6.978
m_lm	3.381	4.409	5.258	6.248	4.076	5.240	3.187	5.198	6.655
m_mr	3.383	4.278	4.817	5.615	3.486	5.247	3.295	4.805	6.269
TF_IDF_qe	3.974	5.106	6.535	7.664	5.232	6.849	4.497	6.428	8.010
TF_IDF	3.663	4.744	6.464	7.443	5.091	6.399	4.179	6.280	7.796
BM25_qe	3.903	4.994	6.352	7.397	5.072	6.447	4.317	6.277	7.700
BM25	3.641	4.707	6.364	7.337	5.012	6.201	4.062	6.185	7.661
DirichletLM	3.694	4.724	5.952	6.839	4.632	6.599	4.578	5.844	7.340
DirichletLM_qe	2.423	3.691	5.362	6.341	4.082	6.366	4.285	5.345	6.960

proposed in this paper will be further analyzed: different learning to rank features will be explored, and an ensemble algorithm will be investigated.

References

[1] H. Suominen, L. Goeuriot, L. Kelly, L. A. Alemany, E. Bassani, N. Brew-Sam, V. Cotik, D. Filippo, G. González-Sáez, F. Luque, P. Mulhem, G. Pasi, R. Roller, S. Seneviratne, R. Upadhyay, J. Vivaldi, M. Viviani, C. Xu, Overview of the clef ehealth evaluation lab 2021., in: CLEF 2021 - 11th Conference and Labs of the Evaluation Forum, Lecture Notes in Computer Science (LNCS), Springer, September 2021.

[2] H. Suominen, L. Kelly, L. Goeuriot, E. Kanoulas, L. Azzopardi, R. Spijker, D. Li, A. Névéol, L. Ramadier, A. Robert, J. Palotti, Jimmy, G. Zuccon, Overview of the clef ehealth evaluation lab 2018., in: CLEF 2018 - 8th Conference and Labs of the Evaluation Forum, Lecture Notes in Computer Science (LNCS), Springer,September„ 2018.

[3] Jimmy, G. Zuccon, J. Palotti, L. Goeuriot, L. Kelly, Overview of the clef 2018 consumer health search task., in: CLEF 2018 Evaluation Labs and Workshop: Online Working Notes, CEUR-WS,September„ 2018.

[4] L. Goeuriot, G. Pasi, H. Suominen, E. Bassani, N. Brew-Sam, G. Gonzalez-Saez, R. G. Upadhyay, L. Kelly, P. Mulhem, S. Seneviratne, M. Viviani, C. Xu, Consumer health search at clef ehealth 2021, in: CLEF 2021 Evaluation Labs and Workshop: Online Working Notes, CEUR-WS, September 2021.

[5] J. Jimmy, G. Zuccon, J. Palotti, L. Goeuriot, L. Kelly, Overview of the clef 2018 consumer health search task (2018).

[6] T.-Y. Liu, et al., Learning to rank for information retrieval, Foundations and Trends® in Information Retrieval 3 (2009) 225–331.

[7] J. H. Friedman, Greedy function approximation: a gradient boosting machine, Annals of statistics (2001) 1189–1232.

[8] L. Breiman, Random forests, Machine learning 45 (2001) 5–32.

[9] Y. Freund, R. Iyer, R. E. Schapire, Y. Singer, An efficient boosting algorithm for combining preferences, Journal of machine learning research 4 (2003) 933–969.

[10] Q. Wu, C. J. Burges, K. M. Svore, J. Gao, Adapting boosting for information retrieval measures, Information Retrieval 13 (2010) 254–270.

[11] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. Hullender, Learning to rank using gradient descent, in: Proceedings of the 22nd international conference on Machine learning, 2005, pp. 89–96.

[12] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, H. Li, Learning to rank: from pairwise approach to listwise approach, in: Proceedings of the 24th international conference on Machine learning, ACM, 2007, pp. 129–136.

[13] J. Xu, H. Li, Adarank: a boosting algorithm for information retrieval, in: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2007, pp. 391–398.

[14] C. Macdonald, R. L. Santos, I. Ounis, B. He, About learning models with multiple query dependent features, ACM Transactions on Information Systems (TOIS) 31 (2013) 11.

[15] C. Macdonald, R. L. Santos, I. Ounis, The whens and hows of learning to rank for web search, Information Retrieval 16 (2013) 584–628.

[16] D. Metzler, W. B. Croft, Linear feature-based models for information retrieval, Information Retrieval 10 (2007) 257–274.