UCAS at TREC-2018 Precision Medicine Track

Zhi Zheng, Canjia Li, Ben He, and Jungang Xu

School of Computer Science and Technology
University of Chinese Academy of Sciences
{zhengzhi18, licanjia17}@mails.ucas.ac.cn, {benhe, xujg}@ucas.ac.cn

Abstract. This paper describes the system developed for the TREC 2018 Precision Medicine track. We adopt BM25F model with query expansion to retrieve clinical trials. For scientific abstract task, we use BM25 model to generate an initial ranking list and then adopt two methods to re-rank. Experimental results show that a new model that penalizes the articles unrelated to treatment, prevention, and prognosis improves the performance for scientific abstracts task.

1 Introduction

TREC Precision Medical track 2018 (PM2018) focuses on matching patients with existing articles from PubMed Central (PMC) and experimental treatments in clinical trials from ClinicalTrials.gov website. Specifically, there are two tasks in PM track: clinical trials and scientific abstracts. The goal of retrieving clinical trials is to identify trials for which the given patient is eligible to enroll. The goal of retrieving scientific abstracts is to identify relevant articles for the treatment, prevention, and prognosis of the disease under the specific conditions for the given patient. There are 50 topics concerning patients' condition: disease, genetic variants, demographic. For each collection, participants are allowed to submit a maximum of five runs.

In this paper, we describe the system developed for the TREC 2018 Precision Medicine track. We adopt BM25F model with query expansion to retrieve clinical trials. For scientific abstract task, we use BM25 model to generate an initial ranking and then adopt two methods for the re-ranking.

The rest of the paper is organized as follows. Section 2 gives a detailed introduction to our retrieval system for the two tasks, respectively. Section 3 presents the experimental results and analysis. Finally, Section 4 concludes our experiments.

2 Method

In this section, we give a detailed introduction to our approach for each task, respectively.

2.1 Topic Expansion and Preprocessing

For a given topic, a document can be judged as relevant if it contains synonyms or abbreviations of the disease. For example, “head and neck squamous cell carcinoma” has an abbreviation “HNSCC” and “gastric cancer” has a synonym “stomach cancer”. We add these abbreviations and synonyms into the original topic. Moreover, a document might also be relevant to the topic if it doesn’t contain the exact disease but more general concepts about diseases. For example, in clinical trials task, a document which doesn’t contain the given disease “melanoma” but contains “tumor” can be relevant to the topic if it matches other conditions, i.e., genetic variants and demographic. We also add these terms into topic.

The topics consist of three parts, namely the disease, genetic variants and demographic. We ignore the demographic part by filtering out clinical trials that don’t match the demographic requirements during post-processing. We assume that disease is a better indicator of patients’ conditions, and the more general the term is, the lower weight it should be assigned. Thus, the disease (including its synonyms and abbreviations), the name of the gene, the type of mutation (e.g., “amplification”), the general disease term (e.g., “tumor”) are assigned with term weights of 4.0, 3.0, 2.0, 2.0. For example, topic 7 is reformulated as follows:

$$\boxed{melanoma^{4.0} braf^{3.0} amplification^{2.0} tumor^{2.0}}$$

2.2 Index

We use Terrier[1] to index clinical trials and scientific abstracts after Porter stemming and stopword removal. For clinical trials, the indexed fields are *NCT ID, official title, brief summary, detailed description, eligibility criteria, arm group* and *keyword*. The *PMID, title, abstract* fields are indexed for all abstracts from PubMed and AACR/AASCO proceedings. For PubMed abstracts, we additionally index *MeSH terms, journal title, publication type* and *chemical compounds* fields.

2.3 Document Retrieval and Ranking

In this section, we introduce our methods for document retrieval and ranking for two tasks, respectively.

2.3.1 Scientific Abstracts

BM25 model. An initial ranking list is generated by the well-established BM25 model[2]. Given a document d and a query q, the ranking function is

$$score_1(q,d) = \sum_{t \in q} w_t \frac{(k_1+1)tf}{K+tf} \frac{(k_3+1)qtf}{k_3+qtf} \tag{1}$$

where t denotes one term in the query, and qtf is the term frequency of t in query q. tf is the term frequency of query term t in document d. K is calculated as follows:

$$K = k_1((1-b) + b \cdot \frac{l}{\mathrm{avg}_l} \tag{2}$$

where l and avg_l denote the length of document d and the average length of documents in the whole collection. k_1, k_3 and b are free parameters whose default setting is $k_1 = 1.2$, $k_3 = 1000$ and $b = 0.75$, respectively. w_t is the weight of query term t, which is given by:

$$w_t = \log_2 \frac{N - df_t + 0.5}{df_t + 0.5} \tag{3}$$

where N is the number of documents in the collection, and df_t is the document frequency of query term t, which denotes the number of documents that contains t.

K-NRM. In this work, we employ a modified version of K-NRM[3], which is adopted for the PM tasks as proposed in [4], to re-rank the initial ranking. K-NRM which is a state-of-the-art neural retrieval model using Gaussian kernels for the relevance matching of term pairs.

Given a query q and document d, K-NRM first maps each term t to an L-dimension embedding $\vec{v_t}$. Then the cosine similarity of each query-document term pair is computed, results in a translation matrix M:

$$M_{ij} = \cos(\vec{v}_{t_i^q}, \vec{v}_{t_j^d}) \tag{4}$$

After that, K-NRM uses f Gaussian kernels to obtain the soft match between each query term and terms in a document, which is given by

$$K_k(M_i) = \sum_j \exp(-\frac{(M - \mu_k)^2}{2\sigma_k^2}) \tag{5}$$

where μ_k and σ_k are the mean and variance of kernel k. Subsequently, the kernel vector for query term t_i is composed by the f kernels, as shown in Equation 6.

$$\vec{K}(M_i) = \{K_1(M_i), ..., K_f(M_i)\} \tag{6}$$

Then the query-document ranking features $\phi(M)$ can be calculated as follows:

$$\phi(M) = \sum_{i=1}^{m} \log \vec{K}(M_i) \cdot w(t_i^q) \tag{7}$$

where $w(t_i^q)$ is the query term weights. Next, $\phi(M)$ is fed into a learning to rank layer to produce the relevance score as follows:

$$score_2(q,d) = \tanh(w^T \phi(M) + b) \tag{8}$$

Ultimately, we interpolate the ranked list generated by K-NRM with the initial one after normalizing scores by Min-Max normalization, as shown in Equation 9.

$$score(q, d) = \lambda \cdot score_1(q, d) + (1 - \lambda) \cdot score_2(q, d) \tag{9}$$

Treatment information. The goal of retrieving scientific abstracts is to identify relevant articles for the *treatment, prevention, and prognosis* of the disease under the specific conditions for the given patient. Abstracts discussing information not useful for these goals will not be considered relevant.

In our experiments, we find that the MeSH terms like "therapy" and "diagnosis" are indicative to the treatment. Thus, we use MeSH terms to judge whether a document in the initial ranking list contains treatment information, and promote its ranking in during re-ranking if it is indeed related to the treatment.

2.3.2 Clinical Trials

BM25F model. For clinical trials task, we obtain document ranking using BM25F model [5]. Given a query q and a document d, for each field f in d, a normalized term frequency is computed as follows:

$$\overline{tf}_f = \frac{tf_f}{((1 - b_f) + b_f \cdot \frac{l_f}{\mathrm{avg}_{l_f}})} \tag{10}$$

where tf_f is the term frequency in field f, b_f is a field-dependent parameter, l_f is the length of field f and avg_{l_f} is the average length of field f in the whole document collection.

Then, these term frequencies can be combined in a linearly weighted sum to obtain the term pseudo-frequency:

$$\overline{tf} = \sum_f W_f \cdot \overline{tf}_f \tag{11}$$

where W_f is the weight for each field. Finally, the ranking function is given by:

$$score(q, d) = \sum_{t \in q} \frac{\overline{tf}}{k_1 + \overline{tf}} \cdot w_t \tag{12}$$

where w_t is defined by Equation 3 and k_1 is a free parameter.

Query Expansion. The query expansion mechanism extracts the most informative terms from the top-returned documents as the expanded query terms. In our experiments, we use Terrier's DFR-based term weighting models, namely Bo1 and KL [6].

Post-processing. In order to filter out the clinical trials for which the patient is not eligible to enroll, we extract *gender, minimum_age* and *maximum_age* fields from the documents returned to check whether they match the patient demographics. Documents that do not meet the criteria will be discarded.

3 Results

3.1 Run Description

For scientific abstracts task, we submitted five runs, all of which use BM25 model with no query expansion and perform Porter stemming and stopword removal, so we omit these parts in the table. Differences between fives runs are shown in Table 1.

Table 1. Runs submitted to the SA task

RunID	Topic Expansion	Re-rank Method
UCASSA1	Yes	-
UCASSA2	No	-
UCASSA3	Yes	K-NRM
UCASSA4	No	K-NRM
UCASSA5	No	Treatment Information

For clinical trials task, we submitted five runs which are summarized in Table 2. We employ topic expansion for all five runs, so this part is omitted in the table. The column *Preprocessing* denotes stopword removal and stemming.

Table 2. Runs submitted to the CT task

RunID	Baseline Model	Field Weights	Query Expansion	Preprocessing
UCASCT1	BM25F	Group 1	Bo1	No
UCASCT2	BM25F	Group 2	Bo1	No
UCASCT3	BM25F	Group 2	Bo1	Yes
UCASCT4	BM25F	Group 1	Bo1	Yes
UCASCT5	BM25	-	KL	No

3.2 Evaluation Results

Table 3. Evaluation results for SA task

RunID	infNDCG	P@10	R-prec
UCASSA1	0.5352	0.5700	0.3492
UCASSA2	0.5450	0.5640	**0.3654**
UCASSA3	0.5452	0.5720	0.3480
UCASSA4	0.5346	0.5720	0.3560
UCASSA5	**0.5580**	**0.5980**	0.3646

Table 4. Evaluation results for CT task

RunID	infNDCG	P@10	R-prec
UCASCT1	0.5303	**0.5500**	0.3931
UCASCT2	0.5313	0.5480	0.4009
UCASCT3	0.5226	0.5240	0.4004
UCASCT4	**0.5347**	0.5440	**0.4019**
UCASCT5	0.5221	0.5240	0.3746

The evaluation results of our runs for scientific abstracts and clinical trials are shown in Table 3 and 4, respectively. Top scores are highlighted in boldface. For scientific abstracts task, the run *UCASSA5* outperforms other runs, according to infNDCG and P@10 metrics, suggesting that the treatment information plays an important role in this task. For clinical trials task, the run *UCASCT4* outperforms other runs, according to infNDCG and R-prec metrics.

4 Conclusions

In this paper, we describe the system developed for the TREC 2018 Precision Medicine track. We adopt BM25F model with query expansion to retrieve clinical trials. For scientific abstract task, we use BM25 model to generate an initial ranking list. Two different methods are employed to re-rank the initial results. Experimental results show that the effectiveness of the abstract retrieval can be improved by penalizing articles that are not related to treatment, prevention, and prognosis.

Acknowledgements

This work is supported in part by the Beijing Natural Science Foundation under Grant No. 4162067, and the National Natural Science Foundation of China (61472391).

References

1. C. Macdonald, R. McCreadie, R. L. Santos, and I. Ounis, "From puppy to maturity: Experiences in developing terrier," *Open Source Information Retrieval*, vol. 60, 2012.
2. S. Robertson, H. Zaragoza, *et al.*, "The probabilistic relevance framework: Bm25 and beyond," *Foundations and Trends® in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009.
3. C. Xiong, Z. Dai, J. Callan, Z. Liu, and R. Power, "End-to-end neural ad-hoc ranking with kernel pooling," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pp. 55–64, 2017.
4. C. Li and B. He, "Neural precision medicine by mining implicit treatment concepts," in *2018 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2018, Madrid, Spain, December 3-6, 2018*, 2018.

5. H. Zaragoza, N. Craswell, M. J. Taylor, S. Saria, and S. E. Robertson, “Microsoft cambridge at TREC 13: Web and hard tracks,” in *Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004*, 2004.
6. G. Amati, *Probability models for information retrieval based on divergence from randomness*. PhD thesis, University of Glasgow, UK, 2003.