ECNU at TREC 2016: Web-based query expansion and experts diagnosis in Medical Information Retrieval

Hongyu Liu[1,2], Yang Song[1,2], Yun He[1,2], Yueyao Wang[1,2], Qinmin Hu[1,2], and Liang He[1,2]

[1] Shanghai Key Laboratory of Multidimensional Information Processing
[2] Department of Computer Science & Technology, East China Normal University, Shanghai, 200241, China
{liuhy,ysong,yhe,yywang}@ica.stc.sh.cn, {qmhu,lhe}@cs.ecnu.edu.cn

Abstract. In this paper, we present our work in TREC 2016 Clinical Decision Support Track. Among five submitted runs, two of them are based on summary topics and the others on note topics. In summary version run, we expand the original text with external data on web. Note topics are much longer than the summary, which contain a significant number of medical abbreviations as well as other linguistic jargon and style. An automatic method and a manual method are applied to process note topics. In the automatic method, we utilize KODA, a well-known knowledge drive annotator, to extract key information from the original text. In the manual one, we ask medical experts to diagnose and give their advice. For all of the five runs, we adopt Terrier search engine to implement various retrieval models. Furthermore, results combinations are applied to improve the performance of our model.

1 Introduction

Similar to track in 2014 and 2015, the focus of the 2016 Clinical Decision Support Track is the retrieval of biomedical articles for answering clinical questions about medical records. However, different from previous years, note parts are added into the topics. They are generated by the clinicians during the first few hours for the patient in the hospital, including patients chief complaint, relevant medical history and lots of other necessary information. They contain a significant number of medical abbreviations as well as other linguistic jargon and style[3]. According to the instruction this year, three of our five submitted runs use summary topics and the left 2 runs use note topics.

Since summary is short and terse, we simply Google it to find the most related information. Top 10 web pages are crawled and only particular content of

[3] http://www.trec-cds.org/

these pages are reserved. MeSH[4] thesaurus is then used to map candidate words in the reserved content. MeSH is the National Library of Medicines controlled vocabulary thesaurus. We expand summary text with these candidate words. Those expanded summaries are the final queries in summary version run.

In the note version run, we utilize KODA[5] to map key words from the original text. With the help of knowledge base like DBpedia[6], KODA is able to identify the most important words. For example, if original text is "A 55-year-old woman with sarcoidosis, presenting today with confusion and worsening asterixis", the annotated words will be "woman", "sarcoidosis" and "worsening asterixis". MeSH thesaurus is also combined with KODA to reduce redundant words. In the manual run, we call for medical experts' diagnoses of patients with their advices regarded as queries.

In document processing, five parts of the given documents are reserved including id, title, abstract, body and reference title. We adopt Terrier[7] [1] to build index with which to retrieve. Results from different retrieval model are combined to form the final run. The whole procedure is illustrated in Figure 1.

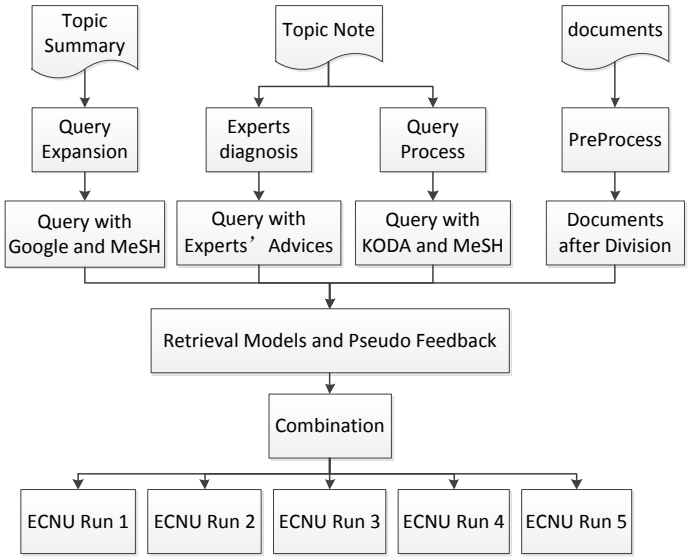

Fig. 1. Process of Retrieval Algorithm

[4] https://www.nlm.nih.gov/pubs/factsheets/mesh.html
[5] http://smartdocs.tudor.lu/koda/about.html
[6] http://wiki.dbpedia.org/
[7] http://terrier.org/

2 Methodology

2.1 Web based Query Expansion

Summary parts usually contain one or two sentences which are brief description of a patient. Generally speaking, related biomedical articles will be more likely to be retrieved if more related information is involved in. Google search is adopted as it is the most-used search engine on the World Wide Web. We Google the summary text. Qualified key words returned from the search engine will be added into the original query[2]. Then the new query is searched using different IR models. We combine the outputs to form final run. The web-based query expansion model is proposed as follow.

- Google search engine is used to search original summary text Q_0 and web titles and snippets are extracted from the top ten related web pages.
- Remove ambiguous words in MeSH file to produce a MeSH dictionary.
- Identify the candidate words from the titles and snippets using MeSH dictionary.
- Sort the candidate words according to their occurring frequency in descending order. Q_{web} consists of the top N candidate words.
- Terms in Q_0 are assigned with weight w_1 and terms in Q_{web} are assigned with weight w_2.
- Final query is denoted as $Q = Q_0 \cup Q_{web}$.
- Adopt Terrier search engine to run classical information retrieval model, improved by pseudo feedback[3].
- Combine the output from different IR models. The combine strategy is illustrated in section 2.4.

2.2 Knowledge based Query Expansion

Note topic is pretty long compared to summary. It may not be used directly to search articles as it contains lots of medical abbreviations. It's necessary to identify the most important words in note. The final query consists of these important words. Our model procedure is illustrated as follow.

- Annotation platform KODA is applied to identify important words in notes. These annotated words are added into candidate words set W_c. We choose DBpedia-en as our knowledge base.
- Use MeSH dictionary to extract words from W_c. Matching words are reserved as final query Q.
- If we are able to find the original text of an abbreviation word in MeSH, we also put that original text into final query Q.
- Adopt Terrier search engine to run classical information retrieval model, adding pseudo feedback.
- Combine the outputs from different IR models. The combine strategy is illustrated in section 2.4.

2.3 Experts diagnosis

This method is a manual one. As we mentioned in section 2.2, an admission note describes a patient's chief complaint, relevant medical history, and any other information. It is generated by human and is usually pretty long. We call medical experts in and ask them to read the whole note text. Then based on the description of symptom, they may infer the name of corresponding disease. For different clinical question type, we expand the disease name by adding the name of type. For example, if a disease name is "common fever" and type is "diagnosis", the final query is "common fever diagnosis". We adopt terrier as our search engine and use different search models. Finally we combine the outputs from these search models. Combine strategy is illustrated in section 2.5.

2.4 Combination

We adopt Terrier to run multiple IR models including BM25[4], TF-IDF, BB2[5], etc. Outputs from different IR models are combined. There are two combination strategies.

2.4.1 Combination Method 1

The outputs are in this format:

TOPIC_NO Q0 PMCID RANK SCORE RUN_NAME

Documents with bigger score are shown in topper position. For query Q_i, we use the following formulation to calculate the score for document D_j:

$$\sum_{N}(\frac{S_{ijk}}{\sum_{M} S_{ijk}}) \tag{1}$$

S_{ijk} denote the score of document D_j for query Q_i using IR model F_k;
M is 1000, denote the top 1000 related documents for query Q_i;
N is the total number of used IR model.
Then we rank the documents based on score in descending order.

2.4.2 Combination Method 2

First , we use the 0-1 normalization[8] on score for each output. Then, for each IR model, we get intersection documents from the outputs. For each document, we sum its score of different IR model. Finally we rank the documents based on score in descending order. The score formulation for document D_j follows:

$$\begin{cases} \sum_{N} S_{ijk} & , if D_j \in O \\ 0 & , if D_j \notin O \end{cases} \tag{2}$$

[8] https://en.wikipedia.org/wiki/Feature_scaling

$$O = O_{i1} \cap O_{i2} ... \cap O_{in} \tag{3}$$

S_{ijk} denote the normalized score of document D_j for query Q_i using IR model F_k;

N is the total number of used IR model;

O_k denote the output of IR model F_k for query Q_i.

3 Experiments and Evaluation

3.1 Document Processing

The original documents are in NXML format, they contain lots of useless web tags. We use regular expression to extract PMCID, title, abstract, body and reference title. PMCID is the unique identifier of the document and is specified by the $< article - id >$ element within each document.

3.2 Submission and Evaluation

Table 1. Summary of evaluation

Run	infAP	infNDCG	R-prec	P @ 10
ECNU 1	0.0296	0.2225	0.1598	0.2867
ECNU 2	0.0243	0.1810	0.1229	0.2433
ECNU 3	0.0276	0.2168	0.1485	0.2733
ECNU 4	0.0242	0.1814	0.1118	0.2433
ECNU 5	0.0313	0.2334	0.1572	0.3367

We totally submit five runs where three runs use summary and two runs use note. The description for each run is as follows. And the evaluation of our submissions is summarized in Table 1.

- ECNU 1: We use Google and MeSH to do query expansion and use combination method 1 to improve performance. (summary)
- ECNU 2: We ask the medical experts to diagnosis. (note)
- ECNU 3: We use Google and MeSH to do query expansion and use combination method 2 to improve performance. (summary)
- ECNU 4: We use KODA and MeSH to process query and use combination method 1. (note)
- ECNU 5: We use the same strategy as ECNU 1 and only the parameters of the model are different from it. (summary)

Table 2. Automatic runs using clinical note topics

Run	infAP	infNDCG	R-prec	P @ 10
Best	0.0599	0.3302	0.19935	0.51
Median	0.00989	0.12279	0.0792	0.1833

Table 3. Automatic runs using summary topics

Run	infAP	infNDCG	R-prec	P @ 10
Best	0.0869	0.4377	0.25535	0.63
Median	0.01959	0.18589	0.122	0.2633

3.3 Discussion

Table 2 and Table 3 display the best and median score of the whole participant using note topics and summary topics. Our note version runs ECNU 2 and ECNU 4 both perform better than the median, so do our summary version runs. This denotes that our model could achieve a relative good performance.

Figure 2 displays the max, median and ECNU 5 scores of infNDCG per summary topic. We may find that there are 9 topics score are lower than the median, especially topic 17 and topic 25. In topic 17, there is old woman has disease related to heart. However the expansion words for topic 17 are "physicians", "woman", "blood", etc. The expansion words have nothing to do with the disease. Although we assign a lower weight to the expanded words than the original words in topic, the uncorrelated expanded words could still hurt the performance. Expansion for topic 25 also has the same kind of problem as topic 17.

Figure 3 display the max, median and ECNU 4 scores of infNDCG per note topic. There are 5 topics score are lower than the median, particularly topic 8, topic 11 and topic 21. However we could not find the difference between these topics and other topics. We need to find more detail information of these topics in the future.

4 Conclusion and Future Work

In this paper, we propose web-based query expansion and experts diagnosis models in the 2016 TREC Clinical Decision Support Track. Our experimental results show that our model could achieve a relative good performance. All of our runs' scores are better than median. However we find that the scores of some topics are lower than median since we add too many noisy words into the original query. A more sophisticated model with query expansion based on both words occuring times and context shall be constructed.

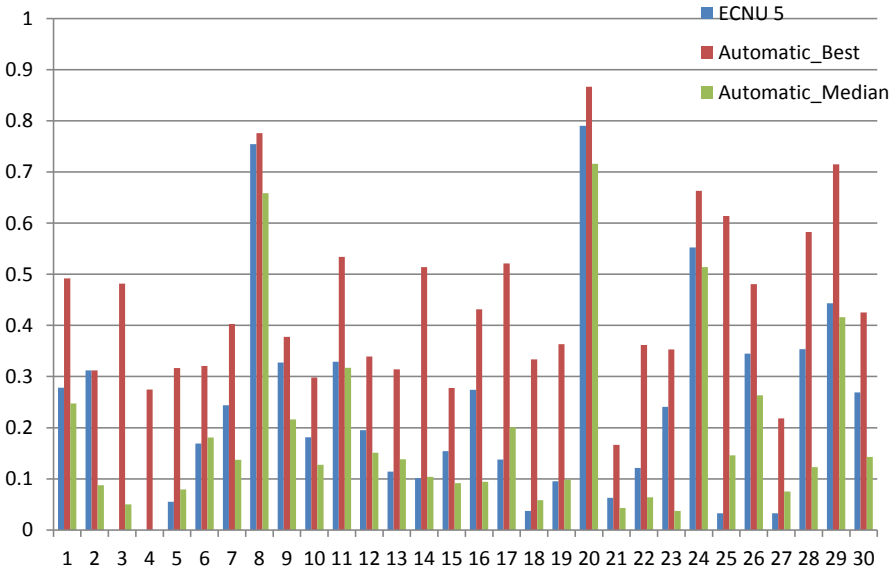

Fig. 2. max, Median and ECNU 5 scores of infNDCG per summary topic

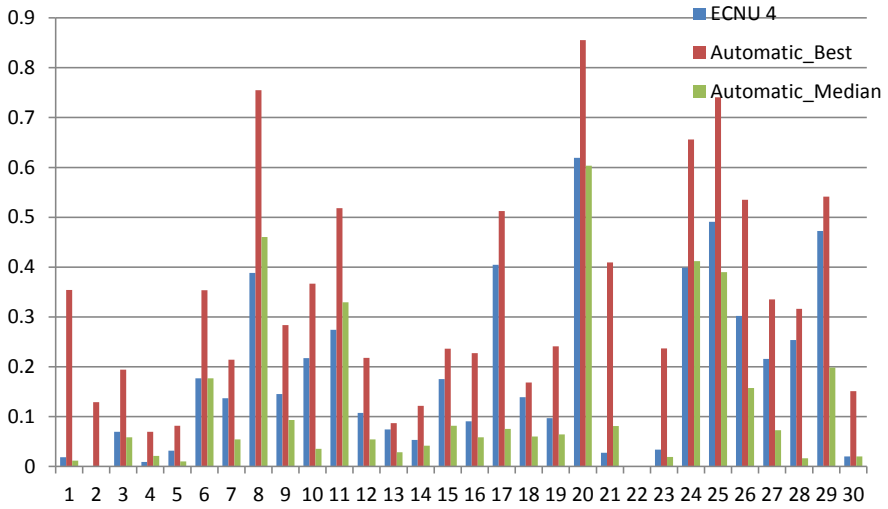

Fig. 3. max, Median and ECNU 5 scores of infNDCG per note topic

Acknowledgment

This research is funded by the National Natural Science Foundation of China (No. 61602179) and the Science and Technology Commission of Shanghai Municipality (No.15PJ1401700).

References

1. V.P.Iadh Ounis, G. A. (2005). terrier information retrieval platform. advances in information retrieval.
2. Yang SongHe, Qinmin Hu, Liang HeYun. (2011). ECNU at 2015 CDS Track: Two Re-rangking Methods in Medical Information Retrieval. The 25th Text Retrieval Conference Proceeings TREC.
3. Kelly, Diane, and Jaime Teevan. "Implicit feedback for inferring user preference: a bibliography." ACM SIGIR Forum. Vol. 37. No. 2. ACM, 2003.
4. Hu.X. H. QinminJ. (2010). passage extraction and result combination for genomics information retrieval. journal of intelligent information systems, 34(3):249-274.
5. Probabilistic Models for Information Retrieval based on Divergence from Randomness. G. Amati. PhD Thesis, School of Computing Science, University of Glasgow, 2003.
6. Kirk Roberts, Matthew S. Simpson, Ellen Voorhees, William R. Hersh. Overview of the TREC 2015 Clinical Decision Support Track. In The 25th Text Retrieval Conference Proceedings TREC, 2016.