Modelling the Usefulness of Document Collections for Query Expansion in Patient Search

Nut Limsopatham*
University of Cambridge, UK
nl347@cam.ac.uk

Craig Macdonald
University of Glasgow, UK
craig.macdonald@glasgow.ac.uk

Iadh Ounis
University of Glasgow, UK
iadh.ounis@glasgow.ac.uk

ABSTRACT

Dealing with the medical terminology is a challenge when searching for patients based on the relevance of their medical records towards a given query. Existing work used query expansion (QE) to extract expansion terms from different document collections to improve query representation. However, the usefulness of particular document collections for QE was not measured and taken into account during retrieval. In this work, we investigate two automatic approaches that measure and leverage the usefulness of document collections when exploiting multiple document collections to improve query representation. These two approaches are based on resource selection and learning to rank techniques, respectively. We evaluate our approaches using the TREC Medical Records track's test collection. Our results show the potential of the proposed approaches, since they can effectively exploit 14 different document collections, including both domain-specific (e.g. MEDLINE abstracts) and generic (e.g. blogs and webpages) collections, and significantly outperform existing effective baselines, including the best systems participating at the TREC Medical Records track. Our analysis shows that the different collections are not equally useful for QE, while our two approaches can automatically weight the usefulness of expansion terms extracted from different document collections effectively.

Categories and Subject Descriptors: H.3.3 [Information Storage & Retrieval]: Information Search & Retrieval
Keywords: Patient Search, Query Expansion

1. INTRODUCTION

To improve the healthcare service quality, electronic medical records (EMRs) are used to document the medical conditions (e.g. symptom, treatment) of patients visiting a hospital. These EMRs can also be leveraged in medical research. For example, using a patient search system, EMRs can help to identify cohorts of patients suitable for particular clinical trials. In this paper, we tackle the patient search task. In this task, healthcare practitioners describe the medical conditions of patients of interest as a query and the search system retrieves patients based on their relevance towards the stated medical conditions.

Existing work showed that the complexity of medical terminology makes patient search a challenging task [7, 10, 18]. For instance, patients whose medical records state that they are suffering from "deafness" may not be retrieved for a query to find patients with "hearing loss", even though "deafness" and "hearing loss" share the same meaning. Limsopatham et al. [9, 11], as well as Qi and Laquerre [18] dealt with this problem by representing queries and medical records in the forms of medical concepts. As a result, "deafness" and "hearing loss" were represented with the same concept. On the other hand, several approaches exploited various document collections to improve query representation (e.g. [5, 10, 25]). For example, Limsopatham et al. [10] used the relationships between medical concepts found in medical resources, such as UMLS[1] and MeSH[2], to improve the representation of medical conditions in the queries. King et al. [5], whose system achieved the best retrieval performance at the TREC 2011 Medical Records track [21], expanded the queries with terms extracted from the medical record collection itself. Later, Zhu et al. [25] used a query expansion (QE) technique to extract expansion terms from four different document collections, including web documents, medical records and two sets of medical articles. In particular, Zhu et al. [25] suggested that using expansion terms extracted from all of those four collections to expand a query is more effective than using expansion terms from each collection individually. In contrast, we hypothesise that the usefulness of different document collections for QE could be estimated and more effectively exploited when ranking patients based on the relevance of their medical records.

In this work, we investigate two approaches for modelling the likelihood that expansion terms extracted from each document collection are effective for improving query representation. The first approach adapts a resource selection technique [3] to measure the likelihood that a document collection can provide good expansion terms, and then uses the measured likelihood to weight the expansion terms extracted from that collection. On the other hand, the second approach combines the relevance scores computed from the expanded query using each of the document collections by using a learning to rank (LTR) technique (e.g. [15, 23]) to learn an effective combination.

Using the TREC 2011 and 2012 Medical Records track's test collection [20, 21], we evaluate our two proposed approaches when applied with a QE technique to exploit 14 different document collections that are either generic or domain-specific. Our results show that both approaches are effec-

*The work was done at the University of Glasgow.

CIKM'15, October 19–23, 2015, Melbourne, Australia.

DOI: http://dx.doi.org/10.1145/2806416.2806614.

[1] http://www.nlm.nih.gov/research/umls/
[2] http://www.ncbi.nlm.nih.gov/mesh/

tive, as they could significantly outperform existing effective baselines with and without QE. In particular, the resource selection-based approach outperforms the approach of Zhu et al. [25] by up to 32%.

2. RETRIEVAL MODELS

We first describe the retrieval models used for ranking patients based on the relevance of their medical records (Section 2.1), and for extracting expansion terms from a particular document collection (Section 2.2).

2.1 The Patient Ranking Model

Existing work (e.g. [5, 10, 25]) used either *a patient model* or *a two-stage model* to rank patients based on the relevance of their medical records [12]. The patient model ranks patients based on the relevance scores of the concatenations of medical records associated to particular patients, while the two-stage model aggregates the relevance scores of the medical records using an aggregate function (e.g. expCombSUM [14]) to rank their associated patients. In this work, we use the expCombSUM voting technique [14] to rank patients, as it has been shown to be effective in previous works (e.g. [12, 13]). In particular, expCombSUM calculates the relevance score of patient p towards query Q, as follows:

$$score_{patient}(p, Q) = \sum_{d \in R(Q) \cap profile(p)} e^{score(d,Q)} \quad (1)$$

where $R(Q)$ is a ranking of the medical records retrieved using query Q, $R(Q) \cap profile(p)$ is the set of medical records in ranking $R(Q)$ that are also associated to patient p; $score(d, Q)$ is the relevance score of medical record d given the query Q. We follow [13] and use DFR DPH [2] to calculate $score(d, Q)$, and limit the number of voting records ($|R(Q)|$) to 5,000.

2.2 The Query Expansion Model

Query expansion (QE) models (e.g. DFR Bo1 [1] or a relevance model [8]) extract expansion terms from a document collection by firstly retrieving the top N ranked documents, and then extract the top k most informative terms from those documents as expansion terms. Traditional QE uses the targeted collection to extract the expansion terms, while *external QE* (e.g. [4]) extracts the expansion terms from the documents retrieved from an external collection. In this work, we use both the targeted and external collections to expand a query. We deploy the DFR Bo1 model to extract the top k expansion terms from each collection, since it has been shown to be effective for this patient search task [10]. In particular, we follow [10] and extract the 10 most informative terms, as well as their weight, from the top 3 ranked documents in each collection (i.e. $N = 3$ and $k = 10$).

3. THE USEFULNESS OF COLLECTIONS

Sections 3.1 and 3.2 introduce our two proposed approaches to model the usefulness of different document collections when exploiting their extracted expansion terms.

3.1 Resource Selection

Resource selection techniques have been used to select collections that are likely to contain relevant documents, so that a retrieval system can focus on those collections during retrieval [3]. Our first approach adapts a resource selection technique to measure the likelihood that expansion terms extracted from a particular collection are effective for QE. In particular, it is intuitive that expansion terms extracted from collections that are related to the original query are more likely to be useful than those extracted from unrelated collections. When taking into account the likelihood that expansion term t_k extracted from collection c_i is effective for QE, the relevance score of medical record d towards query Q (i.e. $score(d, Q)$) can be calculated as follows:

$$score(d, Q) = \sum_{t_j \in Q} score(d, t_j) + \sum_{c_i \in C} w_c(c_i) \cdot \sum_{t_k \in Qc_i} w_t(t_k) \cdot score(d, t_k) \quad (2)$$

where $score(d, t_j)$ and $score(d, t_k)$ can be calculated using any retrieval function (e.g. DFR DPH [2]), C is the set of used collections, $w_c(c_i)$ is a collection weight, Qc_i is a set of expansion terms extracted from collection c_i, and $w_t(t_k)$ is the weight of expansion term t_k, which can be computed from each document collection using a QE model. Note that in the approach of Zhu et al. [25], $w_c(c_i)$ is equally set to 1 for all collections c_i.

We use the CORI resource selection algorithm [3] to calculate the collection weight $w_c(c_i)$ in Equation (2). Indeed, CORI is calculated as a combination of the score $p(t_j|c_i)$ of each term t_j in query Q, using a probabilistic operator. Using the operator AND, OR or SUM, the CORI score (i.e. $w_c(c_i)$) of collection c_i can be calculated as follows [3]:

$$CORI_{AND}(c_i, Q) = \prod_{t_j \in Q} p(t_j|c_i) \quad (3)$$

$$CORI_{OR}(c_i, Q) = 1 - \prod_{t_j \in Q} (1 - p(t_j|c_i)) \quad (4)$$

$$CORI_{SUM}(c_i, Q) = \frac{\sum_{t_j \in Q} p(t_j|c_i)}{|Q|} \quad (5)$$

where $p(t_j|c_i)$ is calculated as follows [3]:

$$p(t_j|c_i) = b + (1 - b) \cdot T \cdot I \quad (6)$$

$$T = \frac{df}{df + 50 + 150 \cdot \frac{cw}{avg_{cw}}} \quad (7)$$

$$I = \log\left(\frac{|C| + 0.5}{cf}\right) / \log(|C| + 1) \quad (8)$$

df is the number of documents in collection c_i that contain term t_j, cw is the number of terms in collection c_i, avg_{cw} is the average number of terms in each collection, $|C|$ is the number of used collections, cf is the number of collections containing term t_j, and b is the default belief, which is set to 0.4 as recommended in [3].

3.2 Learning to Rank

Our second approach uses an LTR technique to combine rankings produced from the query expanded using each collection separately. In general, the LTR techniques aim to learn an effective ranking model from a set of features using training data. As we aim to leverage QE from multiple collections to improve the patient search performance, our features are the relevance scores computed from the query expanded using each of the used document collections. For a linear LTR technique, such as Automatic Feature Selection (AFS) [15], the weights of these features can be viewed as the usefulness for QE of the associated document collections.

We deploy four existing effective LTR techniques (namely, AdaRank [23], AFS [15], Coordinate Ascent (CA) [16] and LambdaMART [22]) that aim to optimise a targeted evaluation measure (e.g. MAP) during training. Indeed, these four

Table 1: List of document collections used for QE.

Collection	Description[3]
EMRs11	100,710 medical records from the TREC 2011 & 2012 Medical Records track
Genomics04	4,591,008 MEDLINE abstracts from the TREC 2004 & 2005 Genomics track
Genomics06	162,259 full-text medical articles from the TREC 2006 & 2007 Genomics track
eHealth12	∼1M documents crawled from health and medicine sites by the CLEF/ShARe eHealth evaluation lab 2012
OHSUMED	348,566 MEDLINE references from TREC-9
ClueWeb09B	∼50M English webpages from the TREC 2009 Web track
Wiki09	∼21M Wikipedia pages of ClueWeb09B from the TREC 2009 Web track
WT10G	1,692,096 English web documents from the TREC 2000 Web track
WT2G	250,000 English web documents from the TREC 1999 Web track
Blog06	3,215,171 blog posts from the TREC 2006 Blog track
Blog08	28,488,767 blog posts from the TREC 2008 Blog track
CERC	370,715 web documents from the TREC 2008 Enterprise track
W3C	331,037 documents of `w3c.org` sites from the TREC 2005 Enterprise track
Disk45	528,155 documents of Disks 4&5 from TREC-8

techniques deploy different types of algorithms to learn a suitable ranking model from a set of features. AdaRank [23] applies a boosting technique to optimise the targeted evaluation measure by considering each feature as a weak ranker. AFS [15] and CA [16] apply a greedy algorithm to learn an effective linear combination of features by using different underlying optimisation techniques. Our used implementations of AFS and CA deploy simulated annealing [6] and coordinate ascent, respectively, when learning the feature weights. LambdaMART [22] deploys boosted regression trees to find an effective combination of features that optimises a targeted evaluation measure.

4. EXPERIMENTAL SETUP

Test Collection: To evaluate the two proposed approaches, we use the test collection provided by the TREC Medical Records track [20, 21]. The task is to retrieve patient visits relevant to a given query. A patient visit contains medical records related to a particular visit to the hospital by a patient. Due to privacy concerns, a patient visit is used to represent a patient [20, 21]. The topic set includes 34 and 47 queries from TREC 2011 and 2012, respectively. We report the retrieval performance in terms of the track primary measures, which are bpref and infNDCG for TREC 2011 and 2012, respectively.

Expansion Collections: For reproducibility, we leverage 14 widely available document collections for QE, which include both generic and domain-specific collections, as described in Table 1.

Retrieval Toolkits: We conduct experiments using the Terrier platform [17][4], applying Porter's English stemmer and removing stopwords. We use the models discussed in Sections 2.1 and 2.2 to rank patients and to extract expansion terms, respectively. For the LTR techniques, we use the implementation of RankLib[5] with the default setting. In addition, as the LTR techniques require a set of training data to learn a model, we use the queries from TREC 2011 to train an LTR model when testing on TREC 2012, and vice versa. In addition, as the targeted measure of TREC 2011 and 2012 are different, we follow [24] and train the learned model to optimise the MAP measure.

[3] Please consult the TREC/CLEF overview papers for details about each document collection. [4] `http://terrier.org`
[5] `http://www.lemurproject.org/ranklib.php`

5. EXPERIMENTAL RESULTS

We compare the retrieval performances of our proposed approaches with four categories of baselines. The first two categories of baselines are basic baselines that either do not apply QE or use expansion terms from only one of the 14 used collections. The third and the fourth categories of baselines are more advanced. In particular, the third category of baselines uses expansion terms extracted from all 14 collections (as in [25]). The fourth category of baselines firstly retrieves patients for the query that is expanded using each of the 14 collections, and then uses a data fusion technique [19] (i.e. CombSUM, CombMNZ, expCombSUM or expCombMNZ) to combine the relevance scores of the patients. Note that the CombSUM data fusion technique is in some sense similar to the approach of [25] in that the relevance scores of expansion terms extracted from different collections are equally combined to rank patients. However, CombSUM is arguably more robust to topic drift, since the focus of each expanded query is typically still on the original query.

Table 2 shows our experimental results in terms of bpref and infNDCG for TREC 2011 and 2012, respectively. First, we discuss the performance of the baselines that apply QE on each collection individually. From Table 2, we observe that, for TREC 2011, 12 out of 14 collections are useful for improving the query representation, as expansion terms extracted from these collections improve the retrieval performance over the baseline that does not apply QE (i.e. 'No-QE'). Meanwhile, 10 out of 14 collections are effective for TREC 2012. In general, the expansion terms extracted from all medical-related collections (e.g. EMRs11, Genomics04 and Genomics06) effectively improve the retrieval performance for both TREC 2011 and 2012. However, surprisingly ClueWeb09B and WT10G, which are generic collections, are the most effective collections for QE for TREC 2011 and 2012, respectively, and even outperform traditional QE, which expands the queries using the targeted collection (i.e. EMRs11). In particular, leveraging expansion terms from ClueWeb09B is overall the most effective and significantly (paired t-test, $p < 0.05$) outperforms the 'No-QE' baseline for both TREC 2011 and 2012 (bpref 0.5323 vs. 0.4871 and infNDCG 0.4446 vs. 0.4167). Nevertheless, we observe that expanding queries with terms extracted from all collections (i.e. 'All Collections') is not effective. This differs from the conclusion reported in [25]. However, note that our experiments consider more document collections both in number and variety. Meanwhile, the data fusion baselines are more effective than the 'No-QE' and 'All Collections' baselines.

When considering the performance of our two proposed approaches, we observe that they significantly (paired t-test, $p < 0.05$) outperform both the 'No-QE' and 'All Collections' baselines. Indeed, our adaptation of CORI (i.e. CORI_{SUM} and CORI_{OR}) to weight the expansion terms from each collection (i.e. Equation (2)) outperforms all of the evaluated approaches, including when applying QE with the ClueWeb09B collection. This shows that expansion terms extracted from different document collections are not equally useful, and our adaptation of CORI could estimate the effectiveness of expansion terms extracted from different collections. However, CORI_{AND} is less effective than CORI_{SUM} and CORI_{OR} since it applies the stronger constraint that a collection should contain all of the query terms. It is of note that both CORI_{SUM} and CORI_{OR} perform better than the best TREC 2011 system [5], and comparably to the top performing systems in TREC 2012 [20], without deploying other well-known performance boosting techniques for pa-

Table 2: Retrieval performances on TREC 2011 and 2012 Medical Records track of different QE approaches. Statistical significance (paired t-test) at $p < 0.05$ over the baseline that does not apply QE and the baseline that uses all expansion terms extracted from the 14 used document collections are denoted $^{\oplus}$ and $^{\odot}$, respectively.

Approaches	2011 (bpref)	2012 (infNDCG)
No-QE	0.4871	0.4167
+ EMRs11	0.5264$^{\oplus}$	0.4408$^{\oplus}$
+ Genomics04	0.5214$^{\oplus}$	0.4399$^{\oplus}$
+ Genomics06	0.5239$^{\oplus}$	0.4399
+ eHealth12	0.5185$^{\oplus}$	0.4372
+ OHSUMED	0.4978	0.4333
+ ClueWeb09B	**0.5323**$^{\oplus}$	0.4446$^{\oplus}$
+ Wiki09	0.5268$^{\oplus}$	0.4155
+ WT10G	0.5215$^{\oplus}$	**0.4454**
+ WT2G	0.5105$^{\oplus}$	0.4343
+ Blog06	0.5045	0.4386
+ Blog08	0.5118$^{\oplus}$	0.4159
+ CERC	0.4657	0.4159
+ W3C	0.4671	0.4117
+ Disk45	0.4942	0.4182
+ All Collections [25]	0.4833	0.3551
Data Fusion		
CombSUM	0.5074$^{\oplus}$	**0.4385**$^{\oplus\odot}$
CombMNZ	0.5076$^{\oplus}$	**0.4385**$^{\oplus\odot}$
expCombSUM	**0.5084**$^{\oplus}$	0.4350$^{\oplus\odot}$
expCombMNZ	0.5080$^{\oplus}$	0.4356$^{\oplus\odot}$
Resource Selection		
CORI$_{SUM}$	**0.5597**$^{\oplus\odot}$	0.4603$^{\oplus\odot}$
CORI$_{OR}$	0.5594$^{\oplus\odot}$	**0.4689**$^{\oplus\odot}$
CORI$_{AND}$	0.4958$^{\oplus}$	0.4216$^{\odot}$
Learning to Rank		
AdaRank	0.4997	0.4343$^{\odot}$
AFS	0.5212$^{\oplus}$	0.4330$^{\odot}$
CA	**0.5289**$^{\oplus}$	**0.4475**$^{\oplus\odot}$
LambdaMART	0.3590	0.3057

tient search such as negation handling [20, 21]. Meanwhile, our approach that uses LTR techniques could not outperform the ClueWeb09B baseline. In particular, the most effective LTR technique is CA, which achieves bpref 0.5298 and infNDCG 0.4475. The other LTR techniques, e.g. LambdaMART, are less effective. This could be due to the small number of available queries for training the learned models.

When analysing the expansion terms extracted from different collections, we observe that the expansion terms from some collections are not related to the query medical conditions. For instance, for the query "hearing loss", our approach gives low weights to W3C and CERC, from which the extracted expansion terms, such as 'individual', 'argument', 'plan' and 'strategy' were off-topic. Meanwhile, collections such as ClueWeb09B and Wiki09, which obtain high weights, provide expansion terms that are more related to the query e.g. 'earwax', 'sensorineural' and 'deaf'. In contrast, expansion terms extracted from EMRs11, which received a medium weight from our approach, also include an off-topic term (i.e. 'woman'), after a list of related terms (e.g. 'cerumen', 'ear', 'canal'). This might be the reason why QE using ClueWeb09B or Wiki09 is more effective than when applied to EMRs11, since highly ranked documents from the web might contain more effective related terms to "hearing loss" than highly ranked medical records, as the latter could also mention other health information.

6. CONCLUSIONS

We introduced the use of resource selection and learning to rank techniques to weight the expansion terms extracted from particular document collections differently. Our experimental results conducted on the TREC Medical Records track's test collection showed that the two approaches were more effective than an existing approach that used all the expansion terms extracted from all of the used collections equally. Specifically, our approach based on a resource selection technique (i.e. CORI$_{SUM}$ and CORI$_{OR}$) was shown to be the most effective and significantly outperformed the aforementioned baseline by up to 32% (infNDCG 0.4689 vs. 0.3551). This shows that expansion terms extracted from different document collections are not equally useful and that our proposed approaches can automatically and effectively weight these expansion terms during retrieval.

7. REFERENCES

[1] G. Amati. Probabilistic Models for Information Retrieval based on Divergence from Randomness. *PhD thesis*. University of Glasgow, 2003.

[2] G. Amati, E. Ambrosi, M. Bianchi, C. Gaibisso and G. Gambosi. FUB, IASI-CNR and University of Tor Vergata at TREC 2007 Blog Track. *TREC*'07.

[3] J. Callan. Distributed Information Retrieval. *Advances in Information Retrieval*, 2000.

[4] F. Diaz and D. Metzler. Improving the Estimation of Relevance Models Using Large External Corpora. *SIGIR*'06.

[5] B. King, L. Wang, I. Provalov and J. Zhou. Cengage Learning at TREC 2011 Medical Track. *TREC*'11.

[6] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi. Optimization by simulated annealing. *Science*, 1983.

[7] B. Koopman and G. Zuccon. Why assessing relevance in medical IR is demanding. *MedIR at SIGIR*'14.

[8] V. Lavrenko and W. B. Croft. Relevance Based Language Models. *SIGIR*'01.

[9] N. Limsopatham, C. Macdonald, and I. Ounis. A Task-Specific Query and Document Representation for Medical Records Search. *ECIR*'13.

[10] N. Limsopatham, C. Macdonald and I. Ounis. Inferring Conceptual Relationships to Improve Medical Records Search. *OAIR*'13.

[11] N. Limsopatham, C. Macdonald, and I. Ounis. Learning to Combine Representations for Medical Records Search. *SIGIR*'13.

[12] N. Limsopatham, C. Macdonald and I. Ounis. Learning to Selectively Rank Patients' Medical History. *CIKM*'13.

[13] N. Limsopatham, C. Macdonald, I. Ounis, G. McDonald and M. Bouamrane. University of Glasgow at Medical Records track 2011: Experiments with Terrier. *TREC*'11.

[14] C. Macdonald and I. Ounis. Voting for candidates: adapting data fusion techniques for an expert search task. *CIKM*'06.

[15] D. Metzler. Automatic feature selection in the markov random field model for information retrieval. *CIKM*'07.

[16] D. Metzler and W. B. Croft. Linear feature-based models for information retrieval. *J. of Information Retrieval*, 2007.

[17] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald and C. Lioma. Terrier: A High Performance and Scalable Information Retrieval Platform. *OSIR at SIGIR*'06.

[18] Y. Qi and P.-F. Laquerre. Retrieving Medical Records with "sennamed": NEC Labs America. *TREC*'12.

[19] J. A. Shaw and E. A. Fox. Combination of multiple searches. *TREC*'94.

[20] E. Voorhees and W. Hersh. Overview of the TREC 2012 Medical Records Track. *TREC*'12.

[21] E. Voorhees and R. Tong. Overview of the TREC 2011 Medical Records Track. *TREC*'11.

[22] Q. Wu, C. J. C. Burges, K. Svore and J. Gao. Adapting Boosting for Information Retrieval Measures. *J. of Information Retrieval*, 2007.

[23] J. Xu and H. Li. Adarank: A boosting algorithm for information retrieval. *SIGIR*'07.

[24] D. Zhu and B. Carterette. An adaptive evidence weighting method for medical record search. *SIGIR*'13.

[25] D. Zhu, S. Wu, B. Carterette and H. Liu. Using Large Clinical Corpora for Query Expansion in Text-based Cohort Identification. *J. of Biomedical Informatics*, 2014.