INFORMATION RETRIEVAL FOR REDUCING MANUAL EFFORT IN BIOMEDICAL AND CLINICAL RESEARCH

by

Dongqing Zhu

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science

Fall 2014

© 2014 Dongqing Zhu
All Rights Reserved

UMI Number: 3685168

All rights reserved

INFORMATION TO ALL USERS
The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript
and there are missing pages, these will be noted. Also, if material had to be removed,
a note will indicate the deletion.

UMI 3685168

Published by ProQuest LLC (2015). Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.
All rights reserved. This work is protected against
unauthorized copying under Title 17, United States Code

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

INFORMATION RETRIEVAL FOR REDUCING MANUAL EFFORT IN BIOMEDICAL AND CLINICAL RESEARCH

by

Dongqing Zhu

Approved: _____

Errol L. Lloyd, Ph.D.
Chair of the Department of Computer and Information Sciences

Approved: _____

Babatunde Ogunnaike, Ph.D.
Dean of the College of Engineering

Approved: _____

James G. Richards, Ph.D.
Vice Provost for Graduate and Professional Education

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Benjamin A. Carterette, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Vijay Shanker, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Cathy Wu, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: _____

Stephen Wu, Ph.D.
Member of dissertation committee

ACKNOWLEDGEMENTS

First of all, I want to express my deepest gratitude towards my advisor Prof. Ben Carterette who brought me onto this fruitful journey. I am very grateful for his patience and support throughout my PhD study. Not only did he kindly provide me the freedom to explore any research ideas, he also carefully guided me and always kept me on the right track. It would not be possible to reach this point without his generous help.

Secondly, I am extremely grateful to Prof. Vijay Shanker, Prof. Cathy Wu, and Prof. Stephen Wu for serving my dissertation committee, sharing their time and expertise, and providing me with insightful comments and suggestions.

I would like to acknowledge my mentors and collaborators from Mayo Clinic, Dr. Stephen Wu, Dr. Dingcheng Li, and Dr. Hongfang Liu, for their tremendous help and guidance. In particular, I am very grateful to Dr. Stephen Wu who found me at the TREC conference and provided me with the internship opportunity at Mayo Clinic, without which I would not be able to explore many interesting ideas for this dissertation.

I also want to thank all my friends and labmates at UD. I learned a lot and received generous help from them. The good memories we enjoyed together will never fade.

Finally, I want to thank my parents and my wife Yan for their love, support, encouragement, and understanding.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Medical professionals leverage health-related data to address questions and support decision-makings. However, many of these medical tasks require intensive manual effort in identifying useful information in the noisy data. The rapid growth of data is making these tasks more and more costly and time-consuming.

In this thesis, we develop effective medical information retrieval (IR) systems to reduce search-related manual work for three representative medical related tasks, namely electronic medical records (EMR) based cohort identification, Medical Subject Headings (MeSH) indexing, and gene ontology annotation (GOA).

For cohort identification, we improve the search precision and recall from three aspects: 1) we design a multi-level evidence aggregation strategy for effective merging and scoring of the distributed evidence in EMR; 2) we develop a novel statistical IR model that significantly alleviates two medical language related issues in medical IR; 3) we further enhance the search performance by effectively incorporating domain knowledge into our system.

For MeSH indexing and GOA, we demonstrate how to use IR to address specific needs. In particular, we investigate different query formulation methods and explore various ways in which IR work together with other techniques such as Natural Language Processing and Machine Learning.

Chapter 1

INTRODUCTION

The increasing availability and usage of health related (e.g., biomedical and clinical) data have been driving innovations in health care and transforming our understanding of wellness and disease. Medical professionals use these data to address questions and support decision-makings. However, many of these medical tasks (e.g., cohort identification, data curation, etc.) require a lot of manual effort in identifying the useful information in the abundant and noisy data, and they cannot be fully automated by computers. With the rapid growth of data, these labor-intensive tasks become more and more time-consuming and costly.

Fortunately, the information retrieval (IR) technology is playing a critical role in helping healthcare professionals accomplish medical tasks in an efficient way. Specifically, IR helps to reduce the manual work for healthcare people by finding the most relevant information from the vast amount of data in a short period of time. However, there are a few challenges in searching clinical and biomedical data that could prevent us from gaining the most out of IR.

First, the retrieving units are not necessarily standard documents. For example, in cohort identification the retrieval units are patients whose relevance supporting evidence can spread across all personalized data, such as lab test results, medical history, radiology report, etc. We need to find an effective way to score and aggregate the multiple components for each retrieving unit.

Second, the usage and vocabulary of medical language is very different from the general English. In particular, the synonymy and polysemy are prevalent in medical language. In addition, there are a number of other problems associated with the

usage of words, especially in clinical narratives, e.g., negations, spelling errors, elliptical sentences, grammatical incompleteness, and non-lexicon words [44]. These medical language related problems have a direct and huge impact on the search precision and recall.

Third, medical domain knowledge is helpful, but how to effectively incorporate it into IR systems is usually not straightforward.

Last but not least, different medical tasks usually need specific design and tailor of their search methods. There is no one-size-fits-all approach. In many cases, we need to combine IR with other techniques such as Natural Language Processing (NLP) and Machine Learning to accomplish the task.

All the things mentioned above can affect the search system performance which further determines how much cost and time we can save for accomplishing the health and biomedical tasks. Therefore, in this thesis work we develop novel medical IR systems to tackle these problems. In particular, we will design systems for three specific medical-related tasks, namely electronic medical records (EMR) based cohort identification, Medical Subject Headings (MeSH) indexing, and gene ontology annotation (GOA). The goal is to reduce manual effort involved in these processes and allow healthcare professionals to focus on other things that are more important. This will also contribute to better patient care in the long term.

1.1 Three Biomedical Tasks

In this section, we will briefly describe the three health related tasks along with the subtopics of this thesis.

1.1.1 EMR-based Cohort Identification

Cohort identification is the task of finding patients that all meet certain inclusion criteria for clinical studies. In the past, cohort identification has been a costly endeavor, requiring hours of trained expertise to accomplish manual chart reviews. With the gradual adoption of EMR, this problem has been mitigated by NLP techniques, such as

entity recognition and information extraction. However, this kind of methods have its limitations. First, it requires extensive collaboration between NLP experts and medical researchers to turn the clinical information needs (or patient matching criteria) into a good set of (usually complicated) matching patterns, and this has to be done manually for each specific information need. Second, the recall depends heavily on the quality of the pattern set. If we need a large number of relevant patients, the manual work grows dramatically for developing and maintaining the pattern set.

Thus, we will investigate how to use advanced IR techniques to minimize the labor work in cohort identification. In particular, in Chapter 3 we will explore ways of expanding, aggregating, and scoring evidence that resides in different parts of patients' EMR for patient searching and ranking;

In Chapter 4, we will design novel retrieval models for alleviating the polysemy and synonymy issues (i.e., two medical language related issues) that will severely compromise search precision and recall;

In Chapter 5, we will investigate how to effectively leverage medical domain knowledge to improve search results.

1.1.2 MeSH Indexing

MEDLINE[1] is the U.S. National Library of Medicine's (NLM) premier bibliographic database that contains over 19 million references to journal articles in life sciences with a concentration on biomedicine. A distinctive feature of MEDLINE is that the records are indexed with NLM Medical Subject Headings (MeSH) and by a relatively small group of highly qualified domain experts at NLM.

Currently, there are about 0.7 million new journal articles being added to the MEDLINE database each year. Manually indexing new articles is very labor-intensive and time-consuming. In addition, the indexing consistency is hard to control. On the other hand, we need to include these new articles into the database in a timely fashion so that the latest research outcomes can quickly become available to the public.

[1] http://www.nlm.nih.gov/pubs/factsheets/medline.html

Thus, in Chapter 6 we will design search-based systems that can suggest and rank MeSH terms to assist domain experts in MEDLINE article indexing and to help reduce manual work and indexing time.

1.1.3 Gene Ontology Annotation

The gene ontology (GO) provides a set of concepts for annotating functional descriptions of genes and proteins in biomedical literature. The resulting annotated databases are useful for large-scale analysis of gene products. However, performing gene ontology annotation (GOA) requires expertise from well-trained human curators. Due to the fast expansion of biomedical data, GOA becomes extremely labor-intensive and costly. Thus, in Chapter 7 we will investigate how to use information retrieval techniques for GO term prediction. Although this task is very similar to MeSH indexing, we want to show that different tasks need specific designing and tailoring of the search methods.

We have briefly introduced the topics of Chapters 3 to 7 above. Chapter 2 will highlight related work while Chapter 8 will conclude the thesis and discuss future work.

1.2 Evaluation Metrics

The evaluation metrics are important for understanding the system design, and we will use them frequently to evaluate our retrieval systems in Chapters 3-7. Therefore, we describe the evaluation metrics early in the thesis:

1) **P10**, which measures the proportion of relevant documents among the top 10 retrieved.

2) **MAP**, as one of the most standard evaluation measures among TREC community, provides a single-figure measure of quality across recall levels [26, 76]. If $\{d_1,...,d_j\}$ is the set of relevant documents for an information need $q \in Q$, then MAP is defined as:

$$\text{MAP}(Q) = \frac{1}{|Q|} \sum_{q \in Q} \frac{\sum_{d \in \{d_1,...,d_j\}} \text{Precision}(\text{rank}(d))}{|\{d_1, ..., d_j\}|}, \tag{1.1}$$

where Precision(k) is the proportion of relevant documents among the top k retrieved. There is another related metric called infAP (inferred AP) which is used to approximate average precision when the relevance judgments are incomplete [122].

3) **bpref**, defined as:

$$\text{bpref} = \frac{1}{R}\sum_{r}(1 - \frac{|n \text{ ranked higher than } r|}{\min(R, N)}), \tag{1.2}$$

where R is the number of judged relevant documents, N is the number of judged irrelevant documents, r is a relevant retrieved document, and n is a member of the first R irrelevant retrieved documents. bpref computes a preference relation of whether judged relevant documents are retrieved ahead of judged irrelevant documents. It is based on the relative ranks of judged documents only [21].

4) **R-prec**, which is the precision after R documents have been retrieved (also known as the break-even point), where R is the number of relevant documents for the topic. It de-emphasizes the exact ranking of the retrieved relevant documents, though it is highly correlated to MAP in practice.

5) **NDCG**, which stands for normalized discounted cumulative gain and is designed for situations of nonbinary notions of relevance [76]. The cumulative gain (CG) of a ranked list of size k is the total gain contributed from each relevant document in this list. When a relevant document is ranked low in the rank list its gain towards CG will be discounted, which lead to the the discounted cumulative gain (DCG). If we normalize DCG by the DCG from the perfect ranking (i.e., when all relevant documents appear at the top of the ranked list), we will have the NDCG which is formally defined as

$$\text{NDCG}(q, k) = \frac{1}{\text{DCG}_{\text{perfect}}(q, k)} \cdot \sum_{i=1}^{k} \frac{2^{R(q,d_i)} - 1}{log(1+i)}, \tag{1.3}$$

where q is the query, $R(q, d_i)$ is the relevance score of document at rank i, and $log(1+i)$ is the discounting factor.

Note that in the thesis, the evaluation scores are averaged over all queries in a system run.

In fact, for this thesis the ideal way to evaluate our retrieval systems is to measure the time reduced for each specific task. Though due to limited resources we are not available to take this ideal approach, we believe that the standard IR evaluation metrics, such as those described above, can indirectly reflect the improvement on time and efficiency. Intuitively, good IR systems can filter away most of the non-relevant information which would otherwise be examined and removed manually. In addition, several IR evaluation workshops (as will be described in Section 2.2) also use these standard metrics to evaluate IR systems that perform the same or similar tasks.

Chapter 2

RELATED WORK

In this chapter, we will highlight the related work in the fields of traditional information retrieval (IR) and domain-specific IR .

2.1 Information Retrieval

Information retrieval is a very broad field, containing topics on representation, storage, retrieval, ranking, evaluation, etc. of various media types, such as web pages, images, and videos. In this section, we focus on reviewing relevant work on retrieval models for text-based documents as they are the underpinning of our thesis work.

Before moving on, we first introduce some terminology. A *document* (D) in information retrieval refers to the unit used in indexing and retrieval. It can be of different media types or at different levels of granularity for a given type (e.g., books, chapters, paragraphs, and sentences for text-based documents). A *term* (t) is the basic element that constitutes a text-based document. A *collection* ($\mathcal{C}$) is a set of documents used to address users' requests. Each request is an *information need*, i.e., a topic the user desires to know more about. A user communicates an arbitrary information need via a *query* (Q) to the search engine, and we call this an *ad hoc retrieval* task. A relevant document is the one that the user perceives as containing information of value with respect to their personal information need [76].

Next, we briefly review several representative IR models.

2.1.1 Basic Retrieval Models

Boolean Retrieval Model

The simplest retrieval model is the boolean retrieval model in which we pose queries as terms combined by boolean operators AND, OR, and NOT. In this model, a document is essentially a set of terms, and the search engine returns an unordered list of documents.

Vector Space Model

In contrast, the vector space model (VSM) allows free-text queries (i.e., terms without boolean operators) and returns an ordered result set [94]. In VSM, we represent documents and queries as vectors of features. Each feature is a *term weight* associated with a term in the vocabulary and calculated by a function of the term statistics in the document and the collection. The relevance score is usually computed by the cosine similarity between the query vector and the document vector.

One popular weighting scheme for assigning term weights is TF-IDF weighting. TF (term frequency) is the raw frequency of a term within a document [72]. It reflects the intuition that key terms conveying the meanings of a document tend to occur frequently within that document. IDF (inverse document frequency) estimates how discriminative a term is [53]. It is defined as:

$$\mathrm{IDF}_t = \log \frac{N}{\mathrm{df}_t}, \tag{2.1}$$

where N is the size of the collection $\mathcal{C}$, i.e., the total number of documents in $\mathcal{C}$, and df_t is the number of documents in $\mathcal{C}$ that contain term t.

Okapi BM25 Model

Another classical retrieval model using TF and IDF is the Okapi BM25 model [102]. It is a probabilistic model and the ranking function is defined as:

$$\mathrm{score}(D, Q) = \sum_{i=1}^{n} \mathrm{IDF}(q_i) \cdot \frac{\mathrm{tf}_{q_i,D} \cdot (k_1 + 1)}{\mathrm{tf}_{q_i,D} + k_1 \cdot (1 - b + b \cdot \frac{|D|}{\mathrm{avgdl}})}, \tag{2.2}$$

where q_i is the ith query term, $|D|$ is the document length, avgdl is the average document length in the collection, and k_1 and b are free parameters. The IDF(q_i) weight is usually defined as:

$$\text{IDF}(q_i) = \log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}, \tag{2.3}$$

where N is the total number of documents in the collection, and $n(q_i)$ is the number of documents containing q_i. The Okapi BM25 retrieval has been shown to perform well on a wide range of collections [112, 93].

Query Likelihood Model

Language modeling (LM) approach [88, 48, 17, 82] in which we assign probabilities over words produces a family of language-model-based retrieval methods. The very basic one is the query likelihood language model [88] which scores documents for queries as a function of the probability that query terms would be sampled (independently) from an urn containing all the words in that document. Formally, the scoring function is a sum of the logarithms of smoothed probabilities:

$$\text{score}(D, Q) = \log P(Q|D) = \sum_{i=1}^{n} \log \frac{\text{tf}_{q_i,D} + \mu \frac{\text{tf}_{q_i,C}}{|C|}}{|D| + \mu}, \tag{2.4}$$

where q_i is the ith term in query Q, n is the total number of terms in Q, $|D|$ and $|C|$ are the document and collection lengths in words respectively, $\text{tf}_{q_i,D}$ and $tf_{q_i,C}$ are the document and collection term frequencies of q_i respectively, and μ is the Dirichlet smoothing parameter [124]. Smoothing is a common technique to estimate the probability of unseen words in the documents [23, 125, 124, 87].

The query likelihood method has been shown to perform well on a variety of tasks, including adhoc retrieval [57, 124], cross-lingual information retrieval [120, 58], distributed information retrieval [101, 119], query difficulty prediction [27], passage retrieval [70], etc.

Relevance Model

In contrast with the query likelihood model, we have the document likelihood model as another LM-based model. Since there is much less text in the query to estimate a language model, it is necessary to incorporate relevance information into this model. One instance of the document likelihood model is the relevance model [59] which is built upon pseudo-relevance feedback. The relevance model is estimated according to:

$$P(w|\hat{\theta}_Q) \propto \frac{1}{|\mathcal{R}|} \sum_{D \in \mathcal{R}} P(w|\theta_D) P(Q|\theta_D), \tag{2.5}$$

where $\mathcal{R}$ is the set of pseudo-relevant document. It has been shown that the LM approach beat the BM25 method on a number of tasks [49].

2.1.2 Advanced Retrieval Models

Markov Random Field Model

In the query likelihood model, it is a strong assumption that query terms are generated independently from the document language model. In reality, related terms are likely to occur in close proximity to each other. The Markov random field (MRF) model [79] improves upon query likelihood model by incorporating term proximity information. It works by first constructing a graph that contains a document node, one node per query term, and edges that represent dependencies among nodes. Then, MRF models the joint distribution over the document random variable and query term random variables. The ranking function of the MRF model is of the form:

$$P_\Lambda(Q|D) \stackrel{rank}{=} \sum_{c \in T} \lambda_T f_T(c) + \sum_{c \in O} \lambda_O f_O(c) + \sum_{c \in O \cup U} \lambda_U f_U(c), \tag{2.6}$$

where T is defined to be the set of 2-cliques containing the document node and a query term node, O is the set of cliques involving the document node and two or more query terms that appear contiguously in the query, and U is the set of cliques involving the document node and two or more query terms that appear non-contiguously within the query. $f(c)$ is the feature function over clique c and λ's are the feature weights. MRF

model has been shown to consistently out-perform the standard unigram model across a range of TREC test collections [79, 80].

Weighted Sequential Dependence Model

The weighted sequential dependence model (WSD) [15] extends the Markov Random Field model (MRF) for information retrieval by automatically learning query concept weights. In particular, the λ parameters in Equation 2.6 are trained using both endogenous (based on target collection) and exogenous (based on external sources) features. WSD has been shown to significantly outperform MRF on a number of corpora [15]. However, WSD and its variants [100, 113] have limitations due to that the improvement comes from the weighting for the explicit query concepts. In other words, the latent concepts associated with the underlying information need are totally discarded. The parameterized query expansion (PQE) proposed by Bendersky et al. [16] addresses this issue by learning weights for both explicit and latent query concepts. It outperforms MRF and WSD on both newswire and web TREC corpora.

Positional Language Model

Another model that effectively uses term proximity information is the positional language model (PLM) [74]. The PLM is estimated for each position based on propagated counts of words within a document through a proximity-based density function. The document relevance score is calculated by scores of its PLMs. PLM is further improved by incorporating pseudo-relevance feedback [75].

Mixture of Relevance Models

Relevance modeling described at the end of Section 2.1.1 can be further improved upon by leveraging information in external document collections [33]. The mixture of relevance models (MRM) constructed in this way has been shown to achieve more stable MAP improvement than traditional pseudo-relevance feedback across a range of news and web collections. The term generation probability function is formally defined

by:

$$P(w|\hat{\theta}_Q) = \sum_{\mathcal{C}_i} k_{\mathcal{C}_i} \frac{P(\mathcal{C}_i)}{|\mathcal{R}_{\mathcal{C}_i}|} \sum_{D \in \mathcal{R}_{\mathcal{C}_i}} P(w|\theta_D) P(Q|\theta_D), \tag{2.7}$$

where $k_{\mathcal{C}_i}$ is the normalization factor for the relevance model estimate using collection $\mathcal{C}_i$.

Divergence from Randomness

The Divergence from Randomness (DFR) [7] paradigm is a generalization of Harter's 2-Poisson indexing-model [43]. The 2-Poisson model is based on the hypothesis that the level of treatment of the informative words is witnessed by an elite set of documents, in which these words occur to a relatively greater extent than in the rest of the documents. On the other hand, the frequency of words in non-elite documents tent to follow a random distribution.

2.2 Medical Information Retrieval

The medical information retrieval has received much attention in the IR research community during the past decade. In this section, we will highlight a few related work. Discussion on more specific work and approaches relevant to this thesis work will be deferred to each of the remaining chapters.

2.2.1 ImageCLEFMed

The Medical Image Retrieval Challenge Evaluation[1] (also known as ImageCLEFmed), as part of the Cross Language Evaluation Forum (CLEF[2]), started in the year of 2005. The goal of this workshop is to promote research on retrieval and classification of medical images. At the 10th year of this medical task, ImageCLEFmed was organized outside Europe for the first time at the annual AMIA (American Medical Informatics Association) meeting in 2013. There are four tasks in 2013 ImageCLEFmed,

[1] http://www.imageclef.org/2008/medical

[2] http://www.clef-initiative.eu

namely modality classification, compound figure separation, image-based retrieval, and case-based retrieval [29].

In the modality classification, images are classified into medical modalities and other images types, such as CT, X-Ray, PET, etc. The modality information is further used in image retrieval to prune the search results by removing the false positives. The compound figure separation task is to segment multi-panel figures into sub-figures so that the image retrieval can be more accurate. In the image-based retrieval task, the retrieving units are images but the queries are textual, semantic, or mixed queries. Participants can leverage both textual and visual features of the images to design their retrieval algorithms. In case-based retrieval, the task is to retrieve relevant medical cases and images given a brief case description along with patient demographics, limited symptoms, and test results.

2.2.2 TREC Genomics Track

The Text REtrieval Conference (TREC[3]) is an annual evaluation workshop held at the National Institute of Standards and Technology (NIST) with the goal of providing a common experimental setting for researchers that want to work on particular search tasks. Each year, there are up to 7 "tracks" devoted to different search tasks. Organizers provide documents and information needs to participants, ensuring that all participants are using the same data and working towards the same task. TREC organizers also oversee the collection of human relevance judgments, which are instrumental in understanding the effectiveness of a search system.

The Genomics Track [5] took place annually at TREC from year 2003 to 2007. The goal is to evaluate systems for information retrieval and related technologies in the genomics domain. The task scenario is that of a user seeking to acquire new knowledge in a sub-area of biology linked with genomics information (as opposed to a domain expert seeking information in his/her area of expertise). The main task is an ad hoc search task which evaluates manual, automatic, and interactive retrieval

[3] http://trec.nist.gov/

systems for full-text article retrieval and passage retrieval [45]. The other tasks include text summarization, categorization, and question-answering, which partially rely on the techniques used in the ad hoc search task.

2.2.3 Electronic Medical Record Search

As EMR become more prevalent, attempts have been made to transfer search engine technology to EMR retrieval for various applications [41]. The EMERSE (Electronic Medical Record Search Engine) system, as one of the earliest and successful non-commercial EMR search engines, has been used by medical professionals in a few hospitals, health centers, and clinics since its initial introduction in 2005 [41, 98]. EMERSE supports free-text queries and offers several advanced features such as query suggestion and collaborative search [126]. Though EMERSE has not achieved widespread adoption and there is little discussion about its search algorithms, a few interesting research work have been done using the EMERSE system:

Seyfried et al. [98] compared EMERSE-facilitated chart reviews with manual reviews, and concluded that using a well-designed EMR search engine for retrieving information in free-text EMR can provide significant time saving while preserving reliability.

Yang et al. [121] analyzed a query log of the EMERSE system recorded over the course of 4 years. One of their interesting findings is that the coverage of EMR query terms by a meta-dictionary (containing all terms in Unified Medical Language System, an English dictionary, and a medical dictionary) is much lower than the usual 85-90% coverage of Web queries by English dictionaries. Thus, they suggested seeking beyond the use of medical ontologies to enhance medical information retrieval.

Apart from these few attempts on improving EMR retrieval, methods emerging from research on information retrieval have not been well explored, largely due to the sensitivity of patient data, preventing its use by academic researchers. Fortunately, TREC organized a Medical Records Track in 2011 & 2012 making a set of real medical

records and human judgments of relevance to search queries available to the research community.

Most TREC participants of Medical Records Track used domain-specific knowledge to enhance retrieval. King et al. [55] annotated segments of the report text as having specific properties/features. They also identified and indexed terms of medical reports that appeared in the Unified Medical Language System (UMLS) Metathesaurus [20]. Meanwhile, they expanded original queries with related terms in UMLS and several commercial medical reference encyclopedias. Their best run improved their baseline by about 18%.

Goodwin et al. [39] used several external utilities for query expansion, including PubMed Central Open Access Subset (a small portion of PubMed Central database), Systematized Nomenclature of Medicine–Clinical Terms (SNOMED–CT) [103], and UMLS. They found that using these external medical-related sources together improved their baseline system performance.

Limsopatham et al. [67] made use of the ICD codes [84] in the reports and enriched reports with ICD code descriptions and related Wikipedia pages. They identified medical concepts in both documents and queries based on medical-domain ontologies in SNOMED-CT and Medical Subject Headings (MeSH), and expanded the concepts with nearby concepts in the ontology hierarchies (i.e., trees in MeSH, ICD, and SNOMED). They also obtained promising results.

Demner-Fushman et al. [31] expanded query terms with UMLS synonyms and expanded drug related terms using RxNorm and Google search. They also expanded terms in documents with their ancestors and children in the MeSH hierarchy. However, their knowledge-based Lucene [1] runs was worse than the baseline Lucene run, though they observed some improvement on their Essie [51] run. In a few other cases of using query expansion, Daoud et al. [28] used UMLS, Wu et al. [116] used disease and symptom descriptions from a healthcare website, and Schuemie et al. [97] used UMLS and DrugBank [114]. However, they all obtained very little or no improvement over their baseline runs.

2.2.4 ShARe/CLEF eHealth Evaluation Lab

The ShARe/CLEF eHealth Evaluation Lab is another CLEF workshop[4] that aims to improve the accessability of health data. Its goal is to develop processing methods and resources to enrich difficult-to-understand health text as well as their evaluation setting. There were three tasks in 2013: 1) identification of disorders from clinical reports and mapping of the SNOMED-CT disorders to UMLS codes, 2) mapping abbreviations and acronyms in clinical reports to UMLS codes, and 3) information retrieval to address questions patients may have when reading clinical reports [38]. Task 3 is very relevant to this thesis work. We will use the test collection of Task 3 to evaluate our retrieval systems in Chapter 4 where we try to tackle the medical language related issues in medical IR.

2.2.5 Biomedical Tools

Existing medical natural language processing (NLP) tools are handy for preprocessing raw clinical and biomedical text, and thus gives us more flexibility in designing our retrieval systems. Some of the well-known tools in biomedical field include MedLee [34], cTakes [96], MetaMap [9], and HITEx [123], which are specifically designed for recognizing medical terms and findings and mapping them to controlled vocabularies.

As we will see, we use the MetaMap tool frequently for in the thesis work. MetaMap was developed at the National Library of Medicine (NLM) to map biomedical text to concepts in the UMLS Metathesaurus. It has been used by many TREC Medical Records Track participants [90, 60, 81, 40]. Thus, we also use MetaMap so that our results can be compared with others. Next, we briefly describe how MetaMap works to get a sense of how reliable it is.

MetaMap's basic procedure of generating a set of concept candidates for a piece of text [10] can be summarized in the following steps:

[4] https://sites.google.com/site/shareclefehealth/

1. Parsing: text is parsed for identifying noun phrases.

2. Variants generation: variants of each noun phrase are generated where a variant contains at least one of the noun phrase words.

3. Candidates generation: a candidate set is formed by including all UMLS Metathesaurus strings that contain one of the variants.

4. Candidates scoring: each candidate is given a confidence score (a measure of the quality of match between a phrase and a UMLS Metathesaurus string) based on four metrics: centrality, variation, coverage, and cohesiveness.

5. Candidates merging: candidates from the disjoint parts of a noun phrase are combined and re-evaluated for the confidence score.

The final candidates (i.e., concepts) are represented by the Concept Unique Identifier (CUI) in UMLS Metathesaurus as shown in Table 2.1.

Table 2.1: CUI candidates for concept "hearing loss".

Score	CUI	Description
1000	C0011053	hearing loss (Deafness) [Disease or Syndrome]
1000	C0018772	hearing loss (Hearing Loss, Partial) [Finding]
1000	C1384666	Hearing Loss (hearing impairment) [Finding]
861	C0018767	Hearing [Physiologic Function]
861	C1455844	hearing (Hearing examination finding) [Finding]
861	C1517945	Loss [Quantitative Concept]

The concept mapping in MetaMap is imperfect. Denny et al. [32] reported a precision of 85% and a recall of 78% for MetaMap. Pratt and Yetisgen-Yildiz [89] reported similar findings: 84.5% ("weak precision") and 70.2% ("weak recall") when comparing MetaMap with human annotators.

Chapter 3

EMR SEARCH - EVIDENCE AGGREGATION

In the cohort identification, due to the special document structure of EMR and the one-to-many relationship between patients and their EMR, the evidence that contributes to patient relevance can scatter in different fields of the same document (e.g., in both meta-data fields and the clinical narrative field), distribute across multiple documents (e.g. across reports from different hospital departments), and even spread over a long time span. Therefore, how to merge and score the distributed evidence becomes critical for improving the retrieval performance. In this chapter, we introduce and evaluate several effective evidence aggregation methods that collect, weight, and combine multi-level evidence in EMR for improving patient ranking.

3.1 Data and Task

We first briefly describe the dataset used for our study. We use the official test collection from the TREC 2011 & 2012 Medical Records Track [111]. The test collection contains 100,866 de-identified medical reports from the University of Pittsburgh NLP Repository. These medical reports were gathered from multiple hospitals in the course of one month. The retrieval task[1] is an ad hoc search task for patient visits. A patient visit to the hospital normally links to multiple medical reports generated from different departments, meaning there is a 1-to-n relationship between visits and reports as shown in Figure 3.1. Based on the report-to-visit mapping information provided by TREC, we have 17,198 unique visits associated with 100,866 reports.

Each medical report is an XML file with a fixed set of fields[2] as shown in

[1] http://www-nlpir.nist.gov/projects/trecmed/2011/tm2011.html

[2] http://www.dbmi.pitt.edu/nlp/report-repository

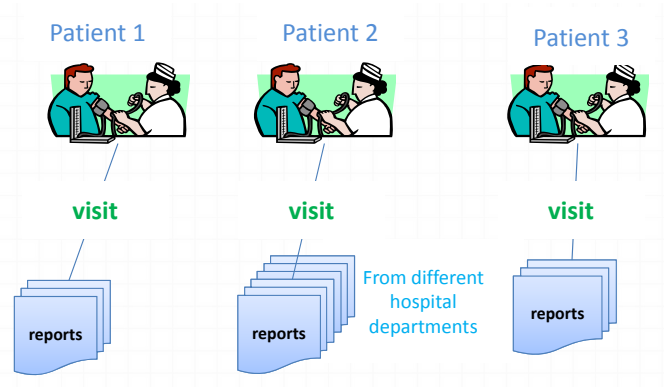


Figure 3.1: Patient visits to reports mapping. The number of reports associated with each visit can vary from one to hundreds.

```
=====================================================================
<report>
<checksum>20070209RAD-woewoefn-838-1342343139</checksum>
<subytpe>CHEST</subtype>
<type>RAD</type>
<chief_complaint>FELL OUT OF WHEELCHAIR</chief_complaint>
<admit_diagnosis>924.01</admit_diagnosis>
<discharge_diagnosis>924.01,E884.3,294.8</discharge_diagnosis>
...
<report_text>
This is a **AGE[65]-year-old male patient. ...EXAMINATION PERFORMED: ...
ONLY **DATAE[Feb 02 08] 300 HOURS CLINICAL HISTORY: Fall. FINDINGS:
Frontal views of the pelvis and specific oblique views of the right
hip show no fractures of dislocations. There is a protrusio acetabulum
on the left with chronic deformity of the acetabular margin and adjacent
femoral head with joint space narrowing. Vessel calcification is evident.
...
</report_text>
</report>
=====================================================================
```

Figure 3.2: Sample medical report. Meta-data fields contain information about the type, subtype, ICD-9 codes, etc. The main body contains clinical narratives.

Figure 3.2. The most important information resides in two diagnosis fields consisting of ICD-9 (International Classification of Diseases, 9th Revision) codes, and one free-text field containing doctors' notes. Our search system will rely on evidence within

these main fields to satisfy search users' information needs.

TREC assessors developed 85 information needs (or "topics" in TREC terminology). These needs were designed to require information mainly from the free-text fields, i.e., topics are not answerable solely by the diagnostic codes. Topics are meant to reflect the types of queries that might be used to identify cohorts for comparative effectiveness research [111]. Table 3.1 shows several TREC topics as examples. The topic usually specifies the patient's gender, age group, condition, disease, treatment, etc. Relevance judgments for the topics were also developed by TREC assessors based on the pooled results from TREC participants. It turned out that 4 out of 85 topics had too few known relevant visits and thus were dropped by TREC organizers. Therefore, we will use 81 topics with their relevance judgments for our experiments.

Table 3.1: Example topics of Medical Records Track.

ID	Topic
107	Patients with ductal carcinoma in situ (DCIS)
118	Adults who received a coronary stent during an admission
109	Women with osteopenia
112	Female patients with breast cancer with mastectomies during admission

In summary, the retrieval task is to find patients matching certain inclusion criteria for clinical studies based on a set of medical reports.

3.2 Multi-level Evidence Aggregation

In this section, we will explore evidence at different document levels and introduce several evidence aggregation methods.

Baseline Retrieval Model

We will build and evaluate the evidence aggregation methods on top of a baseline retrieval model, the query likelihood (QL) language model, which has already been described in Chap 2.1.1. For convenience , we formulate it here again:

$$\text{score}(D,Q) = \log P(Q|D) = \sum_{i=1}^{n} \log \frac{\text{tf}_{q_i,D} + \mu \frac{\text{tf}_{q_i,C}}{|C|}}{|D| + \mu}, \tag{3.1}$$

where q_i is the ith term in query Q, n is the total number of terms in Q, $|D|$ and $|C|$ are the document and collection lengths in words respectively, $\text{tf}_{q_i,D}$ and $\text{tf}_{q_i,C}$ are the document and collection term frequencies of q_i respectively, and μ is the Dirichlet smoothing parameter. The reason for selecting QL model as our baseline is that it has been shown to be a strong baseline compared with other TREC systems [127].

3.2.1 Field Level Evidence

As aforementioned, the main parts of a report are the doctor's notes and the diagnosis codes. Here we describe how we leverage ICD-9 codes in the retrieval model, and how we remove some extraneous information from doctor's notes.

Code Expansion: The "admit diagnosis" and "discharge diagnosis" fields contain ICD-9 codes which, though mainly used for billing purposes, give a high level summary of medical report content, and whose associated descriptions can provide potentially useful terms for retrieval purpose. Thus, we expand ICD codes with their corresponding descriptions[3]. For instance, we substitute code "428.1" with "LEFT HEART FAILURE". Then, if the query is "heart failure", we will find a match in the document after this substitution. We refer to this feature as ICD.

Negation Removal: The "report text" field contains clinical narratives. One distinct feature of clinical narratives is that negation phrases are frequently used to claim the absence of certain conditions or symptoms [22], such as "cannot tell", "not clear", "without evidence", etc. Negations may cause retrieval false positives. For instance, a simple IR system will consider a document with the sentence "The patient comes in with episodes of orthopnea and has ruled out for an acute coronary syndrome." as relevant to the query "acute coronary syndrome". Thus, we use NegEx[4] [42], an open-source clinical negation detection tool, to remove all negated portions of the sentences from the medical records before indexing. For instance, in the above example

[3] https://drchrono.com/public_billing_code_search

[4] http://code.google.com/p/negex/

we will delete the phrase "ruled out for an acute coronary syndrome" from the original report. We refer to this feature as NEG.

Age/Gender Filtering: We use simple regular expressions to search for age/gender indication words and phrases in both the"report text" field and the topics. We use the extracted age and gender information to filter from the retrieval set visits that do not meet the inclusion criteria specified in the topics. We refer to this feature as AGF.

3.2.2 Report Level Evidence

Evidence in a visit may mainly exist in only a small proportion of all the associated reports. This allows us to rely on the strongest evidence of a visit to estimate its relevance. Thus, we use reports as the initial retrieval units (i.e., building an index for reports and applying the retrieval model to each report), and then transform a report ranking into a visit ranking based on the strongest report-level evidence, which is equivalent to using the following report score merging function for ranking visits:

$$\text{score}_{\text{RbM}}(V, Q) = f_{\text{RbM}}(\{\text{score}(r_1^V, Q), \text{score}(r_2^V, Q), ...\}), \tag{3.2}$$

where r_j^V is a report associated with visit V based on the report-to-visit mapping, $\text{score}(r_j^V, Q)$ is the language modeling score of the report with respect to query Q, and f_{RbM} is the function for aggregating the scores. We will experiment with MAX, SUM, and ANZ (averaging over non-zeros) for f_{RbM} in Section 3.2.5.3. We name this evidence aggregation strategy Retrieval-before-Merging (RbM). The merging process involved in RbM corresponds to "merging I" in Figure 3.3.

3.2.3 Visit Level Evidence

Evidence may also spread near evenly across multiple reports, especially when the information need is a complex one. Thus, our second strategy for aggregate evidence is to first merge reports from a single visit field by field into a visit document and then construct an index for visit documents, i.e., using visits as the retrieval units. With this strategy, the language model built on a merged document can naturally combine

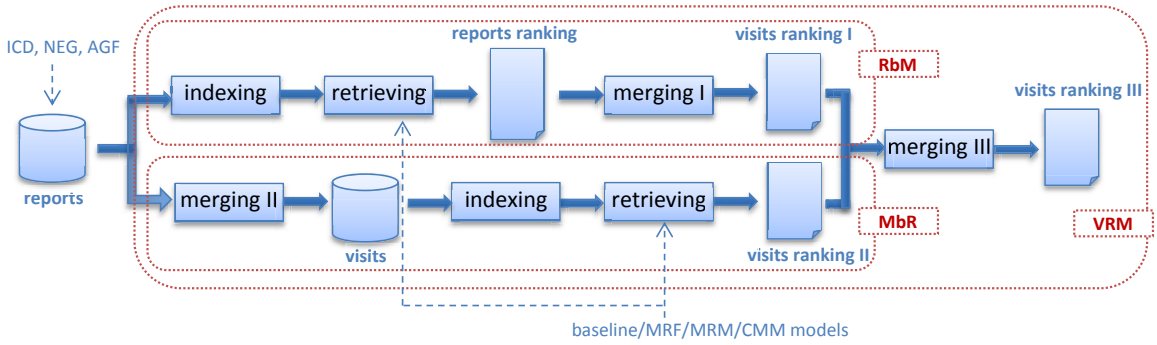


Figure 3.3: Overview of merging multi-level evidence: ICD, NEG, and AFG at the field level, RbM at the report level, MbR at the visit level, and VRM at the top level.

the evidence scattered across multiple reports. Furthermore, this strategy can directly lead to a ranking of visits which are the desired retrieval units. We call this second evidence aggregation strategy Merging-before-Retrieval (MbR). The merging process involved in MbR corresponds to "merging II" in Figure 3.3.

3.2.4 Top Level Evidence

RbM and MbR described above are two different strategies for aggregating evidence and ranking visits. RbM and MbR complement each other because: the former can leverage the strongest evidence (which may be diluted in MbR) to estimate relevance while the latter can naturally aggregate evidence spreading across multiple reports which would be challenging to do at the report-level. This complementing characteristic leads to our third evidence aggregation method in which we take advantage of both RbM and MbR by merging their visit rankings, as demonstrated by "merging III" in Figure 3.3. We call the third strategy Visit-Ranking-Merging (VRM), which is formally defined by:

$$\text{score}_{\text{VRM}}(V, Q) = f_{\text{VRM}}(\text{score}_{\text{RbM}}(V, Q), \text{score}_{\text{MbR}}(V, Q)), \tag{3.3}$$

where $\text{score}_{\text{RbM}}(V)$ and $\text{score}_{\text{MbR}}(V)$ are the language modeling scores for visit V with respect to query Q in the two visit rankings obtained by RbM and MbR respectively, f_{VRM} is the function for score aggregation, and $\text{score}_{\text{VRM}}(V, Q)$ is the final score of visit

V in the merged ranking. We will try different methods for f_{VRM} such as CombMNZ, CombSUM, and CombMAX in Section 3.2.5.4 below.

3.2.5 Evaluation

In this section, we will evaluate each of the evidence aggregation strategies described in the previous sections.

3.2.5.1 Experimental Setup

We use the Indri[5] [104] retrieval system for indexing and retrieving. In particular, we use the Porter stemmer to stem words in both reports and queries, and use a simple standard medical stoplist [44] for stopping words in queries only. Then we conduct 9-fold cross-validation and use top 1000 retrieved visits[6] for each query to evaluate our system under different evidence aggregation settings on top of the QL model.

Particularly for RbM we initially retrieve top 20000 reports to make sure that we have 1000 visits after merging. The topics were initially ordered randomly by NIST and assigned unique ID's (101-185). Thus, we split up the 81 topics into 9 folds in the order of their original ID's, i.e., we put topics 101-109 into the first fold, 110-108 to the second fold, and so on.

In each iteration of the 9-fold cross validation, we train our system on 72 queries to obtain the best setting of the Dirichlet smoothing parameter μ with respect to MAP by sweeping over the parameter space [1000, 20000] with a step size of 2000, and then generate a ranking for each of the remaining 9 queries based on the trained system. When complete, we have full rankings for all 81 topics as a test set. We evaluate the system based on the average evaluation scores over all 81 topics.

---

[5] http://www.lemurproject.org/indri/

[6] The guideline of TREC medical records track requires each retrieval set contain no more than 1000 visits.

We train our systems on MAP. This is because: 1) training on MAP is most commonly used in IR to improve retrieval performance; 2) we find that training on MAP improves the retrieval performance on other evaluation metrics as well while training on other evaluation measures does not improve the overall performance. Thus, MAP will be the primary evaluation measure in this work. In fact, MAP correlates well with other evaluation measures as we will show in the next section.

To access the statistical significance of differences in the performance of two systems, we perform one-tailed paired t-test for MAP (since we train systems on MAP).

3.2.5.2 Impact of Field Level Evidence

In Section 3.2.1, we introduced three field level features, namely ICD, NEG, and AGF. Now we evaluate their impact on the retrieval performance by varying system settings. For each one of the three evidence aggregation methods (i.e., RbM, MbR, and VRM), we start with the raw medical corpus and the QL model, and define this setting as a baseline. Then we add field level features on top of the baseline.

Table 3.2 shows that both ICD and NEG significantly improve the baseline MAP score. However, we observe very little gain from AGF. This is because our test collection contains only a few topics that have age and gender restrictions. Nevertheless, each field level feature presents consistent improvement across different evidence aggregation levels. Furthermore, combining three medical features makes a pronounced, positive impact on the retrieval performance. Since all three field level features are effective, our systems will use them by default in the rest of this thesis unless otherwise specified.

3.2.5.3 Score Merging for RbM

As mentioned in Section 3.2.2, we have several options for choosing the score merging function f_{RbM} in Equation 3.2 (i.e., "merging I" in Figure 3.3) for RbM. Now

Table 3.2: Impact of field level features. Scores shown below are all MAP scores, and they are based on 9-fold cross validation on the 81 topics from 2011 & 2012 Medical Records Track. $\triangle$ indicates statistically significant difference ($p < 0.05$) from the baseline MAP score in the corresponding row. "FIELD" is the combination of ICD, NEG, and AGF features. We will discuss the settings for RbM and VRM (i.e., the MAX and CombWEG merging algorithms) in Sections 3.2.5.3 and 3.2.5.4 respectively.

System Setting	BL (baseline)	BL+AGF	BL+NEG	BL+ICD	BL+FIELD
RbM (MAX)	0.327	0.334	$0.339^{\triangle}$	$0.342^{\triangle}$	$0.355^{\triangle}$
MbR	0.341	0.347	$0.359^{\triangle}$	$0.363^{\triangle}$	$0.382^{\triangle}$
VRM (CombWEG)	0.352	0.356	$0.373^{\triangle}$	$0.376^{\triangle}$	$0.395^{\triangle}$

we describe them formally below:

$$\text{MAX:} \quad \text{score}_{\text{RbM}}(V, Q) = \max(\{\text{score}(r_j^V, Q)\})$$

$$\text{SUM:} \quad \text{score}_{\text{RbM}}(V, Q) = \sum_j \text{score}(r_j^V, Q)$$

$$\text{ANZ:} \quad \text{score}_{\text{RbM}}(V, Q) = \frac{\sum_j \text{score}(r_j^V, Q)}{|\{\text{score}(r_j^V, Q) \neq 0\}|}$$

where again $\text{score}(r_j^V, Q)$ is the language modeling score of the report r_j^V (associated with visit V) with respect to query Q. ANZ stands for "Averaging over Non-Zeros", meaning we only consider reports containing at least one query term. MAX, SUM, and ANZ are similar to CombMAX, CombSUM, and CombANZ proposed by Fox and Shaw [99]. The difference is that CombMAX, CombSUM and CombANZ are used for merging multiple retrieval runs.

Table 3.3: Score merging for RbM. $\triangle$ indicates statistically significant difference ($p < 0.05$) from the other MAP scores. The scores are based on 5-fold cross validation on the 34 topics from 2011 Medical Records Track.

	MAX (selected)	SUM	ANZ
MAP Score	$0.355^{\triangle}$	0.293	0.3077

Table 3.3 shows that MAX significantly outperforms SUM and ANZ. This confirms our assumption that we can rely on the strongest evidence (i.e, the most relevant

report) of a visit to estimate the relevance of that visit. Thus, we will use MAX as the default setting for score merging in RbM in the rest of this thesis.

3.2.5.4 Score Merging for VRM

Similarly, we also have several options for choosing the score merging function f_{VRM} in Equation 3.3 (i.e., "merging III" in Figure 3.3) for VRM, such as CombMNZ, CombANZ [99]. In our case, we are only merging two rankings. Thus, the merging methods are specified as follows:

$$\text{CombMNZ:} \quad \text{score}_{\text{VRM}}(V,Q) = N_V \cdot [\text{score}_{\text{RbM}}(V,Q) + \text{score}_{\text{MbR}}(V,Q)]$$

$$\text{CombSUM:} \quad \text{score}_{\text{VRM}}(V,Q) = \text{score}_{\text{RbM}}(V,Q) + \text{score}_{\text{MbR}}(V,Q)$$

$$\text{CombANZ:} \quad \text{score}_{\text{VRM}}(V,Q) = \frac{\text{score}_{\text{RbM}}(V,Q) + \text{score}_{\text{MbR}}(V,Q)}{N_V}$$

$$\text{CombWEG:} \quad \text{score}_{\text{VRM}}(V,Q) = \lambda_{\text{vrm}} \cdot \text{score}_{\text{RbM}}(V,Q) + (1-\lambda_{\text{vrm}}) \cdot \text{score}_{\text{MbR}}(V,Q)$$

where $\text{score}_{\text{VRM}}(V,Q)$ is the merged score for visit V, and $\text{score}_{\text{RbM}}(V,Q)$ and $\text{score}_{\text{MbR}}(V,Q)$ are the scores for V in two different visit rankings as demonstrated in Figure 3.3, and N_V is the number of rankings that have V in the top 1000 retrieved visits. Note that $\text{score}_{\text{MbR/RbM}}(V,Q) = 0$ if V does not appear in the top 1000 retrieved. CombSUM is a special case of CombWEG. We train λ_{vrm} using cross validation within the range of (0.1, 1.0) with a step size of 0.1.

We compare the performance of these merging methods using MAP and P10 in Table 3.4. As we can see, CombWEG, CombMNZ and CombSUM achieve comparable performance, and are better than CombMAX and CombANZ. Thus, we can infer that a good aggregation strategy for "merge III" should favor visits that appear in both rankings. We use CombWEG as the default merging method for VRM for the rest of this chapter.

Table 3.4: Score merging for VRM. CombWEG, CombMNZ, and CombSUM achieve comparable performance, and are better than CombMAX and CombANZ, which infers that a good merging strategy for VRM should favor visits that appear in both rankings. The scores are based on 5-fold cross validation on the 35 topics from 2011 Medical Records Track.

Method	**MAP**	**P10**
CombWEG (selected)	0.395	0.582
CombSUM	0.393 (different from CombWEG at level $p < 0.10$)	0.572
CombMNZ	0.393 (different from CombWEG at level $p < 0.10$)	0.572
CombMAX	0.368 (different from CombWEG at level $p < 0.05$)	0.547
CombANZ	0.367 (different from CombWEG at level $p < 0.05$)	0.556

3.2.5.5 Performance Comparison

Now we compare the performance of RbM, MbR, and VRM since each one of them can produce a visit ranking as shown in Figure 3.3. Table 3.5 shows that VRM is significantly better than MbR and RbM on MAP, which means that merging visit rankings as the top-level evidence aggregation strategy boosts the retrieval performance significantly. This confirms our assumption in Section 3.2.4 that RbM and MbR complement each other and their combination brings further improvement.

Table 3.5: Evidence Aggregation Methods. The scores are based on 5-fold cross validation on the 35 topics from 2011 Medical Records Track. $\triangle$ indicates statistically significant difference ($p < 0.05$) from the other MAP scores. RbM and MbR complement each other and their combination brings further improvement.

System	**MAP**	**bpref**	**P10**	**Rprec**
RbM	0.355	0.436	0.546	0.386
MbR	0.382	0.463	0.563	0.400
VRM	$0.395^{\triangle}$	0.469	0.582	0.413

3.3 Adaptive Evidence Aggregation

Intuitively, different queries can have different forms of evidence distribution. For some queries the evidence may concentrate in only a few associated reports while for others the evidence may spread near uniformly across many reports. As we have seen, RbM focuses on dealing with the former situation using the local information

each individual report while MbR mainly handles the latter using global information that comes from the combined reports. It would be beneficial to find a balance between RbM and MbR with respect to different queries, i.e., the function f_{VRM} in Equation 3.3 needs to be adaptive to the query.

3.3.1 Query-adaptive Scoring

Therefore, we propose a new query-adaptive scoring function for f_{VRM} as shown below:

$$\text{score}_{\text{VRM-Adaptive}}(V, Q) = \alpha_Q \cdot \text{score}_{\text{R}}(V, Q) + (1 - \alpha_Q) \cdot \text{score}_{\text{V}}(V, Q), \tag{3.4}$$

where $\text{score}_{\text{R}}(V, Q)$ and $\text{score}_{\text{V}}(V, Q)$ are the relevance scores of document V from report-based and visit-based retrievals respectively, and α_Q is the query-adaptive coefficient for scoring merging. If we can adjust α_Q appropriately, Equation 3.4 should be able to deal with the two extreme evidence distribution cases mentioned above and others between those two cases. This idea is also illustrated visually in Figure 3.4.

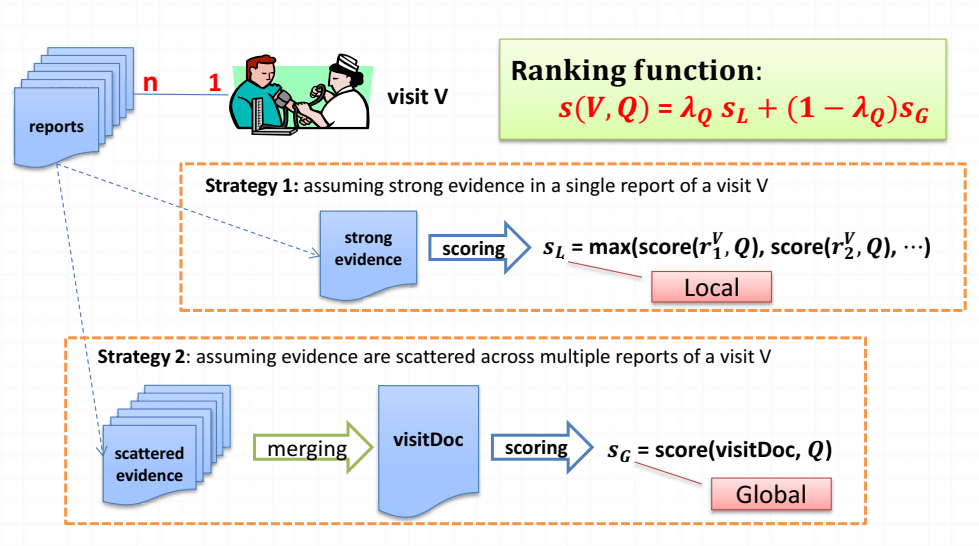


Figure 3.4: Adaptive evidence aggregation. RbM uses local information for concentrated evidence while MbR uses global information to deal with scattered evidence. α_Q is the query-adaptive coefficient for score merging.

3.3.2 Learning Algorithm

We propose to adaptively set α_Q with respect to different queries by learning the weight α_Q based on a set of features.

In particular, we can view α_Q as a mixing probability: the probability that the evidence clusters in only one report rather than spreads across multiple reports. Then, assuming the log-odds of that probability can be expressed as a linear combination of feature values, we may write:

$$\log \frac{\alpha_Q}{1-\alpha_Q} = \beta_0 + \sum_{i=1}^{m} \beta_i x_i + \epsilon_Q \tag{3.5}$$

where β_0 is a model intercept (or bias term), x_i is the value of feature number i, β_i is the weight coefficient of that feature, and ϵ_Q is a slack variable.

This is essentially a logistic regression model[7]. Logistic regression is fit using iteratively reweighted least squares to find the values of the β coefficients that are the best fit to training data. Given feature values and their β coefficients, we can then predict the mixing probability α_Q for new queries.

3.3.3 Features

We propose 14 features that are possibly related to the evidence distribution in visits, and can be used to predict the weight α_Q in Equation 3.4. All these features are based on characteristics of the medical concepts contained in the query. We detect these medical concepts using MetaMap [9], a medical NLP tool developed at the National Library of Medicine (NLM) to map biomedical text to concepts in the Unified Medical Language System (UMLS) Metathesaurus. The concepts are represented by the Concept Unique Identifier (CUI) in UMLS Metathesaurus as already shown in Section 2.2.5. Thus, we use Q_C to represent a concept query that is converted from the original text query Q and contains only CUIs. Next, we describe these 14 features in detail:

[7] While logistic regression is often used for 0/1 classification problems, it can also be used when the target variable is a real number between 0 and 1. In this case it is sometimes called a "quasibinomial" model.

1. Length of the query

Intuitively, evidence is more likely to resides across reports for long queries. Thus, we use the length of query $|Q|$ as the feature to estimate the evidence distribution. It is defined formally as $|Q| = \sum_{w \in Q} \text{cnt}(w, Q)$, where $c(w, Q)$ is the count of term w in Q.

2. Number of concepts in the query

Similarly, if a query contains more medical concepts, it is more likely that the evidence distributes across multiple reports. We define this feature formally as $|Q_C| = \sum_{w_c \in Q_C} \text{cnt}(w_c, Q_C)$, where $\text{cnt}(w_c, Q_C)$ is the count of concept term w_c in Q_C. $|Q_C|$ is a better feature than $|Q|$ because if the query contains a medical concept whose name is very long then $|Q|$ might not be a good indicator of the evidence distribution.

3. Broad/narrow query concepts

A text query can contain several medical concepts, for each of which the MetaMap program will return 1 to 10 candidates. We hypothesize that a concept with more candidates is less specific, and thus is more likely to be a broad or ambiguous concept and tends to appear in multiple reports. Thus, the average number of returned MetaMap candidates for concepts in a query may be a good indicator of evidence distribution. We define this feature as $R_C = \frac{\sum_{w_c \in Q_C} |\text{Meta}(w_c)|}{|Q_C|}$, where $|Q_C|$ is the original concept query length (i.e., the length before expansion), $|\text{Meta}(w_c)|$ is the number of concept candidates returned by MetaMap for concept term w_c in concept query Q_C.

4. Semantic similarity among query concepts

Intuitively, if Q_C contains concepts that are semantically close to each other, the associated evidence tends to co-occur in the same report. However, if the concepts are semantically distant, the corresponding evidence may tend to distribute across reports. Therefore, we use the semantic distance among query concepts to estimate how the evidence distributes.

In particular, we use YTEX[8] to measure semantic similarity. Given a pair of

[8] http://code.google.com/p/ytex/wiki/SemanticSim_V06

UMLS concepts, YTEX can produce knowledge based and distributional based similarity measures. The former uses knowledge sources such as dictionaries, taxonomies, and semantic networks, while the latter mainly uses the distribution of concepts within some domain-specific corpus [36].

We use the 11 measures listed in Table 3.6 as our features. Garla and Brandt provide a detailed overview [36] of these semantic similarity measures.

Table 3.6: Semantic similarity measures for medical concepts in UMLS.

Type	Method	Notation	Name
Knowledge-based	Path-Finding	WUPALMER	Wu & Palmer
		LCH	Leacock & Chodorow
		PATH	Path
		RADA	Rada
	Intrinsic IC based	IC_LIN	Lin
		IC_LCH	Leacock & Chodorow
		IC_PATH	Jiang & Conrath
		IC_RADA	Rada
		JACCARD	Jaccard
		SOKAL	Sokal & Sneath
Distributional-based	Corpus IC based	CIC_LIN	Lin

For each query and each specific measure, we take the mean of the semantic similarity scores from all possible UMLS concept pairs in the query as one feature.

3.3.4 Evaluation

3.3.4.1 Experimental Setup

We use the Indri retrieval system for indexing and retrieving. In particular, we use the Porter stemmer to stem words in both text documents and queries, and use a standard medical stoplist [44] for stopping words in queries only.

To make it more interesting and challenging, we build this adaptive VRM method on top of an advanced model called CME model (which will be described in Section 4.1) as a stronger baseline than the QL model. The collections used for query expansion in CME are the ClueWeb09 Category B corpus, the 2009 Genomics

Track corpus, 2012 Medical Subject Headings (MeSH), and the medical records corpus itself. We will describe these collections in detail in Section 4.2.2. Both the report and visit-based retrievals (i.e., RbM and MbR) will use this setting.

Because the focus of Section 3.3 is to evaluate the adaptive scoring Function 3.4, we set CME model parameters to some default values, and we use the same set of parameter values for both the report and visit-based retrievals. In particular, we set the Dirichlet smoothing parameter μ to 2500. For the MRF model in CME, we set the feature weights $(\lambda_T, \lambda_O, \lambda_U)$ to (0.8, 0.1, 0.1). For the EMRM model in CME, we take take the top-weighted 10 terms from the top-ranked 50 documents for each expansion collection. Again, the details about these model parameter will be described in Section 4.1 when we introduce the advanced models for tackling the medical language related problems in EMR search. For now the reader should be fine without a full understanding of CME.

To evaluate our learning algorithm as described in Section 3.3.2, we first obtain the optimal coefficient $\alpha_{Q\text{-opt}}$ for each topic Q by sweeping [0, 1] (i.e., the valid range of α_Q) at a step size of 0.1. Then we conduct leave-one-out cross-validation (LOOCV), in each iteration of which the system predicts the coefficient α_Q for one new topic based on $\alpha_{Q\text{-opt}}$'s for the other 80 topics. With limited topics available for learning a relatively complex prediction model, using LOOCV can maximize the size of training data we can use in each iteration of the cross-validation, and lead to a better estimate for each feature weight.

Similar to the setup in Section 3.2.5.1, we train our systems on MAP and perform one-tailed paired t-test for MAP to compare the performance of two systems. We report scores for MAP, R-precision (Rprec), bpref, and precision at rank 10 (P10).

3.3.4.2 Feature Selection

To choose a good subset of the 14 features, we take a greedy approach in which we start with a full set of features and iteratively eliminate exactly one feature at a time that has the greatest negative impact on the retrieval performance until when

further removing any feature will degrade the performance. The algorithm for this greedy feature elimination is formally described below:

Require:

(1) $n = 14$; $\mathcal{F} \leftarrow \cup_{i=1}^{n} f_i$ (f_i is the ith feature)

(2) compute score MAP_n using $\mathcal{F}$ in LOOCV

while $|\mathcal{F}| > 1$ **do**
  $k \leftarrow 0$
  for $f_i \in \mathcal{F}$ **do**
    $\mathcal{F}' \leftarrow \mathcal{F} - f_i$; compute MAP using $\mathcal{F}'$ in LOOCV
    if $\text{MAP} \geq \text{MAP}_n$ **then**
      $k = i$; $\text{MAP}_{n-1} \leftarrow \text{MAP}$
    end if
    if $k = 0$ **then**
      return $\mathcal{F}$
    end if
    $n \leftarrow n - 1$
  end for
end while
return $\mathcal{F}$

After the above feature set pruning step, there are 8 features left as shown in Table 3.7. We further study the importance of each feature by analyzing the prediction model trained in a randomly selected iteration of LOOCV using these 8 features. Based on the statistical significance of each feature as shown in Table 3.7, we can infer that:

1) All the intrinsic IC based features except IC_LCH are in the pruned feature set, indicating that this type of similarity measures is effective for predicting α_Q. In fact, the intrinsic IC similarity measure incorporates taxonomical evidence explicitly modeled in ontologies (such as the number of leaves/hyponyms and subsumers), which

Table 3.7: Features in the pruned set using LOOCV, sorted by their statistical significance scores.

Feature	Significance	Feature	Significance
IC_RADA	0.0112	R_C	0.0654
WUPALMER	0.0299	SOKAL	0.0671
RADA	0.0368	IC_LIN	0.0824
JACCARD	0.0647	IC_PATH	0.0876

are not captured by the path-finding based measure. Furthermore, the intrinsic IC similarity measure does not depend on the availability of domain corpora, thus is considered more scalable and easily applicable than the distributional-based measure [95].

2) R_C is an informative feature though it only uses similarity information about each query concept with its neighbors (rather than with other query concepts) in the semantic network.

3) Neither $|Q|$ nor $|Q_C|$ is in the pruned set, suggesting that non-semantic-similarity-based features are generally not useful for estimating the evidence distribution.

4) RADA is a feature that might worth further exploration because both the Path-finding based and the intrinsic IC based RADA features are in the pruned set.

In summary, the most important characteristic of a good feature for predicting α_Q is that the feature should capture the ambiguity or broadness of each individual concept term in Q_c or it should capture the relationship (i.e., semantic distance) among query concepts.

3.3.4.3 Adaptive Weighting

Fixed Weighting

We first evaluate the performance of Equation 3.4 when α is fixed (i.e., not adaptive). In each iteration of the LOOCV, we sweep α from 0 to 1 with a step size of 0.1 to get the best value setting for α on the 80 training topics, and then use the trained α value to test the single testing topic.

We show the results in the "Fixed-weighting" row of Table 3.8. Note that this system corresponds to udelWEG [129] which is one of the top-ranked 2012 Medical Records Track systems.

Optimal Weighting

We obtain the optimal $\alpha_{Q\text{-opt}}$ for each topic separately by sweeping α from 0 to 1 with a step size of 0.1. Then, we use the $\alpha_{Q\text{-opt}}$'s to compute the best retrieval performance (i.e., an upper-bound) Equation 3.4 can possibly achieve, as shown in the 'Optimal-weighting' row of Table 3.8.

We also show the sensitivity of α by plotting the MAP score of several randomly selected topics with a varying α. Figure 3.5 shows that the retrieval performance differs significantly with respect to different settings of α, and the $\alpha_{Q\text{-opt}}$ can be very different for different topics, which confirms the necessity of making VRM query-adaptive.

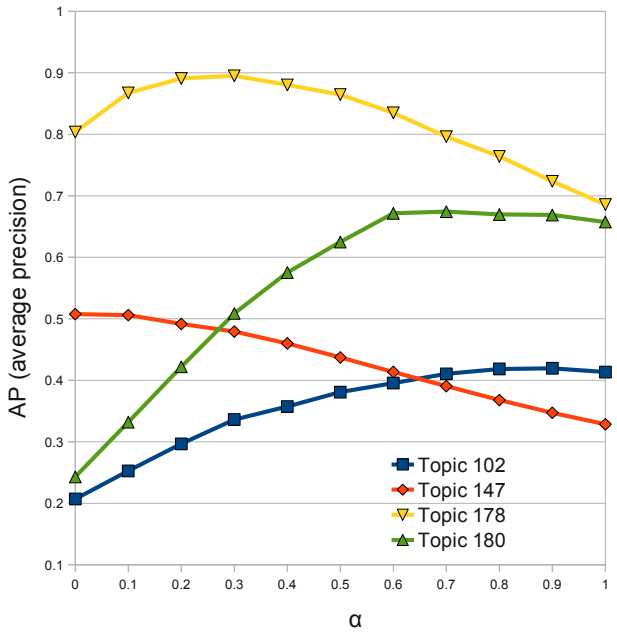


Figure 3.5: Sensitivity of retrieval performance to varying α for different topics. This indicates that making VRM query-adaptive would be beneficial.

Table 3.8: Performance comparison. A superscript on the MAP score of system X corresponds to the initial of system Y, and indicates statistical significance ($p < 0.05$) in the MAP difference between X and Y. The last column is the mean square error of the predicted weights. 'Fixed-weighting' corresponds to one of the top-ranked TREC systems as mentioned in Sections 3.3.4.1 and 3.3.4.3.

System	**MAP**	**R-prec**	**bpref**	**P10**	Pred. MSE
Visit-based	0.4122	0.422	0.499	0.619	–
Report-based	0.4354^{V}	0.435	0.511	0.607	–
Fixed-weighting	$0.4472^{V,R}$	0.443	0.520	0.631	0.128
Adaptive-weighting	$0.4485^{V,R}$	0.447	0.523	0.642	0.125
Optimal-weighting	$0.4639^{V,R,F,A}$	0.457	0.539	0.656	0.000

Performance Comparison

Table 3.8 shows performance comparison of our adaptive merging method with fixed-weighting, optimal-weighting, and two other baselines (report-based retrieval and visit-based retrieval). Our adaptive merging method is better than the fixed weighting method on all the evaluation metrics. The improvement is not statistically significant ($p = 0.191$), possibly because 81 topics may not be enough to train a good prediction model for our adaptive weighting method. In addition, the data are slightly skewed as Figure 3.6 showing that $\alpha_{Q\text{-opt}} = 1$ or 0.9 on about one third of the topics.

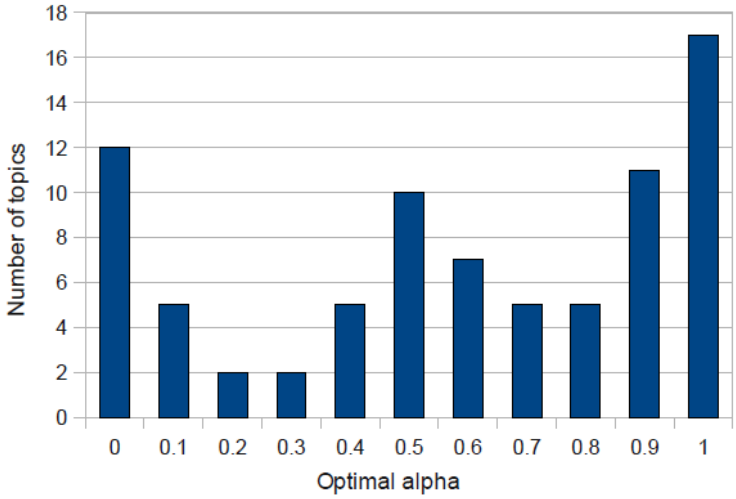


Figure 3.6: Distribution of topics against $\alpha_{Q\text{-opt}}$.

3.4 Related Work

There are a few relevant work that specifically deal with the distributed evidence in the EMR using the same TREC Medical Records Track dataset. Limsopatham et al. [64] explored using the type of medical records for enhancing retrieval performance. They demonstrated that incorporating department level evidence of the medical reports in their extended voting model and federated search model could improve the retrieval effectiveness.

More recently, they [65] proposed a new method which learns to selectively choose between a patient model and a document model based on a binary classifier built on top of features that are indicative of query difficulties. They implemented the method in Terrier [85]. Their results showed significant improvements over several strong baselines such as CombSUM and ComMAX.

Koopman [56] also discussed related issues about the the influences of document length for EMR retrieval. They concluded that the number of reports (or the length of the visit documents) does not correlate with the relevance of the visits.

3.5 Conclusion

In this chapter, we have described and evaluated a number of aggregation strategies for documents formed by EMR at different granularities. At the field level, we explored three features, namely ICD-9 code expansion (ICD), negation detection and removal (NEG), and age/gender filtering (AGF). In particular, ICD focus on improving the recall since it expand the medical record with helpful words that summarizes the medical report. On the other hand, NEG and AGF improves the recall as the former prevents negative instances from being retrieved while the latter removes false positives in the retrieval set.

At the report level, we experimented with SUM, MAX, and ANZ as the combination strategy for RbM (Retrieval-before-Merging). MAX outperforms SUM and ANZ significantly, which indicates that we can indeed rely on the strongest local evidence (i.e., the single most relevant report of each visit) to measure the relevance of the

visits. RbM works best for queries whose corresponding evidence tend to concentrate in a single report.

At the visit level, we aggregated evidence by merging reports for a single visit field by field into a large visit document, and then perform retrieval against an index of visits. We call this method MbR (Merging-before-Retrieval). MbR scores a visit based on the global statistics for that visit. MbR works best for queries whose corresponding evidence tend to distribute across multiple reports.

Finally at the top/patient level, we introduced and examined both the basic merging methods (i.e., CombSUM, CombMNZ, CombMAX, and CombANZ) and the advanced method (query-adaptive scoring) for VRM (Visit-Ranking-Merging). In particular, CombWEG, CombMNZ, and CombSUM achieve comparable performance, and are better than CombMAX and CombANZ, which infers that a good merging strategy for VRM should favor visits that appear in both rankings. That further confirms the necessity to assess the relevance of a visit from both the local (i.e. report level) and global (i.e., visit level) perspectives.

For the adaptive VRM, we studied features that are useful for predicting the evidence distribution in visits with respect to specific queries. In general, it is beneficial to include features that measure the semantic similarity of query concepts (i.e., distance among query concepts in medical ontology), such as the path-finding and intrinsic IC based similarity measures.

Chapter 4

EMR SEARCH - MEDICAL LANGUAGE

We have discussed a few medical language related issues that hurt search performance in Chapter 1. Among them, problems related to polysemy and synonym in medical text are the most common ones.

Polysemy is the capacity of a word to have different meanings under different context. For example, the word "cold" can mean the temperature, or a kind of sensation, or a disease. The abbreviation "PCP" can stand for the drug "phencyclidine", the disease "pneumocystis carinii pneumonia", or even an individual – the "primary care physician". Polysemy causes ambiguity and consequently hurts *precision* as the number of false positives increases in the search results.

On the other hand, synonymy leads to vocabulary gaps between queries and documents, and eventually brings down the recall. For example, given the query "smoker", a simple unigram-matching-based search system will not be able to retrieve documents that contain the phrase "tobacco user" but without the word "smoker" or "smokers". Furthermore, due to the richness of synonym in medical language, medical professionals often find it difficult to formulate a satisfying query for their information needs [121], which suggests that a special search engine is highly desired that can automatically expand the query with related terms to mitigate vocabulary mismatch.

Therefore, in this chapter we will introduce and evaluate several retrieval models specifically designed for alleviating the polysemy and synonymy related problems in medical IR.

4.1 Retrieval Models

We use the query likelihood (QL) language model (which we have already seen in the previous two chapters) as our baseline model. For convenience, we formulate it here again:

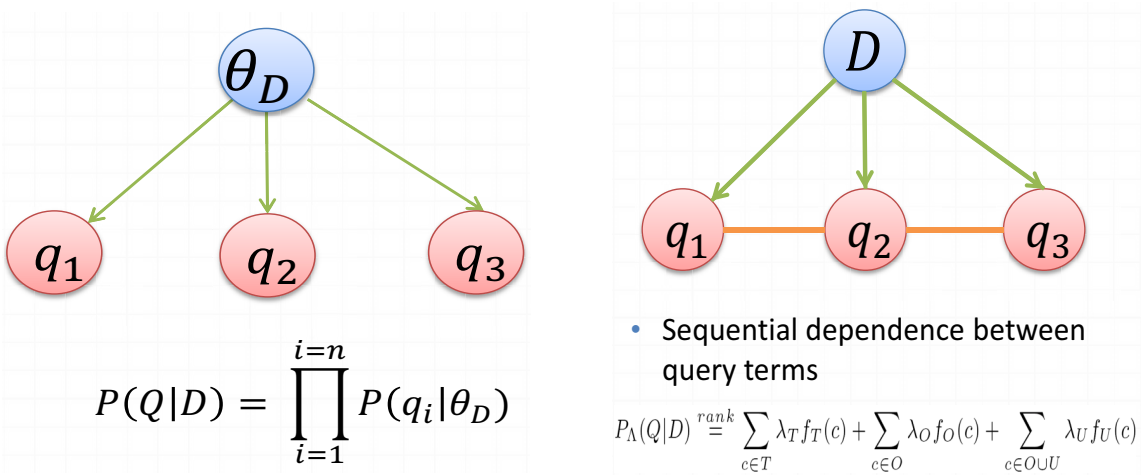
$$\text{score}(D, Q) = \log P(Q|D) = \sum_{i=1}^{n} \log \frac{\text{tf}_{q_i,D} + \mu \frac{\text{tf}_{q_i,C}}{|C|}}{|D| + \mu}, \tag{4.1}$$

where q_i is the ith term in query Q, n is the total number of terms in Q, $|D|$ and $|C|$ are the document and collection lengths in words respectively, $\text{tf}_{q_i,D}$ and $\text{tf}_{q_i,C}$ are the document and collection term frequencies of q_i respectively, and μ is the Dirichlet smoothing parameter. The reason for selecting QL model as our baseline is that it has been shown to be a strong baseline compared with other TREC systems [127]. Next, we will look at several advanced models.

4.1.1 Markov Random Field Model

To mitigate the polysemy issue, we use the Markov random field (MRF) model proposed by Metzler and Croft [79] to model term dependencies. The intuition is that medical queries usually contain phases that describe conditions, symptoms, drug names, treatments, etc. These query terms are likely to occur in close proximity to each other in the relevant documents. In addition, MRF's ability to incorporate contextual information can disambiguate word senses to some extent.

MRF has been shown to be very effective for improving web search and news search, but has not been tested in the biomedical domain. The MRF model works by first constructing a graph that contains a document node, one node per query term, and edges that represent dependencies among nodes, as shown in Figure 4.1b. Then, MRF models the joint distribution over the document random variable and query term random variables. We use their sequential dependence model in particular, which means edges exit only between adjacent query terms nodes in addition to those connecting every query term node to the document node. MRF can be viewed as an extension of the QL model as the former considers dependence between query terms



(a) Query likelihood model (b) Markov random field model

Figure 4.1: Model Comparison I. MRF can be viewed as an extension of QL by incorporating term dependence features.

and the latter simply assumes independence between query terms, as illustrated by Figure 4.1.

The ranking function of the MRF model is of the form:

$$P_\Lambda(Q|D) \overset{\text{rank}}{=} \sum_{c\in T} \lambda_T f_T(c) + \sum_{c\in O} \lambda_O f_O(c) + \sum_{c\in O\cup U} \lambda_U f_U(c), \tag{4.2}$$

where T is defined to be the set of 2-cliques containing the document node and a query term node, O is the set of cliques involving the document node and two or more query terms that appear contiguously in the query, and U is the set of cliques involving the document node and two or more query terms that appear non-contiguously within the query. $f(c)$ is the feature function over clique c. λ_T is the weight given to the original bag-of-words query, λ_O the weight given to ordered phrases, and λ_U the weight given to unordered phrases.

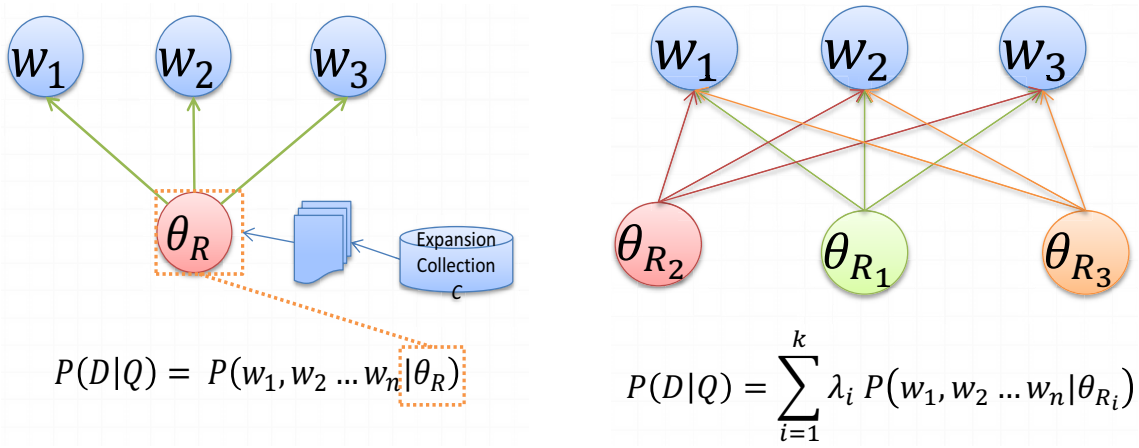
4.1.2 Mixture of Relevance Models and Its Extension

4.1.3 MRM for Query Expansion

To tackle the synonymy issue, we expand the query with additional "related" terms (also called expansion terms) that are derived from a relevance model θ_Q, which

itself is built upon top-ranked k documents from the target collection (i.e., the same collection used for retrieval) with respect to the query.

Relevance modeling can be further improved upon by leveraging information in other document collections [33, 130]. Specifically, we can form relevance models for two or more additional collections, then expand the query using those models, as illustrated in Figure 4.2. This leads to another advanced model called Mixture of Relevance Models (MRM).



(a) Relevance model

(b) Mixture of relevance models

Figure 4.2: Model Comparison II. MRM model is an extension of the relevance model.

To achieve better performance, we linearly interpolate the mixture of relevance models with the maximum likelihood (ML) query estimate by formulating the equation:

$$P(w|\theta_Q) = \lambda_Q \frac{\#(w,Q)}{|Q|} + \sum_C \lambda_C P(w|\hat{\theta}_{Q,C}), \tag{4.3}$$

where the first part is the weighted ML query estimate for word w and the second part represents the mixture of relevance models. In particular, $P(w|\hat{\theta}_{Q,C})$ is the probability of w in the estimated relevance model $\hat{\theta}$ built upon top-ranked documents in expansion collection C. λ's are collection weights and $\lambda_Q + \sum_C \lambda_C = 1$. Mixing the relevance models with the ML model is a common technique to prevent query drift by assigning more weight to the original query terms in the ML model. In fact, Equation 4.3 reduces to the well-known variant of the relevance model, the RM3 model [6], when there is only one relevance model.

We implemented MRM in Indri [104] which naturally supports such queries with the "#weight" operator; we implement Equation 4.3 in Indri by formulating a query of the following format:

#weight(
  λ_Q #combine(w_1 w_2 ... $w_{|Q|}$)
  λ_{C_1}#weight(p_{11} e_{11} p_{12} e_{12} ... p_{1m} e_{1m})
  ...
  λ_{C_n}#weight(p_{n1} e_{n1} p_{n2} e_{n2} ... p_{nm} e_{nm})
).

Here w_i represents a term in the original user query; e_{ij} represents the jth expansion term (in decreasing order of probability p_{ij}) from collection C_i. n is the number of expansion collections, and m is the number of terms to expand with. The "#combine(w_1 w_2 ... $w_{|Q|}$)" phrase corresponds to the ML query estimate while the "#weight(p_{i1} e_{i1} p_{i2} e_{i2} ... p_{im} e_{im})" phrase corresponds to the estimate of relevance model $\hat{\theta}_{Q,C_i}$. Note that p will be automatically normalized by the "#weight" operator in Indri. We will explain how to obtain expansion terms e and estimate their weights p shortly.

Thus, an expanded query based on two expansion collections when the values of λ's are specified as (0.7, 0.2, 0.1) looks like the following:

#weight(
  0.7 #combine(female breast cancer mastectomies admission)
  0.2 #weight(0.225 mastectomy 0.145 women 0.110 risk
    0.107 prophylactic 0.101 bct 0.074 radiate 0.068 therapy
    0.062 radiotherapy 0.058 surgery 0.050 adjuvant)
  0.1 #weight(0.211 mammographic 0.159 tram 0.101 dci
    0.116 mammography 0.93 flap 0.082 mammogram
    0.068 duct 0.063 biopsy 0.059 axillary 0.048 recurrence)
).

4.1.4 Extended MRM

Now we make several extensions on top of the traditional MRM: 1) we choose external expansion collections from different domains rather than from the same (i.e., clinical) domain as the target collection. As we will show in Section 4.2.2 that we test a variety of external collections, ranging from general web datasets to domain-specific datasets, and from small to large datasets. By doing so, we can derive a diversified set of query related expansion terms, and consequently reduce the vocabulary gap caused by medical synonymy; 2) we use information beyond the term frequency (tf) statistics to weight the expansion terms in each relevance model; and 3) we extends the MRM model to accommodate not only free-text collections but also medical thesauri (e.g., MeSH) by designing a special query expansion method. We elaborate on Points 2 and 3 next.

General Expansion

For general free-text collections we sort and select expansion terms by their weights p which are estimated by:

$$p_i = \sum_{j=1}^{k} \exp\{\frac{\mathrm{tf}_{e_i,D_j}}{|D_j|} + \log\frac{|C|}{\mathrm{df}_{e_i,C}} + \mathrm{score}(D_j, Q)\}, \tag{4.4}$$

where $\mathrm{score}(D_j, Q)$ is the query likelihood score for the top jth feedback document in the initial retrieval set ranked by the QL model, tf_{e_i,D_j} is the term frequency of e_i in document D_j, $\mathrm{df}_{e_i,C}$ is the document frequency of e_i in collection C, and $|D_j|$ and $|C|$ are document and collection lengths in words respectively. This formula estimates the importance of term e_i based on its term frequency, inverse document frequency, and feedback document scores. m terms with highest scores p are selected as expansion terms, and they form our estimated relevance model $\hat{\theta}_Q$. Note that we also normalize p so that we have an estimated probability $P(w|\hat{\theta}_Q)$ for each word w.

Medical Thesaurus-based Expansion

Medical thesaurus-based expansion differs from general expansion in that there are no feedback documents for obtaining expansion terms e and estimating weights

p. Thus, we extract medical concepts from the query for expansion, and propose a novel concept weighting method based on information from a query log. In this thesis, we use the MeSH ontology for demonstration, and call this method *MeSH expansion* which contains four steps:

1. Concept identification: use PubMed e-utility [2] to identify MeSH concepts in the query
2. Concept expansion: expand a detected MeSH concept rc by its entry terms and decedent nodes down level l in the MeSH trees rooted at rc, i.e., obtaining expansion terms e for the original query
3. Concept weighting: estimate weight p for each e using a PubMed query log
4. Concepts aggregation: aggregate the weights of expansion terms and form a final expansion list

In Step 2, we also model term proximity using MeSH concepts wherever applicable. For instance, for MeSH terms "Usher Syndromes" and "Hearing Loss, High-Frequency", we will formulate "#1(usher syndromes)" and "#uw16(#1(hearing loss), high-frequency)" respectively in Indri as expansion terms. The former means "usher syndromes" must occur as a phrase while the latter means "high-frequency" and "hearing loss" can occur within a text window of 16 words. Note that we avoid expanding MeSH concepts by their ancestor nodes because broader concepts are more likely to cause query drift and compromise precision. Moreover, we do not split phrase concepts into single terms because single terms are likely to be semantically different or far less discriminative than their associated phrase concepts (e.g., "usher syndromes", "back pain", "sleep walking disorder", etc.).

The PubMed query log used in Step 3 contains 2,996,301 queries submitted by 627,455 different users [47]. We estimate weight of term e_i by:

$$p_i = \frac{\log N_{e_i,G}}{\sum_j \log N_{e_j,G}}, \tag{4.5}$$

where $N_{e_i,G}$ is the number of users whose queries contain e_i in query log G. The logarithm dampens the effect of large differences in counts. Equation 4.5 estimates the

popularity of e_i and its variants among users who use them interchangeably to express a medical concept in general. For instance, "hearing impairment" is more common than "hypoacusis" for expressing the concept "hearing loss" and consequently gets a larger weight.

Finally, we refer to our new model described in this section as Extended Mixture of Relevance Models, or EMRM for short.

4.1.5 A Hybrid Model

MRF improves precision by using contextual information while EMRM enhances recall by expanding the query with related terms. Therefore, we linearly combine MRF and EMRM to get a hybrid model called CME (Combined MRF and EMRM model) which is expected to benefit from the complementing advantages of MRF and EMRM. The scoring function of CME is formally defined as:

$$P(w|\theta_Q) = \lambda_Q \cdot \text{MRF} + \sum_C \lambda_C P(w|\hat{\theta}_{Q,C}), \tag{4.6}$$

which is structurally similar to Equation 4.3.

4.2 Evaluation

4.2.1 Experimental Setup

We use the Indri retrieval system for indexing and retrieving. In particular, we use the Porter stemmer to stem words in both reports and queries, and use a simple standard medical stoplist [44] for stopping words in queries only. Then we perform similar cross-validation as described in Section 3.3.4.1. In particular, to train the EMRM and CME models by sweeping the parameter space according to Table 4.1.

Table 4.1: Parameter space for training EMRM and CME Models.

Parameter	Explanation	From	To	Step Size
μ	Dirichlet smoothing parameter	1000	20000	2000
k	number of top-ranked expansion documents	20	80	30
m	number of expansion terms	10	21	5

The baseline system using the QL model has only one free variable μ to train. We fix μ to 10000 for other systems to reduce the training time. For systems using single expansion collections, we train them to obtain λ_Q, k, and m, except for MeSH expansion which only needs to train λ_Q because, unlike general expansion, low-ranked MeSH expansion candidate terms can still be highly related to the original query terms. For systems using multiple expansion collections, we fix k to 50, m to 10 for efficiency, and thus λ_Q will be the only free variable for training.

We train our systems on MAP. To access the statistical significance of differences in the performance of two systems, we perform one-tailed paired t-test for MAP.

4.2.2 Selection of Expansion Collections

4.2.2.1 MeSH Expansion

We first evaluate the MeSH expansion for EMRM. We compare the effectiveness of different settings for MeSH expansion as listed in Table 4.2: 1) Entry: using entry terms only and without term weighting (i.e., no Step 3 described in Section 4.2; 2) Tree1: using tree terms only, tree expansion level $l = 1$, and no weighting; 3) EntryTree1: using both entry and tree terms with $l = 1$, no weighting; and 4) WEntryTree-1: weighted EntryTree1 using PubMed query log; 5) WEntryTree[2-6]: similar to WEntryTree-1 but using different values (i.e., $2 \sim 6$) for l.

Table 4.2 tells us that our expansion term weighting method brings significant improvement over all other unweighted versions as well as the baseline: we see nearly 12% improvement over the baseline, and 5-7% over the unweighted version. Increasing expansion level l only slightly improves the retrieval effectiveness.

4.2.2.2 General Expansion

Next, we test several general expansion collections for EMRM. In addition to the medical records that are the target of retrieval, we leverage information in several other large, widely-available collections: ImageCLEF 2009 Medical Image Retrieval Task dataset [83], TREC 2007 Genomics Track dataset [46], TREC 2009 ClueWeb09

Table 4.2: Evaluation of MeSH expansion. "$X > S$" means the MAP difference between system X and any system specified in set S is statistically significant. The statistical significance is determined using one-tailed paired t-test on queries and p-value < 0.05. The scores are based on 5-fold cross validation on the 34 topics from 2011 Medical Records Track.

System	MAP	Significance	bpref	P10
Baseline (B)	0.353		0.469	0.506
Tree1 ($T1$)	0.368 (+4.2%)		0.484	0.509
Entry (E)	0.370 (+4.8%)		0.481	0.553
EntryTree1 ($ET1$)	0.377 (+6.8%)		0.490	0.553
WEntryTree1	0.391 (+10.8%)	>{B, E, $T1$, $ET1$}	0.496	0.547
WEntryTree2	0.394 (+11.6%)	>{B, $T1$, E, $ET1$}	0.498	0.556
WEntryTree3	0.395 (+11.9%)	>{B, $T1$, E, $ET1$}	0.498	0.568
WEntryTree4	0.392 (+11.0%)	>{B, $T1$, E, $ET1$}	0.497	0.556
WEntryTree5	0.391 (+10.8%)	>{B, $T1$, E, $ET1$}	0.497	0.556
WEntryTree6	0.391 (+10.8%)	>{B, $T1$, E, $ET1$}	0.497	0.556

Category B dataset (excluding Wikipedia pages), and a Wikipedia dataset (containing those excluded Wikipedia pages). Table 4.3 provides detailed information about these datasets. In particular, the CLEF dataset consists of 74,902 medical images. We crawled 5,704 full-text CLEF articles associated with these images as the actual external collection used in this work.

The ClueWeb09 dataset was created to support research on information retrieval and related human language technologies. It consists of about 1 billion web pages in ten languages that were collected in January and February 2009. The dataset is used by several tracks of the TREC conference. TREC Category B contains first 50 million English pages[1].

TREC 2007 Genomics Track dataset consists of full-text HTML documents from 49 journals[2] published publish electronically via Highwire Press[3].

[1] Availableathttp://lemurproject.org/clueweb09.php/

[2] The full list of journal can be found at http://ir.ohsu.edu/genomics/2007data.html

[3] http://www.highwire.org/

We choose these collections because there are existing topics and relevance judgments for analysis and because we want to compare the effects of different sources on retrieval performance. Note that although the Genomics dataset is much smaller than the ClueWeb09 dataset, the vocabulary size of both datasets is of the same magnitude.

Table 4.3: Collection statistics for EMRM Model.

Collection	**# documents**	**vocabulary size**	**avg doc length**
Medical*	100,866	10^5	423
ImageCLEF	5,704	10^5	6,495
Genomics	162,259	10^7	6,595
Wikipedia	5,957,529	10^6	1,305
ClueWeb09	44,262,894	10^7	756

For simplicity, we use the aggregation strategy MbR (without ICD, NEG, and AGF, which are all described in Section 3.2) and the retrieval model EMRM with a single expansion collection at a time to explore the expansion effectiveness of each collection as shown in Table 4.4. Note that we use the 34 topics from 2011 Medical Records Track as the training data and perform 5-fold cross validation on them. Then, the rest 47 topics from 2012 Medical Records Track will be used for testing which will be further described in Section 4.2.4.1.

As we can see in Table 4.4, ImageCLEF and Wikipedia have comparable improvement over the baseline, though the former is more medical-related, much smaller, and less noisy than the latter. The same situation applies to the pair of Genomics and ClueWeb09. However, Genomics and ClueWeb09 are much larger than ImageCLEF and Wikipedia respectively, and Genomics and ClueWeb09 both have significant improvement over the baseline. Genomics is also significantly better than Wikipedia. Thus, we can infer that expansion effectiveness depends on both the quality (i.e., content similarity to the target collection) and size of the expansion collection.

In addition, MeSH expansion is different from general expansion in that it relies on a controlled vocabulary from which expansion terms derived are not as diversified as those from a general expansion collection. For instance, for the query "hearing loss",

Table 4.4: Evaluation of single expansion for EMRM. "$X > S$" means the MAP difference between system X and any system specified in set S is statistically significant. The statistical significance is determined using one-tailed paired t-test on queries and p-value < 0.05. The scores are based on 5-fold cross validation on the 34 topics from 2011 Medical Records Track.

System	MAP	Significance	bpref	P10
Baseline (B)	0.353		0.469	0.506
ImageCLEF (I)	0.371 (+5.1%)		0.492	0.544
Wikipedia (W)	0.376 (+6.5%)		0.500	0.550
ClueWeb09 (C)	0.390 (+11%)	$>\{B\}$	0.513	0.556
MeSH (S)	0.391 (+11%)	$>\{B, I\}$	0.496	0.547
Medical (M)	0.393 (+11%)	$>\{B\}$	0.520	0.535
Genomics (G)	0.395 (+12%)	$>\{B, W\}$	0.524	0.553

it is difficult for MeSH to provide related expansion terms such as "cochlear", "noise", "auditory", and "binaural" (top-ranked terms from Genomics), "cerumen", "canals", and "tympanic" (from Medical), "vestibular", "ear", and "stape" (from ImageCLEF). Some of these terms do appear in the MeSH trees at upper levels, however, it is hard to find a link to them, i.e., discriminating them from other unrelated tree nodes. Simply including all visited concepts along the path is likely to cause query drift. Moreover, these terms normally appear in phrase concepts having different meanings than individual terms.

MeSH expansion is quite restrictive, yet is comparable to top performing single expansions and is significantly better than the baseline and ImageCLEF. This is most likely because our MeSH expansion emphasizes modeling term proximity which is a big advantage of any medical thesaurus-based expansion over the general expansion. Another merit of MeSH expansion is that, if used properly, it rarely includes bad expansion terms, while we have no control of the quality of each expansion term from the general expansion.

4.2.3 Impact of Advanced Models

We evaluate the impact of MRF, EMRM, and CMM models by adding them on top of the evidence aggregation method VRM (which is described in Section 3.2.4)

respectively. Based on Table 4.4, we use the Genomics, Medical, MeSH, and ClueWeb09 (i.e., the top-performing expansion collections) together for query expansion in the EMRM model.

Table 4.5 shows the performance of systems of different settings. The MAP differences between 1) VRM+MRF and VRM, 2) VRM+EMRM and VRM, 3) VRM+CME and VRM+MRF, and 4) VRM+CME and VRM+EMRM, are all statistically significant ($p < 0.05$), which indicates that each further improvement significantly boosts the retrieval performance. In particular, EMRM is more effective than MRF. However, since EMRM and MRF are improving the system from different aspects, as expected we obtain a further significant enhancement for the CME model. The final system *MedSearch* (VRM+CME) improves the QL baseline MAP by nearly 20%.

Table 4.5: Impact of Advanced Models. † means statistically significant difference ($p < 0.05$) from the MAP scores of Systems VRM and QL. ‡ indicates statistically significant difference ($p < 0.05$) from the MAP scores of VRM+MRF and VRM+EMRM. System VRM+CME improves the baseline MAP by nearly 20%. The scores are based on 5-fold cross validation on the 34 topics from 2011 Medical Records Track.

System	MAP	bpref	P10	Rprec
QL (baseline q)	0.416	0.551	0.594	0.434
VRM (baseline v)	$0.446_{(+7\%^{q})}$	0.563	0.635	0.456
VRM+MRF	$0.468_{(+13\%^{q}\ \dagger)\ (+5\%^{v})}$	0.585	0.644	0.486
VRM+EMRM	$0.475_{(+14\%^{q}\ \dagger)\ (+7\%^{v})}$	0.611	0.632	0.481
VRM+CME (MedSearch)	$0.501_{(+20\%^{q}\ \dagger\ddagger)\ (+12\%^{v})}$	0.631	0.656	0.505

4.2.4 System Comparison

We have shown that our advanced models outperforms the QL model significantly. In this section, we compare variants of *MedSearch* with other top-performing systems from two separate shared tasks that we also participated in.

4.2.4.1 2012 Medical Records Track

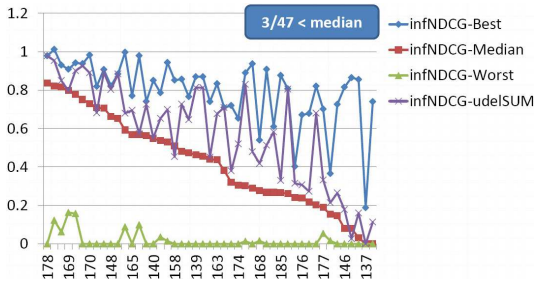
The first shared task is 2012 TREC Medical Record Track in which a total of 82 automatic systems and 6 manual systems were submitted including the 4 automatic

systems from us. Base on previous investigations in Sections 3.2.5, 4.2.2, and 4.2.3, we select and combine multiple features for our final testing on the 2012 TREC Medical Record Track dataset as shown in Table 4.6. The settings for udelMRF and udelMED are for evaluating the impact of MRF and MeSH respectively.

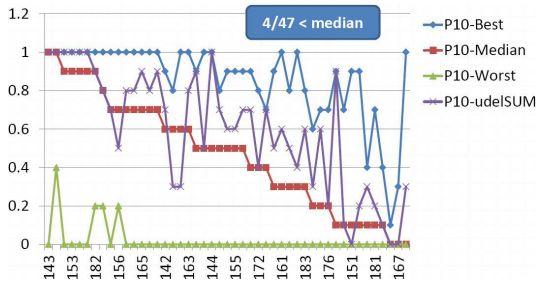
Table 4.6: Feature settings for system variants with results on the 2012 TREC Medical Records Track dataset.

runID	Features				Scores				
	MRF	EMRM		VRM	MAP	infAP	infNDCG	Rprec	P10
		Genomics+Medical+ClueWeb	MeSH						
udelSUM	√	√	√	CombSUM	0.413	0.286	0.578	0.419	0.592
udelMNZ	√	√	√	CombMNZ	0.412	0.285	0.576	0.418	0.594
udelMRF	√	√		CombMNZ	0.408	0.280	0.572	0.415	0.604
udelMED		√	√	CombMNZ	0.398	0.269	0.564	0.410	0.590

Table 4.6 also shows the evaluation scores averaged over 47 official topics. We pick udelSUM, the system with the highest MAP score, for further analysis. Figure 4.3 shows the comparison of infNDCG and P10 scores with TREC results (combining both automatic and manual runs). As we can see, system udelSUM is above TREC medians for the majority of topics. We observe similar results for the other three systems.



(a) **udelSUM** is below TREC medians for 3/47 topics on **infNDCG**.



(b) **udelSUM** is below TREC medians for 4/47 topics on **P10**.

Figure 4.3: Comparison with TREC results.

Table 4.7 shows the results of pairwise one-tail paired t-test on infAP for our four submitted runs. The significance scores indicate that MRF and MeSH are both very effective system features.

Table 4.8 compares the performance between TREC automatic systems from the top-3 ranked participating teams. As we can see, our system *udelSUM* outperforms all

Table 4.7: Pairwise one-tail paired t-test on *infAP*

	udelMNZ	udelMRF	udelMED
udelSUM	0.1635	0.0190	0.0005
udelMNZ	–	0.0335	0.0008
udelMRF	–	–	0.0181

the other systems except one manual system (*NLMManual*) from the National Library of Medicine team. In their work [30], they segmented EMR into sections based on pre-defined topic frames, such as "problem list", "past medical history", "procedure results", etc. Each topic frame slot was assigned a weight between 0 to 1.0. Then they used an internally developed search engine (called Essie) to search over positive text in the sections and combine the weighted scores. They manually and interactively refined the search queries until the top 10 retrieved looked mostly relevant.

Table 4.8: Performance comparison between top-ranked 2012 TREC Medical Records Track systems. Manual systems are marked with *. Our system udelSUM outperforms all the other systems except a manual one from the National Library Medicine team.

Systems	infNDCG	infAP	P10
NLMManual*	0.680	0.366	0.749
udelSUM	0.578	0.286	0.592
sennamed2	0.547	0.275	0.557
ohsuManBool*	0.526	0.250	0.611
atigeo1	0.524	0.224	0.519

The *sennamed2* system was built upon the vector space model with pseudo-relevance feedback and the UMLS concept represented EMR. Although the retrieval model is relatively simple, *sennamed2* showed very good performance.

The *ohsuManBool* system [14] is another manual system whose queries were constructed by relevant ICD-9 codes and phrases for all the conditions in the topic.

The *atigeo1* system [106] benefited mainly from injecting ICD-9 code descriptions and the careful processing (e.g., tag removal, sentence segmentation, tokenization, token normalization, etc.) of the corpus.

The major difference between our system and other automatic TREC systems is that our retrieval model CME deals with the polysemy and synonymy related issues in a more explicit and effective way. In addition, our evidence aggregation methods (described in Chapter 3) allows us to effectively and maximally exploit the useful information contained in EMR.

4.2.4.2 2013 ShARe/CLEF eHealth Evaluation Lab

The second shared task is the Task 3 of 2013 ShARe/CLEF eHealth Evaluation Lab which simulated web searches for health information by patients [38]. The web searches were designed to be connected to hospital discharge summaries from the patient's electronic medical record, thus effectually modeling a post-visit information need. This task differs from both the EMR search and the web search in that it is an ad-hoc retrieval of webpages that mainly contain medical content.

Among the seven systems we submitted to CLEF, one (named *TeamMayo.5.3*) is based purely on the CME model (without using any information from the discharge summaries). In particular, for the query expansion in the EMRM model we used several external sources: the TREC 2011 Medical Records Track test collection, the TREC 2007 Genomics Track test collection, a subset of Mayo Clinic clinical notes (which will be described in Section 5.2.1), and the 2012 MeSH ontology [136].

Table 4.9 compares the performance of *TeamMayo.5.3* with three baselines (namely query likelihood, BM25, and BM25 with pseudo-relevance feedback) and the top performing systems from several other teams. Since it is a web search, P@10 and NDCG@10 are two primary evaluation measures. As we can see, *TeamMayo.5.3* ourperforms other systems by a margin that is large enough to make a big difference for web search tasks.

In particular, *teamAEHRC.5.3* [137] was built upon a Dirichlet-smoothed language modeling provided by the Terrier system. It incorporated additional features such as query spelling correction (using Google Search) and query acronym expansion.

Table 4.9: Performance comparison between top-ranked CLEF systems. Our system *TeamMayo.5.3* ourperforms other systems by a margin that is large enough to make a big difference for web search tasks.

Systems	P@10	NDCG@10
TeamMayo.5.3	0.5040	0.4618
teamAEHRC.5.3	0.4840	0.4226
MEDINFO.1.3	0.4800	0.4377
uogTr.5.3	0.4400	0.3840
BM25_FB	0.4860	0.4328
BM25	0.4700	0.4169
QL (TeamMayo.1.3)	0.4720	0.4408

MEDINFO.1.3 [25] was similar to our *QL* baseline system in that both systems used the unigram language model with Dirichlet prior smoothing on the Indri search engine. The small difference between the scores of *MEDINFO.1.3* and *QL* may result from different settings of the Dirichlet prior and corpus preprocessing.

uogTr.5.3 [66] used Divergence from Randomness and pseudo relevance feedback models within the Terrier framework.

Again, the major advantage of our system compared with other CLEF systems is the effectiveness of our CME model in reducing the vocabulary gap and the term ambiguity in searching unstructured medical text.

4.3 Conclusion

In this chapter, we have introduced several retrieval models specifically designed for alleviating the polysemy and synonymy related issues in medical IR. In particular, the MRF disambiguates word senses and improves the search precision by incorporating contextual information in the query. On the other hand, the EMRM model enhances the recall by deriving query expansion terms from multiple external collections including both in-domain and out-of-domain collections.

The CME model, as a combination of the MRF and EMRM, benefits from the distinct strengths of both models. In particular, the negative interaction between MRF and EMRM is minimal, as Table 4.5 shows that the respective MAP gains from MRF

and EMRM on top of the VRM model approximately sum up to the gain from CME on top of VRM. This is very desirable since it is often hard to improve the precision and recall at the same time. Our systems built upon CME has exhibited this advantage over other systems when performing the same retrieval tasks.

Furthermore, we proposed two different expansion methods for the EMRM model, i.e., the general expansion and the medical thesaurus-based expansion. Particularly for the latter one, we showed that using an external medical query log to weight the expansion terms is very beneficial. We also tested several expansion collections from different domains and found that the size and content similarity of the expansion collections are two important factors that determines the expansion effectiveness.

Chapter 5

EMR SEARCH - DOMAIN KNOWLEDGE

Domain knowledge is very helpful to domain-specific search engines [45]. In the previous chapters, we have already seen a few cases where medical knowledge contributes to retrieval performance, e.g., ICD-9 code expansion and MeSH expansion in the EMRM model. In this chapter, we will further explore how to leverage more medical domain resources and how to use them effectively to improve the search results. In particular, in the first half of the chapter we will describe and evaluate a joint search model that can naturally incorporate medical knowledge in UMLS. In the second half, we will study the utility of a large clinical corpus for query expansion and discuss how to choose effective expansion collections for the EMRM model.

5.1 Joint Search in Text and Concept Spaces

Health search systems typically work in either the "text space", in which queries and documents are free-text and represented as sequences of terms, or the "concept space", in which documents are represented by the medical concepts to which they pertain and users search for those concepts. Traditional retrieval tasks such as web search exist primarily in the text space; we refer to retrieval methods in that space "text-based retrieval" (TBR). In contrast, the concept space is defined by mapping terms uniquely to medical concepts; a query or document comprises a sequence of concepts. We refer to search in the concept space "concept-based retrieval" (CBR). In this section we present a novel and effective system that can search jointly in both of these spaces, and adaptively merge results with respect to the user's query to provide optimal results.

One major benefit of separating the concept space from text space is that it allows us to focus on: 1) improving the CBR by enriching and expanding domain knowledge in the concept space; and 2) transferring any effective text-based retrieval technique to the concept space without worrying about applying medical knowledge since that knowledge already exists in the concept space. In this section, we will demonstrate the former by using an existing medical NLP tool for expanding medical domain knowledge in the concept space, and demonstrate the latter by exploring a novel approach for external expansion in the concept space.

The major challenge in this framework is how to merge the search results obtained by text-based and concept-based retrieval. In this study, we use a learning approach based on features of the text, concepts, and ranked results to adaptively merge for each query based on several heuristics. We will evaluate our algorithm on the official test collections of 2011 and 2012 TREC Medical Records Track.

5.1.1 System Architecture

We propose a novel framework for building medical record search systems that can easily incorporate domain knowledge as shown in Figure 5.1. This framework separates the retrieval space into two, namely the text space and the concept space, allowing the system to search jointly in those two spaces. The meaning of this "joint search" mechanism are twofold: 1) each module in the text space has its counterpart in the concept space as demonstrated in Figure 5.1, and 2) the system can dynamically merge for different queries the results returned by TBR and CBR respectively.

5.1.2 Concept-based Retrieval

In this subsection, we will demonstrate how to effectively transfer two standard text-based retrieval models, the QL model and the EMRM model described by Equations 4.1 and 4.3 respectively, from the text space to the concept space.

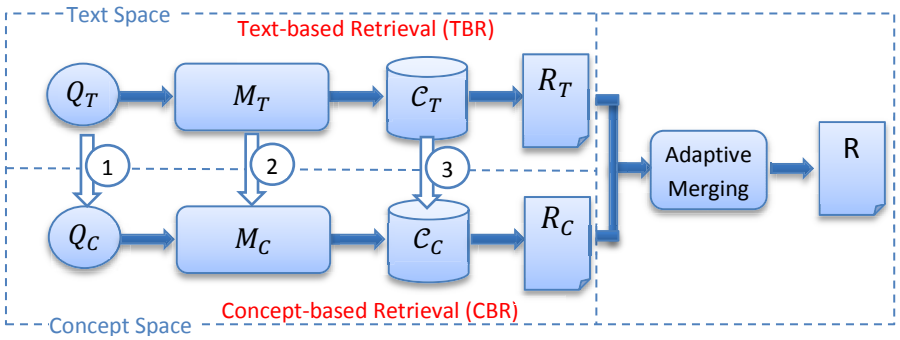


Figure 5.1: A novel framework for building medical record search systems. Q, M, $\mathcal{C}$, R are the query, retrieval model, document collection, and ranked list, respectively. T and C refer to text space and concept space respectively.

5.1.2.1 From Text to Concepts

Performing Text-Based Retrieval (TBR) using the aforementioned retrieval models is straightforward. However, Concept-Based Retrieval (CBR) requires every module in the text space be mapped into their counterparts in the concept space as demonstrated by the labeled processes 1 to 3 in Figure 5.1. We first illustrate processes 1 and 3, i.e., how to map the query and collection to the concept space. In this study, we use MetaMap [9], a medical NLP tool developed at the National Library of Medicine (NLM), to detect concepts in the medical records. For each detected medical concept, MetaMap will return a list of candidates (i.e., concepts) that are represented by the Concept Unique Identifier (CUI) in UMLS Metathesaurus. Thus, processes 1 and 3 in Figure 5.1 involve using MetaMap to convert medical text to a sequence of CUIs in the concept space as shown in Table 5.1. This results in a full second collection of documents composed of CUIs instead of natural language.

Table 5.1: Mapping text to CUI's using MetaMap.

Text:	Chest pain	that feels	like a	heart attack.
Concepts:	C0008031			C0027051
Text:	He has a	history	of	hearing loss.
Concepts:		C0019664		C0011053
Text:	Clear evidence	of	MRSA.	
Concepts:	C0332120		C0343401	

5.1.2.2 Enriching the Concept Space

If we simply transform text into its best matched concepts (as shown in Table 5.1) and use the same ranking function, the performance of CBR is usually not as good as TBR because the concept mapping is imperfect and because important contextual information possibly contained in non-conceptual words is lost. This situations applies to both text queries (Q_T) and documents (D_T), and degrades the performance of CBR. Our idea is that although a "connection" (i.e., matching of text terms) between Q_T and D_T in the text space might be lost after the mapping (e.g., between "hearing" and "hearing loss"), we can still form a new type of "connection" by expanding Q_C and D_C with their related concepts, again using MetaMap.

Table 5.2: CUI candidates for "hearing loss" sorted by the confidence scores.

Scores	CUI	Description
1000	C0011053	hearing loss (Deafness) [Disease or Syndrome]
1000	C0018772	hearing loss (Hearing Loss, Partial) [Finding]
1000	C1384666	Hearing Loss (hearing impairment) [Finding]
861	C0018767	Hearing [Physiologic Function]
861	C1455844	hearing (Hearing examination finding) [Finding]
861	C1517945	Loss [Quantitative Concept]

For each identified noun phrase in the text, MetaMap will return one or more concept candidates depending on the ambiguity of the text phrase as shown in Table 5.2. We propose to expand Q_C and D_C with their CUI candidates whose confidence scores are above thresholds L_Q and L_D respectively. By doing so, we augment the concept space with extra domain knowledge. Conceptually, we are expanding Q_C and D_C with their own related CUIs in the UMLS ontology, and these two sets of CUI's may overlap with each other and thus a new "connection" (i.e., matching of CUIs) is formed between Q_C with D_C. We will discuss the choice of L_Q and L_D on the performance of CBR in Section 5.1.4.

5.1.2.3 Retrieving in the Concept Space

Another aspect of the convenience of applying medical knowledge under our framework is that we can transfer effective techniques from the text space to the concept space, which corresponds to the process 2 in Figure 5.1. For instance, transferring the baseline model to the concept space is straightforward: we simply replace term frequency with "concept frequency", which is the number of times the CUI appears in the translated document; we replace collection term frequency with "collection concept frequency", which is the total number of times the CUI appears in all translated documents. In this way we obtain a query-concept likelihood score:

$$\text{score}_C(D, Q) = \log P(Q_C|D_C) = \sum_{i=1}^{n} \log \frac{\text{cf}_{c_i,D_C} + \mu_C \frac{\text{cf}_{c_i,\mathcal{C}_C}}{|\mathcal{C}_C|}}{|D_C| + \mu_C}, \tag{5.1}$$

where Q_C, D_C, and $\mathcal{C}_C$ represent the concept-mapped query, document, and corpus, respectively, cf_{c_i} is the concept frequency of query concept c_i, $\text{cf}_{c_i,\mathcal{C}_C}$ is the collection concept frequency of query concept c_i, and μ_C is a Dirichlet smoothing parameter for the concept space.

The full EMRM model based on Equations 4.3 and 4.4 can similarly be transferred to the concept space.

To let the expansion collections work for EMRM the concept space, a direct approach is to first map the text collections into concept collections, and then apply the same method as described in Section 4.1.2. However, this approach has two major limitations. First, the size of an effective external expansion collection is usually much larger than the size of the target collection. Converting it into a concept collection is time-consuming and costly (complex phrases and sentences may require hours of computation due to the thoroughness of MetaMap [11]). Second, in some cases external expansion collections may not be fully accessible (e.g., only top-ranked results from some commercial search engine are available via APIs), and it becomes impossible to map them into concept collections completely.

Thus, we propose an alternative approach in which we first use the Text-based Retrieval to obtain the initial top-ranked documents for expansion from the external

collection. Then we map those expansion documents to the concept space. Finally, we can apply the full EMRM model directly in the concept space. However, we still do not have the statistics about the whole expansion concept collection in order to compute the inverse document frequency in Equation 4.4. We propose two possible solutions: one is to use the statistics of the target concept collection, the other is to sample the expansion text collection using a set of sample queries and then build a small concept collection based on the returned documents. In this study, we use the latter approach.

5.1.3 Adaptive Joint Search in Text and Concept Spaces

Given a query Q and a document collection $\mathcal{C}$, the TBR and CBR methods described in Section 5.1.2 will return two separate sets of search results. We next need to find a way to combine them into a single ranked list suitable for a user. We formulate this problem as a linear interpolation between the scores of documents in the two sets of results:

$$\text{score}(D, Q) = \alpha_Q \cdot \text{score}_T(D, Q) + (1 - \alpha_Q) \cdot \text{score}_C(D, Q), \tag{5.2}$$

where score_T and score_C are relevance scores of TBR and CBR respectively (both calculated by the language model with query expansion described in Section 5.1.2), and $\text{score}(D)$ is the merged score for D in the returned ranked list. Thus, our goal is to optimize the coefficient α_Q for different queries. We use a learning approach to predict α_Q based on several heuristics.

5.1.3.1 Learning Algorithm

We can view α_Q in Equation 5.2 as a mixing probability: the probability that the query has been sampled from document text rather than translated from a concept sampled from the document's concept space. Then, assuming the log-odds of that probability can be expressed as a linear combination of feature values: $\log \frac{\alpha_Q}{1-\alpha_Q} = \beta_0 + \sum_{i=1}^{m} \beta_i x_i$, where β_0 is a model intercept (or bias term), x_i is the value of feature number i, and β_i is the weight coefficient of that feature. This is essentially a logistic

regression model[1]. Logistic regression is fit using iteratively reweighted least squares to find the values of the β coefficients that are the best fit to training data. Given feature values and their β coefficients, we can then predict the mixing probability α_Q for new queries.

5.1.3.2 Features

We propose a number of features that are heuristically related to the performance of TBR and CBR, and can be used to predict the result merging coefficient α_Q in Equatioin 5.2.

Length of the text query: Generally a long query is more discriminative than a short one since the former contains more information [73]. Thus, we use the length of text query $|Q_T|$ as the feature to estimate the performance of TBR. It is defined formally as

$$|Q_T| = \sum_{w \in Q_T} \text{cnt}(w, Q_T), \tag{5.3}$$

where $c(w, Q_T)$ is the count of term w in Q_T.

Length of the concept query: The intuition for using feature No.1 applies in the concept space as well. The length of concept query tells how discriminative concept query is, and thus relates to the performance of CBR. We define this feature formally as

$$|Q_C| = \sum_{w \in Q_C} \text{cnt}(w, Q_C), \tag{5.4}$$

where $\text{cnt}(w, Q_C)$ is the count of term w in Q_C.

Concept ratio: This feature computes the proportion of concept-related words in the original query, and is defined as

$$R_C = \frac{\sum_{w_c \in Q_T} \text{cnt}(w_c, Q_T)}{|Q_T|}, \tag{5.5}$$

[1] While logistic regression is often used for 0/1 classification problems, it can also be used when the target variable is a real number between 0 and 1. In this case it is sometimes called a "quasibinomial" model.

where w_c is a concept-related term. For example, if the query is "elderly people with a history of hearing loss", MetaMap will detect the concept "hearing loss" as a phrase. Then, "hearing" and "loss" are both concept-related terms. Since only concept-related terms can make an impact on document scoring in the concept space, CBR will generally have a better performance if the concept ratio of the text query is higher.

Concept expansion ratio: A text query can contain several noun phrases, for each of which MetaMap may return a set of candidates. We hypothesize that a noun phrase is more difficult for TBR if it has more CUI candidates returned by MetaMap, because having more candidates means that the noun phase is more ambiguous, and finding those counterpart candidates in the text space will be harder. Thus, the average number of returned candidates for concepts in a query may be a good indicator of the text query difficulty. Since the candidates are also expansion concepts as described in Section 5.1.2.2, we call this feature the concept expansion ratio, defined as

$$ER_C = \frac{|Q_C|}{\sum_{w \in Q_C} |\text{Meta}(w)|}, \tag{5.6}$$

where $|Q_C|$ the original concept query length (i.e., the length before expansion), and $|\text{Meta}(w)|$ is the number of concept candidates returned by MetaMap for term w in concept query Q_C.

Weighted pseudo-AP difference: Our final feature is based on an estimate of differences in retrieval effectiveness. We first define the set as $\mathcal{D}$ which contains the common documents shared in the top k retrieved of the two ranked lists returned by TBR and CBR respectively. We argue that documents in $\mathcal{D}$ are more likely to be relevant than other documents in both ranked lists.

Thus, we treat $\mathcal{D}$ as a pseudo-relevance set (i.e., assuming all the documents in $\mathcal{D}$ are relevant) and compute a pseudo-AP (average precision) score based on $\mathcal{D}$ for each query in TBR and CBR respectively. This pseudo-AP score is expected to be a good indicator of the true retrieval performance of TBR and CBR. We use the

weighted pseudo-AP difference as our last feature, which is defined as

$$\text{WAP} = \frac{|\mathcal{D}|}{k}(\text{PAP}_T(\mathcal{D}, Q) - \text{PAP}_C(\mathcal{D}, Q)) \tag{5.7}$$

where $\text{PAP}_T(\mathcal{D}, Q)$ and $\text{PAP}_C(\mathcal{D}, Q)$ are the pseudo-AP scores of query Q for TBR and CBR respectively, and $|\mathcal{D}|$ is the size of the relevance set. Intuitively, if $|\mathcal{D}|$ is small, PAP will be more sensitive to the ranks of documents in $\mathcal{D}$. Thus, we use $|\mathcal{D}|/k$ to dampen this effect caused by small $|\mathcal{D}|$. In this work, we experimentally set k to 300 because we observe that $|\mathcal{D}|$ increases relatively slow when k rises above 300.

5.1.4 Evaluation

5.1.4.1 Experimental Setup

We use the Indri[2][104] retrieval system for indexing and retrieving in both the text space and the concept space. In particular, we use the Porter stemmer to stem words in both text documents and queries, and use a simple standard medical stoplist [44] for stopping words in queries only. Note that in the concept space as described in Section 5.1.2, we do not need to do stemming and stopping since the vocabulary in the concept space consists of concept unique identifiers only.

To evaluate the baseline model and EMRM model in both text and concept spaces (i.e., M_T and M_C in Figure 5.1), we conduct 9-fold cross-validation and use the top 1000 retrieved visits (top 1000 is a TREC standard) for each query to evaluate our system under different settings.

In each iteration of the 9-fold cross validation, we train our system on 72 queries to obtain the best parameter setting for mean average precision (MAP) by sweeping over the parameter space ((1000, 16000, 5000) for μ, (20, 100, 20) for k, and (10, 20, 5) for m. The last number in this pair of parenthesis is the step size), and then generate a ranking for each of the remaining 9 queries based on the trained system. When complete, we have full rankings for all 81 topics as a test set. Note that we do

[2] http://www.lemurproject.org/indri/

cross-validation for both TBR and CBR, and thus obtain separate parameter values (i.e., μ, k, and m) for TBR and CBR.

Finally, we evaluate the system based on the average evaluation scores over all 81 topics. This corresponds to evaluating the ranked lists R_1 and R_2 in Figure 5.1.

To evaluate our learning algorithm for the result merging, we first obtain the optimal coefficient $\alpha_{Q\text{-opt}}$ for each topic Q by sweeping [0, 1] (i.e., the valid range of α_Q) at a step size of 0.1. Then we conduct leave-one-out cross-validation (LOOCV), in each iteration of which the system predicts the coefficient α_Q for one new topic based on $\alpha_{Q\text{-opt}}$'s for the other 80 topics.

Similar to previous chapters, we train our systems on MAP and perform one-tailed paired t-test on MAP scores to access the statistical significance of MAP improvement. We also report scores for R-precision (Rprec), bpref, and precision at rank 10 (P10).

5.1.4.2 Expanding the Concept Space

As discussed in Section 5.1.2.2, we can expand queries and documents respectively with their CUI candidates returned by MetaMap. In this section, we evaluate the effectiveness of this approach for improving retrieval performance.

We run 9-fold cross-validation on the baseline retrieval model in the concept space. We vary the confidence score threshold from 400 (all the confidence scores we observed are above 400) to 1000 at a step size of 100 for both L_D and L_Q. Table 5.3 shows the MAP scores for different threshold settings. The rows show that MAP has a negative correlation with L_D, indicating expanding the documents always helps. The columns also suggest expanding queries with related CUIs is very helpful, except when $L_Q = 1000$ and 900, which further indicates that when the documents are not well expanded aggressively expanding only queries may cause query drift and lead to severe performance degradation.

Further analysis shows that setting L_D and L_Q both to 500 results in the best performance and is significantly better ($p < 0.05$) than other settings where $L_D > 600$

or $L_Q >= 600$. This tells us that enriching the concept documents with more related domain information improves retrieval performance. Thus, we use 500 as the default value for L_D and L_Q unless otherwise specified.

Table 5.3: MetaMap for concept expansion.

		L_D						
		1000	900	800	700	600	500	400
L_Q	1000	0.212	0.236	0.259	0.250	0.265	0.265	0.265
	900	0.187	0.234	0.259	0.259	0.266	0.264	0.264
	800	0.167	0.210	0.297	0.298	0.312	0.310	0.310
	700	0.169	0.194	0.284	0.296	0.311	0.310	0.310
	600	0.159	0.185	0.285	0.298	0.321	0.320	0.320
	500	0.160	0.185	0.286	0.300	0.322	**0.333**	0.333
	400	0.160	0.185	0.286	0.300	0.322	0.322	0.322

The purpose of choosing the best confidence score thresholds is that we want to build on a strong baseline to demonstrate the effectiveness of other retrieval methods in this work.

5.1.4.3 EMRM in Text and Concept Spaces

We evaluate the effectiveness of EMRM in both text and concept spaces. We use two expansion collections for EMRM model: the medical record collection itself (i.e., the target collection) and the 2007 TREC Genomics Track dataset.

Before discussing results of the comparison, we compare some key statistics of the text space and concept space in Table 5.4. It is interesting to note that while the base concept collection is much smaller than the text collection, the expanded concept collection $\mathcal{C}_C$ becomes almost 50% larger than $\mathcal{C}_T$. Expanded concept queries are more than twice as long as their text-based counterparts, and nearly four times as long as the non-expanded concept queries.

Table 5.5 show the EMRM model works very well in the text space. We obtain significant improvement when just using a single expansion collection. Using both collections in the EMRM model further improves the performance significantly.

Table 5.4: Statistics comparison of text and concept spaces. The expanded concept collection $\mathcal{C}_C$ becomes almost 50% larger than $\mathcal{C}_T$. Expanded concept queries are more than twice as long as their text-based counterparts, and nearly four times as long as the non-expanded concept queries.

	$\mathcal{C}_T$	$\mathcal{C}_C$ ($L_D = 1000$)	$\mathcal{C}_C$ ($L_D = 100$)
# uniq. terms	83,978	43,711	62,703
# total terms	33,454,213	13,347,125	47,540,697
# avg document length	1945	744	2,764
	Q_T (after stopping)	Q_C ($L_Q = 1000$)	Q_C ($L_Q = 100$)
avg length	5.3	3.7	14.9

Table 5.5: Effectiveness of EMRM in text and concept spaces. * means the MAP difference from the baseline is statistically significant ($p < 0.05$). † means that the MAP score is significantly better ($p < 0.05$) than other systems. $^{\triangle}$ means the score is significantly better ($p < 0.05$) than the baseline score.

Text-based Retrieval	**MAP**	**R-prec**	**bpref**	**P10**
baseline	0.363	0.382	0.434	0.541
EMRM (Target)	0.379*	0.392	0.450	0.558
EMRM (Genomics)	0.377*	0.391	0.446	0.563
EMRM (Target + Genomics)	0.392†	0.407	0.461	0.572
Concept-based Retrieval	**MAP**	**R-prec**	**bpref**	**P10**
baseline	0.333	0.352	0.420	0.519
EMRM (Target)	0.339	$0.364^{\triangle}$	$0.431^{\triangle}$	$0.538^{\triangle}$
EMRM (Genomics)	0.337	$0.366^{\triangle}$	$0.432^{\triangle}$	$0.546^{\triangle}$
EMRM (Target + Genomics)	0.350†	$0.372^{\triangle}$	$0.445^{\triangle}$	$0.549^{\triangle}$

In the concept space, there is no significant improvement on the MAP score for EMRM using a single expansion collection. This might be because the query has already been expanded with MetaMap concept candidates and thus EMRM does not help much for MAP (Our further analysis shows that the MAP difference is statistically significant ($p < 0.05$) when $L_Q \leq 600$ or $L_D \geq 600$). Nevertheless, the score difference on other evaluation metrics is statistically significant; moreover, EMRM with two expansion collection improves the MAP significantly.

Overall, TBR is better than CBR. However, TBR and CBR have varied performance on different topics: while TBR is better on average, CBR outperforms it

for quite a few topics. The ideal system should take advantage of both types of retrieval and adaptively merge their results to achieve further improvement. This is our motivation for proposing the adaptive result merging algorithm which we will analyze next.

5.1.4.4 Adaptive Result Merging

Now we evaluate our learning algorithm described in Section 5.1.3.1.

Best Fixed Coefficient

We first compute the retrieval performance for result merging using a fixed coefficient for all topics. We sweep α from 0 to 1 with a step size of 0.1, as demonstrated by Figure 5.2. We find that $\alpha_{\text{best-fixed}} = 0.7$, which is expected since the overall performance of TBR is better than CBR. Retrieval performance with this value is given in Table 5.6.

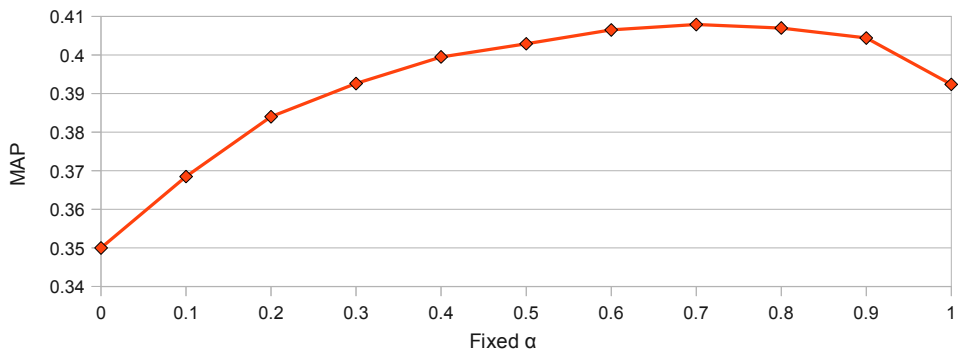


Figure 5.2: Finding the best value for the combination parameter α in the 'Best-fixed' strategy.

Optimal Coefficient

We also obtain the optimal $\alpha_{Q\text{-opt}}$ for each topic separately by sweeping α from 0 to 1 with a step size of 0.1. Then, we use the $\alpha_{Q\text{-opt}}$'s to compute the best retrieval performance (i.e., an upper-bound) we can achieve by our adaptive merging algorithm, as shown in Table 5.6.

Table 5.6: Adaptive Result Merging. * means "Adaptive" is significantly ($p < 0.05$) better than "Best-fixed". The last column is the mean square error of the predicted weights.

α	**MAP**	**R-prec**	**bpref**	**P10**	Pred. MSE
Best-fixed	0.408	0.411	0.471	0.619	0.124
Adaptive	0.416*	0.424	0.481	0.624	0.098
Optimal	0.441	0.447	0.512	0.637	0.000

Performance Comparison

We compare the performance of our adaptive merging method with the fixed weighting method and the optimal weighting method in Table 5.6. As we can see, our adaptive merging method outperforms the fixed weighting method on all the evaluation metrics, and the improvement on MAP is statistically significant. Furthermore, the adaptive merging method outperforms our best TBR result, indicating that both text and concept space can contribute to a good retrieval system.

5.2 Using Large Clinical Corpora for Query Expansion

In the EMRM approach to query expansion described in Section 4.1.2, multiple large external text corpora from general-domain collections have been shown to select reasonable terms and improve retrieval performance. What sort of improvement, if any, should be expected if *clinical-domain* collections are used for this query expansion?

In this section, we analyze the effects of including a large, unlabeled corpus of clinical notes (as another way of exploiting domain knowledge) into an statistical IR system for cohort identification. In particular, we evaluate the helpfulness of a corpus of Mayo Clinic clinical notes for the TREC task of IR-based cohort identification, considering the effects of collection size, the inherent difficulty of a query, and the interaction with other widely-available collections.

5.2.1 Auxiliary Collections for Query Expansion

This study mainly performs an analysis based on a clinical text collection: a 39 million-document subset of Mayo Clinic clinical notes between 1/1/2001–12/31/2010,

retrieved from the Mayo Clinic Life Sciences System (MCLSS). This includes data from a comprehensive snapshot of Mayo Clinic's service areas, excluding only microbiology, radiology, ophthamology, and surgical reports. Additionally, each possible note type at Mayo was represented: Clinical Note, Hospital Summary, Post-procedure Note, Procedure Note, Progress Note, Tertiary Trauma, and Transfer Note. This corpus has been characterized for its clinical information content (namely, medical concepts[117] and terms[118]) and compared to other corpora, such as the 2011 MEDLINE/PubMed Baseline Repository and the 2010 i2b2 NLP challenge dataset[109].

Table 5.7: Collection Statistics

Collection	**# documents**	**vocabulary size**	**avg doc length**
PittNLP*	100,866	10^5	423
Genomics	162,259	10^7	6,595
ClueWeb09	44,262,894	10^7	756
MayoClinic	39,449,222	10^6	346

In addition to the medical records from Mayo Clinic, we leverage information in several other large, widely-available collections: the TREC 2007 Genomics Track dataset [46], the TREC 2009 ClueWeb09 Category B dataset, and the Pittsburgh NLP Repository itself (the target collection, as indicated by * in Table 5.7).

Table 5.7 provides statistics about these datasets. The ClueWeb09 Cat-B dataset has comparable size to Mayo Clinic dataset in terms of the number of documents, however, it is less similar in content to the target collection (i.e., the Pittsburgh NLP Repository) and is considered more noisy than Mayo Clinic dataset. The Genomics dataset is much smaller than the ClueWeb09 Cat-B dataset, however, the knowledge domain where the Genomics dataset comes from overlaps more with the clinical domain than the general web domain where the ClueWeb dataset is derived from.

5.2.2 Experimental Setup

We use the query likelihood (QL) language model (Equation 4.1) as the baseline, and the EMRM model (Equation 4.3) for query expansion.

For corpus preprocessing, we used the Porter stemmer and a simple standard medical stoplist [44] for stemming and stopping words in queries during retrieval. Then we conducted 9-fold cross-validation and used the top 1000 retrieved visits[3] for each query to evaluate our system under different settings. In each iteration, we trained our system on 72 queries to obtain the best parameter setting for MAP by sweeping over the parameter space according to Table 5.8 below, and then generate a ranking for each of the remaining 9 queries based on the trained system. When complete, we had full rankings for all 81 topics as a test set. We evaluated the system based on MAP over all 81 topics.

Table 5.8: Parameter space for training.

Parameter	**From**	**To**	**Step Size**
Dirichlet smoothing parameter μ	1000	20000	5000
# of feedback documents k	20	60	20
# of expansion terms m	10	30	10

Note that the baseline system using Equation 4.1 has only one free variable μ to train. In this work, we fix expansion weight λ_Q to 0.7 and use equal weights for λ_C. This is because we need to test various system settings with multiple parameters. Including λ in training will be computationally expensive when two or more expansions collections are used the mixture of relevance models. In fact, expansion collection weighting itself is an interesting research problem and we plan to explore it in our future work.

To assess the statistical significance of differences in the performance of two systems, we perform one-tailed paired t-test for difference in MAP.

5.2.3 Evaluation

In this section, we show and discuss the results of including the Mayo Clinic corpus under various settings.

[3] Medical Records track guideline requires each retrieval set contain no more than 1000 visits.

5.2.3.1 Clinical Corpus vs. Other Single Collections

Table 5.9 shows the retrieval performance when a single collection was used to produce query expansions.

Table 5.9: MAP scores for single expansion collections, and the significance of their differences (p value).

Collection	PittNLP	Genomics	ClueWeb	Baseline	**MAP score**
Mayo	0.225	0.125	0.077	8.39×10^{-07}	**0.391**
PittNLP		0.363	0.354	2.50×10^{-04}	**0.388**
Genomics			0.443	1.12×10^{-05}	**0.387**
ClueWeb				1.57×10^{-06}	**0.386**
Baseline					**0.373**

It can be seen that the best single MAP score is using the Mayo Clinic corpus. This is particularly interesting because it outperforms the target collection (PittNLP) itself, though the difference is not statistically significant. The Mayo Clinic data does significantly (at the $p < .10$ level) outperform ClueWeb, showing the domain of similar-sized corpora matters.

In these single expansion collection tests, the similarity of the collection appears to be a suitable measure of quality. Similar corpora will tend to reduce noise and so improve precision; while dissimilar corpora will attempt to increase recall with novel terms but contribute noise, thus hurting precision.

5.2.3.2 Performance by Collection Size and by Query Difficulty

To test the impact of the collection size on the query expansion effectiveness, we created multiple expansion collections of different size in an incremental way based on the original Mayo Clinic corpus. In particular, we built the smallest sub-collection C_0 by randomly sampling a set of clinical notes in the Mayo Clinc corpus, and then built the next sub-collection C_1 by adding more clinical notes that are randomly selected, and then built C_2 by adding more notes to C_1, and so on. Thus C_j is a superset of C_i for $i < j$. We built an index for each sub-collection and use it for query expansion. The number of terms in each sub-collection is shown on the x-axis

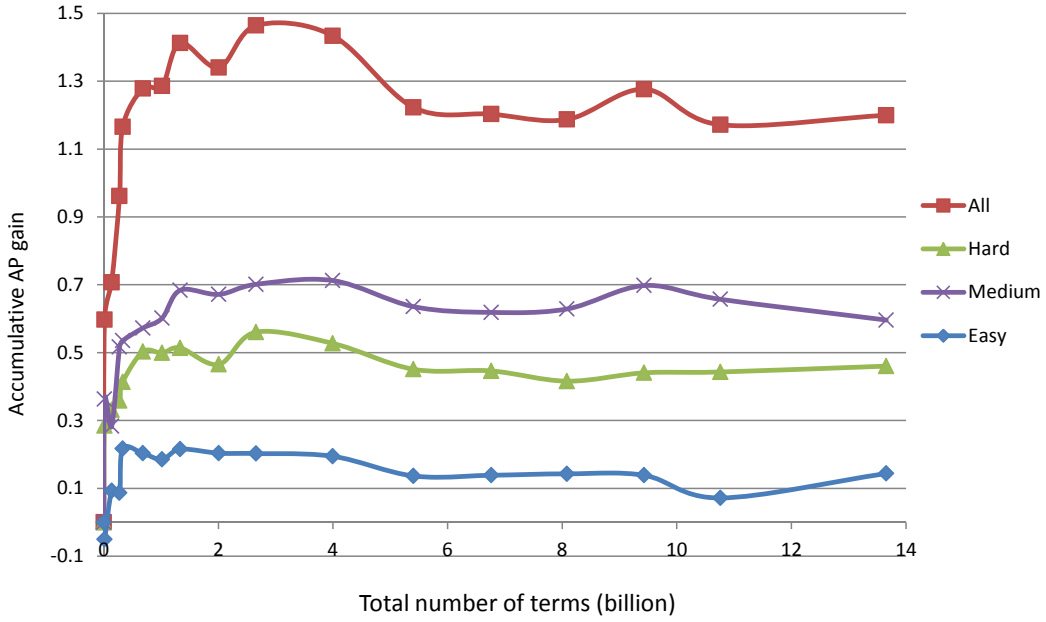


Figure 5.3: Performance curve of incorporating different-sized clinical collections as relevance models for query expansion.

in Figure 5.3. Figure 5.3 further shows the accumulated AP gain on the y-axis, where the accumulated AP gain is the sum of MAP score improvements. We have divided the queries into three classes, based on their performance (without any query expansion): *hard* = MAP< .33; *medium* = .33 <MAP< .67; and *easy* = MAP> .67. It is clear that "more is better" does not hold here. There is a peak at about 2.5 billion terms, at which the Mayo clinical notes no longer contribute positive query expansions, and they are more likely contributing noise instead. This is an interesting result, because it counters the common wisdom that more data will always solve the problem. In the case of query expansion, it is helpful to have the right amount of in-domain information.

Figure 5.3 also shows that the beneficial effects of query expansion are less pronounced for *easy* queries. Because the query difficulty categorizations were made without query expansion, *easy* queries are already able to be retrieved without the help of query expansion. Medium and hard queries are more helped.

5.2.3.3 Clinical Corpus among Multiple Expansion Collections

In this experiment, we compared the individual performance of several expansion collections listed in Table 5.7. Since we had 4 expansion collections in total, there are 11 different combinations of two or more expansion collections.

Table 5.10: Using multiple expansion collections (PittNLP, ClueWeb09, TrecGenomics, Mayo Clinic) for extended mixture of relevance models (EMRM) query expansion

	MAP		Expansion Collection			
			PittNLP	ClueWeb	Genomics	Mayo
EMRM-1	0.4011	2 coll.	X	X		
EMRM-2	0.4031		X		X	
EMRM-3	0.3996		X			X
EMRM-4	0.3979			X	X	
EMRM-5	0.3987			X		X
EMRM-6	0.3943				X	X
EMRM-7	0.4144	3 coll.	X	X	X	
EMRM-8	0.4089		X	X		X
EMRM-9	0.4116		X		X	X
EMRM-10	0.4061			X	X	X
EMRM-11	0.4223	4 coll.	X	X	X	X

First, we note that the 4-corpus EMRM-11 run achieves the highest MAP score of any tested combination. Thus, while Figure 5.3 showed that bigger was not necessarily better for the size of the clinical corpus, more multi-corpus data was helpful to performance.

Furthermore, Table 5.10 suggests that the *diversity* of expansion collections should be considered when dealing with more than one expansion collection. With multiple expansion collections, the effect of noisy terms from a single collection can be mitigated by terms from other collections, as long as the collections are diverse. Thus, diversity (appropriate dissimilarity in domains) counterbalances similarity, providing additional recall with suppressed loss of precision.

Thus, while an in-house clinical collection (here, the Mayo corpus) is the single most beneficial resource (according to Table 5.9), it is not necessarily the best resource in a multiple expansion collections case. However, because in-house collections may be

the most available resource within an institution (no subscription or data use agreement), the next subsection explores whether it is worth the effort to produce such an in-house collection.

5.2.3.4 Adding a Clinical Corpus to an Existing Setup

We ran query expansion using multiple collections and computed the relevance scores according to Equation 4.3. Using the 11 runs calculated in Table 5.10, we considered the significance of adding the Mayo corpus given that one or more of the other expansion collections were already present. This is a realistic setting when implementing an IR-based cohort identification system with a local EMR. The significance of adding the Mayo clinical corpus is very clear. Regardless of what collections have been used for the mixture of relevance models, results will be improved by adding the corpus. This implies that any locally-implemented IR-based cohort identification system can significantly improve its performance by utilizing a large unlabeled corpus within their institution.

Table 5.11: Change in performance (Δ MAP) and significance (p-values $< .05$), upon adding the clinical corpus to any existing configuration.

EMRM Model	Δ MAP	p-value
PittNLP adding Mayo	0.0117	2.66×10^{-05}
ClueWeb adding Mayo	0.0124	0.000513297
Genomics adding Mayo	0.0075	0.003126875
PittNLP + ClueWeb adding Mayo	0.0078	0.004912243
PittNLP+Genomics adding Mayo	0.0085	0.005416947
ClueWeb+Genomics adding Mayo	0.0082	0.015875188
PittNLP + ClueWeb + Genomics adding Mayo	0.0162	0.023945989

5.2.4 Discussion

5.2.4.1 Analysis of Performance Factors

The quality, size, and diversity of the expansion collections are three important factors that impact performance gain.

First, larger expansion collections tend to have a better coverage of query-related expansion terms. However, an expansion collection can also introduce more noise if it is too large. Table 5.12 shows the top weighted expansion terms (word stems) for the query "hearing loss". The first three columns are terms derived from Mayo sub-collections of different sizes. As we can see, M30 produces a much better set of expansions terms than M10. However, the set derived from M80 is apparently contaminated by noise.

The quality of expansion collection is estimated by the overlap between two domains, i.e., the content similarity of the expansion collection to the target collection. Expansion collections containing similar content to the target collection tend to use a similar underlying language model (i.e., vocabulary and term distributions) and thus can derive a better "relevance model".

Moreover, a diversified set of expansion collections work better than a specialized set of collections. This is because expansion collections from different domains contribute differently to the retrieval performance with respect to different queries. If one collection in that diversified set fails to improve retrieval the others might still help (as shown in Table 5.12), which is not the case if we use a set of similar collections.

5.3 Related Work

Most participants in TREC Medical Records Track tried using medical knowledge to enhance retrieval, but only a few of them achieved positive results. King et al. [55] identified and indexed terms of medical reports that appeared in the UMLS. Meanwhile, they expanded original queries with related terms in UMLS and several commercial medical reference encyclopedias. Goodwin et al. [39] leveraged information from SNOMED-CT, UMLS, and a subset of PubMed Central database for query expansion. These three teams all obtained large improvement over their baselines which used no medical-specific knowledge.

Demner-Fushman et al. [31] expanded query terms with UMLS synonyms and

Table 5.12: Comparison of top 15 expansion terms for query "hearing loss".

M10 (10% Mayo)	M30 (30% Mayo)	M80 (80% Mayo)	ClueWeb	Genomics
ear	ear	sensorineur	heare	hear
sensorineur	sensorineur	inherit	shakespeare	deaf
aid	aid	gene	herbert	hhie
gene	audiogram	connexin	campion	cochlear
audiogram	hi	ear	nniina	ear
inherit	nois	autosom	jokinen	sensorineur
genet	right	genet	alphabeticall	loss
tinnitu	cochlear	recess	cawdrey	ttss
hi	tinnitu	aid	tiiaa	nois
caus	left	ag	ierde	syndrom
connexin	bilater	slope	george	paget
nois	sudden	matern	hiele	audiometr
baud	ha	mutat	babel	fechter
carrier	db	famili	renee	cochlea
mitochondria	hz	patern	har	auditori

MeSH terms and expanded drug related terms using RxNorm and Goolge search. However, their knowledge-based system built upon the open-source Lucene[4] system did not improve over a simple baseline. In a few other cases of using query expansion, Daoud et al. [28] used UMLS, Wu et al. [116] used disease and symptom descriptions from a healthcare website, and Schuemie et al. [97] used UMLS and DrugBank. However, they all obtained very little or no improvement over their baselines.

5.4 Conclusion

In this chapter, we explored how to use both structured and unstructured domain resources to improve EMR search.

In the first part, we proposed and evaluated a joint searching framework for building an EMR search system in which we can flexibly apply structured medical domain knowledge. In particular, after transforming text into UMLS CUI concepts we can easily explore and build new types of connections between query and document concepts by expanding them with related concepts using MetaMap, which is the main

[4] http://lucene.apache.org/core/

advantage of our concept-based retrieval (CBR) method compared with the text-based retrieval (TBR). However, CBR suffers from losing certain amount of contextual information during the text-to-concept transformation, and furthermore its effectiveness depends on the performance of MetaMap. Nevertheless, the performance of CBR and TBR varies a lot with respect to different queries. Our post-analysis shows that CBR outperforms TBR for about one third of topics.

Therefore, out joint search strategy learns to strike a balance between CBR and TBR with respect to different queries in order to get the benefits from both CBR and TBR. In particular, we have shown that the helpful features for predicting the combination weight between CBR and TBR should be indicative the performance of CBR versus the TBR with respect to different queries. Our cross-validation results show that our adaptive CBR and TBR merging algorithm is more effective than a well-tuned fixed-weight merging algorithm, and furthermore, there is still plenty of room for further improvement after comparing the 'adaptive' merging strategy with the upper-bound 'optimal' strategy. For future work, we will focus on exploring more features on a larger test collection.

In the second part of this chapter, we further investigated the criteria of selecting good free-text expansion collections for the EMRM model, following what we learnt from Chapter 4. We showed that the expansion effectiveness for the EMRM model depends on several properties, namely the size, the content similarity (or quality), and the number and diversity of expansion collections. In general, large collections outperform small collections, and in-domain collections are better than out-of-domain collections. However, we also showed that more data is not necessarily better for query expansion, implying that there is value in collection curation. Furthermore, a set of collections from diverse domains tend to work better than a set of similar collections since if one of the collections fails to provide good expansion terms the other collections can still come to the rescue.

We also studies the usefulness of Mayo Clinic corpus for query expansion in the EMRM model. As a large size, in-domain collection, the Mayo Clinic corpus is always

beneficial when added into any existing settings of EMRM model.

Chapter 6

MESH INDEXING

In this chapter, we will investigate how IR can help reduce human effort in the biomedical sematic indexing task.

6.1 Background

MEDLINE[1] is the U.S. National Library of Medicine's (NLM) premier bibliographic database that contains over 19 million references to journal articles in life sciences with a concentration on biomedicine. MEDLINE records are indexed with Medical Subject Headings (MeSH) and by highly qualified domain experts.

Currently, there are about 0.7 million new journal articles being added to the MEDLINE databsese each year, which makes manual indexing extremely difficult and costly. Besides, the indexing consistency among domain experts is unpredictable and hard to control. Funk and Reid [35] reported a consistency of only 48.2% for MeSH-based indexing. Moreover, the relatively slow speed of indexing new articles and making them available in the search database hinders technology transfer and advancement more or less.

In order to alleviate those problems, the NLM has developed a tool called Medical Text Indexer (MTI) to assist human annotators with MEDLINE article indexing [12]. Recently, the BioASQ challenge [107, 8] has initiated a series of shared tasks, among which Task 1a (Large-scale Biomedical Indexing) specifically targets on the MEDLINE indexing problem and encourages participants to contribute to the development of tools and systems to automatically suggest MeSH terms to MEDLINE literature.

[1] http://www.nlm.nih.gov/pubs/factsheets/medline.html

In this work, we propose three approaches, one building upon another in an incremental way, to automatic MeSH term suggestion: 1) MetaMap-based labeling, which relies on the MetaMap tool to detect MeSH-related concepts for indexing; 2) Search-based labeling, which builds on MetaMap-based approach and further leverages information retrieval techniques for finding similar articles whose existing annotations are used for MeSH suggestion; 3) LLDA-based labeling, which further trains a multi-label classifier based on MeSH ontology for MeSH candidate list pruning. The evaluation on the BioASQ challenge data presents promising results and produces interesting findings that may benefit future exploration.

The rest of the chapter proceeds as follows: Section 6.2 highlights the related work. Section 6.3 describes the data and the task. Then, Section 6.4 elaborates our methods and Section 6.5 presents and discusses the evaluation results. Finally, Section 6.6 summarizes our work and points out future research directions.

6.2 Related Work

There are many existing works related to MeSH-based MEDLINE indexing. We will only highlight a few that are most relevant to our approaches in this section.

The most well-known system for MeSH indexing is the Medical Text Indexer (MTI) developed at NLM [54, 12]. The latest version of MTI consists of three major components: MetaMap [9], Trigram Phrase Matching, and Trigram PubMed Related Citations (Trigram PRC) [68]. MetaMap is a tool that can map text into UMLS concepts, represented by Concept Unique Identifiers (CUI). Trigram Phrase Matching[2] is a method of identifying phrases that have a high probability of being synonyms. It is based on the idea of representing each phrase by a set of character trigrams that are extracted from that phrase. The character trigrams are used as key terms in a representation of the phrase much as words are used as key terms to represent a document. The similarity of phrases is then computed using the vector cosine similarity measure. Trigram PRC is a probabilistic topic-based model for retrieving and ranking

[2] http://ii.nlm.nih.gov/MTI/trigram.shtml

related documents with respect to the target document. These three components work independently and in parallel to suggest separate lists of MeSH candidates which are merged in the final stage. Our search-based systems (which will be described in Section 6.4.2) differ from MTI in that we used MetaMap and information retrieval techniques in a sequential way.

Jimeno-Yepes et al. [52] analyzed the MeSH recommended by MTI and studied a few issues of using machine learning approaches for MeSH suggestion. Their work gives useful insights for improving our LLDA-based system.

Huang et al. [50] formulated the indexing task as a ranking problem. In particular, they used a learning-to-rank algorithm to rank MeSH main headings that were extracted from 20 neighbor documents of the target document. Our search-based approach differs from theirs in that we proposed different query formulation strategies and MeSH candidate ranking methods. We also explored the impact of system parameters on the performance.

6.3 Data and Task

The dataset provided by BioASQ challenge contains over 10 million journal articles, each of which consists of the title, abstract, PubMed identifier (PMID), and gold standard MeSH labels that are manually annotated by experts. BioASQ releases 18 test sets of different sizes (ranging from hundreds to tens of thousands documents) over 18 week. Each set consists of new journal articles (<title, abstract, PMID> triples) that have not been annotated or indexed into the PubMed database. The task is to develop systems that can automatically suggest MeSH terms to the unlabeled articles.

We remove duplicated articles that have same PMID in the BioASQ dataset and obtain a pool of 10,699,707 articles with unique PMID[3]. Furthermore, we randomly

[3] Note that we will not distinguish between the singular and plural forms of acronyms (such as CUI and DUI) in this work, i.e., PMID can either stands for PubMed identifier or identifiers depending on the context.

sample 2,000 articles from this pool as a training set for system parameter tuning, and another 2000 random articles as the testing set, as shown in Table 6.1.

Table 6.1: Data

Data	# of articles	Purpose
TRN-0	10,691,707	Training data from BioASQ
TRN-1	10,687,707	Subset of TN-0 used for finding similar articles to the target article
TRN-2	2,000	Subset of TN-0 used for optimizing system parameters
TET	2,000	Subset of TN-0 used for evaluation

6.4 Systems

6.4.1 MetaMap-based Labeling

Concept Detection We process an article by MetaMap while restricting the resource of MetaMap to the MeSH ontology. We obtain and store the following information: 1) concepts (denoted as K) which are phrases or terms that map to UMLS CUI; 2) the list L of MetaMap generated CUI candidates c with confidence scores S_c for each K; and 3) the negation information for each K.

Figure 6.1 gives a concrete example in which "cervical cancer" is a detected, non-negated phrase concept (i.e., K) with a MeSH-related CUI candidates list L (C0007847, C0302592, C0006826, C0998265, etc.). Each c in L has its individual confidence scores S_c, e.g., "C0006826" has a confidence score of 861.

Concept Weighting We first select all non-negated CUI whose confidence scores are above the threshold h. Then, we merge and rank these selected c by aggregating their weighted confidence scores. Here we use superscripts T and A to denote title and abstract respectively. The final ranking score of a specific c looks like:

$$\text{score}(c) = \alpha \sum_{L \in T} S_c^L + \beta \sum_{L \in A} S_c^L, \tag{6.1}$$

where α and β are the weights assigned to c in abstract and title respectively, L is the candidate list for each detected concept K, S_c^L is the confidence score of c in list L. If L does not contain c, S_c^L will be zero.

```
{ "candidates": [
                { "cui": "C0007847",
                    "name": "cervical cancer",
                    "preferredname": "Malignant tumor of cervix",
                    "score": 1000                          },
                { "cui": "C0302592",
                    "name": "CERVICAL CANCER",
                    "preferredname": "Cervix carcinoma",
                    "score": 1000                          },
                          ...                              },
                { "cui": "C0998265",
                    "name": "Cancer",
                    "preferredname": "Cancer Genus",
                    "score": 861                           },
                          ...
                 ],
            "neg": false,
            "phrase": "cervical cancer"                              }
```


Figure 6.1: CUI candidates for a detected concept by MetaMap, shown as a JSON object.

In particular, we fix β to 1.0. However, we vary α (i.e., the weights of c^T) to explore the optimal value of α. We use Equation 6.1 to rank c and select the top-ranked m ones. Finally, we convert the selected c to MeSH Descriptor Unique Identifiers (DUI).

The above method has three free parameters, i.e., h, α, and m. We set their values by exploring the parameter space as will be described in Section 6.5.1.

6.4.2 Search-based Labeling

We describe another approach for MeSH suggestion which is based on information retrieval techniques. This approach starts by finding related articles to the target article, and then leverages their existing annotations to suggest MeSH candidates for the target article.

We use the open-source search engine Indri[4] [104] to build an index for the training set TRN-1. In particular, we remove stop words in the title and abstract based on a medical stoplist [44] and stem words by the Porter stemmer.

There are three components in our retrieval system: 1) the retrieval model for ranking documents; 2) the query generation module which formulates a query based on the target article; and 3) MeSH aggregation module that aggregates and scores the existing annotations for labeling the target article. Next, we will describe each component in detail.

Retrieval Model

Our retrieval model computes the relevance score of a document based on the following function:

$$\text{score}(Q, D) = \sum_{q_i \in Q} w_i f(q_i, D), \tag{6.2}$$

where w_i is the weight associated with a matched feature q_i, and $f(q_i, D)$ is the feature matching function defined as:

$$f(q_i, D) = \log \frac{\text{tf}_{q_i,D} + \mu \frac{\text{tf}_{q_i,C}}{|C|}}{|D| + \mu}, \tag{6.3}$$

where q_i is the ith query term used for text matching. Note that q_i can be either a single word or a phrase. $|D|$ and $|C|$ are the document and collection lengths in words respectively, $\text{tf}_{q_i,D}$ and $\text{tf}_{q_i,C}$ are the document and collection term frequencies of q_i respectively, and μ is the Dirichlet smoothing parameter. Smoothing is a common technique for estimating the probability of unseen words in the documents [23, 124].

The above matching function assigns a score to each match of a query term q, and Equation 6.2 aggregates the scores based on weight w to obtain the final document relevance score. We implement this retrieval model in Indri by formulating queries that look like: #weight(w_0 q_0 w_1 q_1 ...w_i q_i...).

[4] http://www.lemurproject.org/indri/

Query Formulation

Our next step is to formulate a query Q that can be representative of the content of the article. We will describe how we generate query terms q as well as their weights w for the ranking function shown by Equation 6.2.

Term Query (TQ) The first type of queries is based on single words/terms in the article, i.e., terms in a term query are all single-word expressions. In particular, we formulate Q based on words occurring in the concepts detected by MetaMap in both title and abstract, i.e., query terms q come from words in K^T and K^A. Similar to what we have described in Section 6.4.1, we assign equal weight 1.0 to all q^A (i.e., query terms from K^A), but use a varying weight γ for all q^T. A term query in Indri looks like:

```
#weight(2.0 examination 2.0 cow 2.0 ultrasonographic 3.0 navel
3.0 urachal 3.0 extra-abdominal 2.0 pathologic 2.0 abscess)
```

Phrase Query (PQ) The second type of queries are from K^T and K^A directly, i.e., we use concepts (usually phrases) as query terms q_i. Again, we assign equal weight 1.0 to all q^A (i.e., K^A), but use a varying weight γ for all q^T (i.e., K^T). The following shows an Indri phrase query example:

```
#weight(3.5 #uw2(hiv-1 infection) 4.5 #uw2(differential
susceptibility) 2.0 #uw2(actin dynamics) 2.0 actin
4.5 #uw2(cortical actin) 4.5 #uw3(naive t cells)
2.5 dichotomy 3.5 #uw2(human memory)
3.5 #uw3(chemotactic actin activity) 2.0 cd45ro)
```

"#uwN(t1 t2)" means words t1 and t2 can be in any order within a text window of N words, and thus it takes possible variants of a phrase into consideration.

Long Query (LQ) The term query considers single words only and ignores the term proximity information in concepts. Thus, it may hurt retrieval precision. On the other hand, the phrase query poses "stricter" matching criteria, i.e., if a relevant document does not have an exact match for a concept phrase K (e.g., for "#uw2(hearing loss)" to match "loss of hearing"), it will not get any credit by Equation 6.3. Therefore, we

formulate a long query that consists of qi from both TQ and PQ, i.e., both single word query terms and phrase query terms.

To prevent Q from being too long (computationally expensive when retrieving against a large database) for the above three types of queries (i.e., TQ, PQ, and LQ), we remove q that occur only once in the the abstract and title combined unless no q occur more than once (which is a very rare case).

Result Aggregation

For each target article, we formulate query Q and rank documents based on Equations 6.2 and 6.3. Then, we take the top-ranked k documents, weight their existing MeSH annotations (i.e., DUI) by their individual relevance scores shown in Equation 6.2, and aggregate the wegiths for each DUI. Finally, we select the top-ranked m DUI as MeSH annotations for the target article.

In our Search-based Labeling method, we will also allow three free parameters: μ (the Dirichlet parameter in Equation 6.3), k (the number of top-ranked documents used for DUI aggregation) and m (the number of DUI). We will discuss how to set these parameter in Section 6.5.1.

6.4.3 LLDA-based Labeling

The MeSH indexing can also be cast as a multi-labeled classification task. Therefore, the labeled latent Dirichlet allocation (LLDA) [91], a supervised variation of the unsupervised LDA used for credit attribution in multi-labeled corpora, fits well to this MeSH indexing task.

In LDA, each document may be viewed as a mixture of various topics, and the topic distribution has a Dirichlet prior. As an extension of LDA, LLDA further incorporates observed label information, and thus can generate topics that predict labels. Therefore, we train an LLDA model with a subset ($\sim$15%) of set of TRN-0 (see Table 6.1) and use the existing MeSH annotations as labels.

However, the MeSH ontology contains too many labels (over 25,000 descriptors) for our LLDA to handle. Therefore, we only use the 12 MeSH terms at the category level (i.e., children of the root) to form our label set. MeSH annotations of articles are all converted to their corresponding ancestors in this category-level set.

Given a target article, our LLDA will predict its category level labels which will be further used to filter irrelevant labels assigned by previous MetaMap-based or search-based systems. Our goal is to remove false positives and improve precision.

6.5 Evaluation

BioASQ evaluates the MeSH annotation results by two different groups of metrics, i.e., the flat and hierarchical precision (P), recall (R), and F-1 (F) measures, among which Micro F-measure (MiF) and Lowest Common Ancestor F-measure (LCA-F) are the primary evaluation metrics. Thus, we will report P/R/F for both Microaveraging and LCA measures, i.e., (MiP, MiR, MiF) and (LCA-P, LCA-R, LCA-F).

We have five systems, namely the MetaMap-based system (MM), Search-based systems (TQ, PQ, and LQ), and the LLDA-based system (LLDA). Note that for convenience in the rest of work we will refer to each system by their short names given in parentheses.

6.5.1 Parameter Exploration

As mentioned in Section 6.3, we use set TRN-2 to train system parameters. In this section, we show how each free parameter affects performance.

MetaMap-based Labeling

System MM has three free parameters, i.e., h (title concept weight), α (confidence score threshold for CUI candidates), and m (number of DUI in the final suggested list). To get the best setting for MM, we explore the range (400, 1000, 100) for h, (0, 5.0, 0.5) for α, and (8, 41, 4) for m, and try all different value combinations. Note that the third element is the range is the step size.

Table 6.2a shows that MM achieves the best MiF score (0.2697) when $w = 4.5$, $h = 600$, and $m = 12$ (the best setting). To explore the impact of each free parameter on the performance, we fix two of them based on the best setting, vary the left one, and obtain the performance curves as shown in the left column of Figure 6.2.

In particular, the performance curve in Figure 6.2a, where we vary the weight of title concepts, shows that we should assign higher weights to the title concepts. This is expected because the title of an article usually contains the most representative information and the concepts in title are very likely to associate with MeSH annotations.

In Figure 6.2c, as we lower the confidence score threshold for MetaMap CUI candidates from 1000 to 700, the precision declines while the recall improves. However, the precision bounces back when h is below 700, and the best performance for MiF, MiP, and MiR all appears at 600.

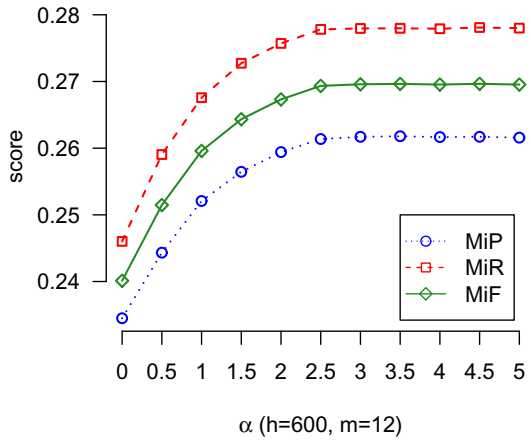
In Figure 6.2e, the precision decreases and the recall increases, both monotonically, as we increase m, the number of DUI for annotating an article. This is also expected because DUI ranked lower down the list are less likely to be correct annotations, and consequently hurt the precision but improve the recall.

Search-based Labeling

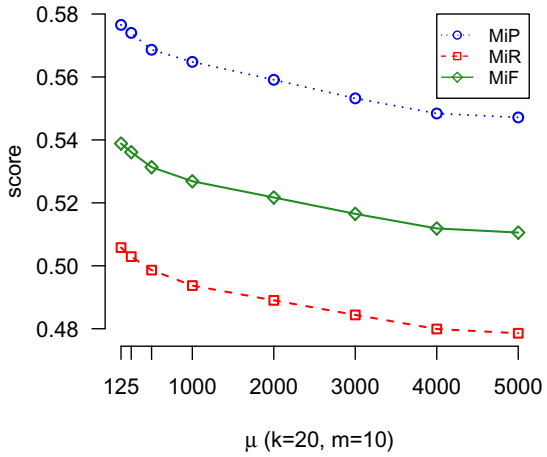
Now we explore the parameter setting for search-based systems, which also have three free parameters: μ (the Dirichlet parameter in Equation 6.3), k (the number of top-ranked documents used for DUI aggregation) and m (the number of DUI). In particular, we will train system TQ and use it as a reference for setting corresponding parameters in PQ and LQ.

Table 6.2a shows that TQ achieves the best MiF score (0.5389) when $\mu = 125$, $k = 20$, and $m = 12$ (the best setting). Again, to explore the impact of each free parameter on the performance, we fix two of them based on the best setting, vary the left one, and obtain the performance curves as shown in the right column of Figure 6.2.

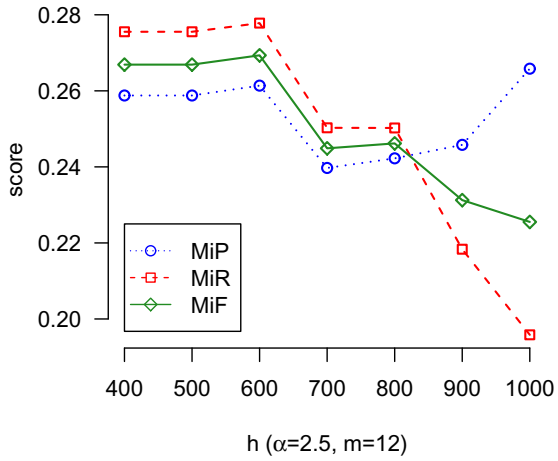
In Figure 6.2b, the performance degrades as we increase μ (i.e., more smoothing



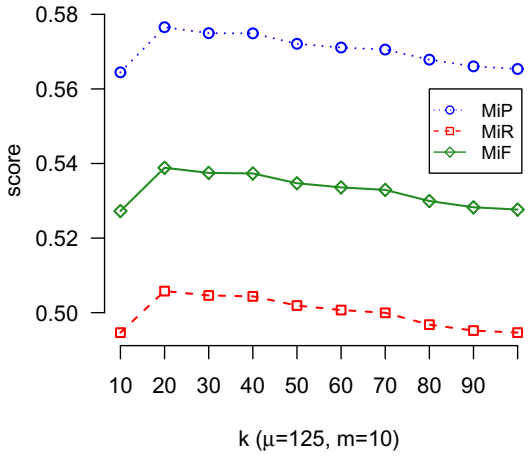
(a) Meta: Title concept weight



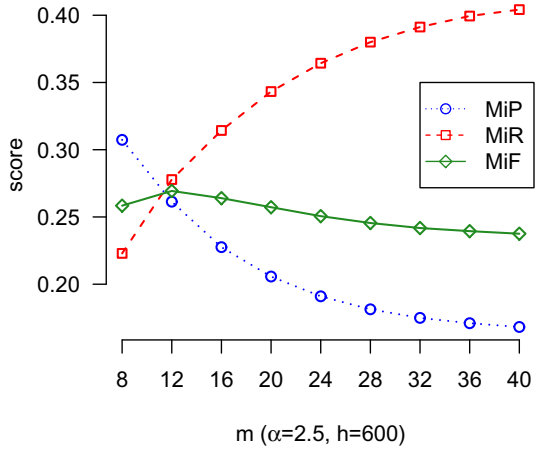
(b) Search: Dirichlet parameter



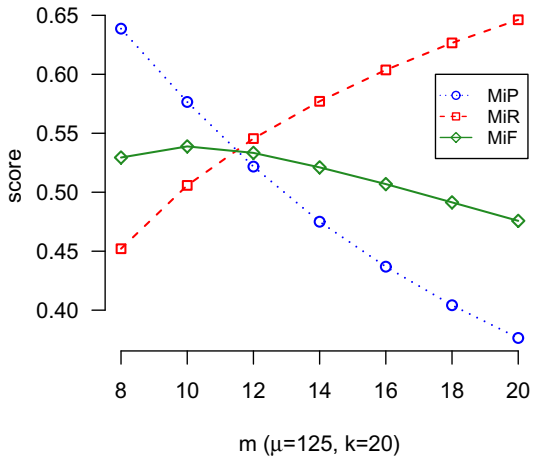
(c) Meta: Confidence score threshold



(d) Search: Top documents



(e) Meta: Cutoff of MeSH candidate list



(f) Search: Cutoff of MeSH candidate list

Figure 6.2: Parameter setting for MetaMap-based and Search-based labeling methods

with the collection-level statistic). This may be because our search-based labeling uses the top-ranked documents for MeSH suggestion and it desires a document set that has a high precision, and on the other hand, less smoothing makes sure that the relevant information remain highly concentrated in these documents which consequently appear among the top of the rank list.

In Figure 6.2d, as we increase the number of top-ranked documents the performance peaks early at $k = 20$ and declines after that point, which is expected because our search-based labeling desires only a few highly relevant documents that can provide a more reliable set of MeSH candidates.

Figure 6.2f looks very similar to Figure 6.2e in which the system tries to strike a balance between precision and recall by varying m. However, method TQ only needs top 10 candidates to achieve the best MiF, as opposed to top 12 in MetaMap-based labeling, indicating that our search-based method is more precision-focused.

Due to the high similarity among search-based systems, we will simply use the best parameter setting of TQ ($\mu = 125$, $k = 20$, and $m = 12$) for systems PQ and LQ in our testing stage which is presented next.

6.5.2 Test and Comparison

Table 6.2 shows the evaluation results. System MM and TQ both obtain comparable test results to those on the training set, indicating that our parameter setting process results in consistent performance.

System TQ, as the simplest among search-based systems, achieves the best performance. However, systems PQ and LQ are doing worse than TQ. The reason might be that the simple term frequency based phrase weighting strategy could not well distinguish important concepts from unimportant ones, and consequently hurts the precision.

In System LLDA, we use the predicted category labels to prune the annotation list from system TQ. We start with a long candidate list by setting m to 20, and then prune this list with LLDA. Table 6.2b shows that LLDA does not produce positive

Table 6.2: Evaluation

(a) Training

System	MiP	MiR	**MiF**	LCA-P	LCA-R	**LCA-F**
MM ($w = 4.5$, $h = 600$, $m = 12$)	0.2617	0.2781	0.2697	0.3303	0.2831	0.2931
TQ ($\mu = 125$, $k = 20$, $m = 12$)	0.5766	0.5058	0.5389	0.4143	0.4655	0.3978

(b) Testing

System	MiP	MiR	**MiF**	LCA-P	LCA-R	**LCA-F**
MM ($w = 4.5$, $h = 600$, $m = 12$)	0.2660	0.2780	0.2719	0.3322	0.2862	0.2963
TQ ($\mu = 125$, $k = 20$, $m = 12$)	0.5842	0.5044	*0.5413*	0.4697	0.3979	*0.4168*
PQ (same setting as TQ)	0.5141	0.4389	0.4735	0.4257	0.3496	0.3710
LQ (same setting as TQ)	0.5748	0.4953	0.5321	0.4638	0.3918	0.4110
LLDA ($\mu = 125$, $k = 20$, $m = 20$)	0.5843	0.4400	0.5017	0.3322	0.2842	0.2950

(c) Comparison

BioASQ System	MiP	MiR	**MiF**	LCA-P	LCA-R	**LCA-F**
MTIFL Baseline	0.602	0.513	0.554	0.455	0.550	0.475
MeSH Indexing	0.425	0.598	0.497	0.531	0.470	0.473
system3	0.444	0.521	0.480	0.430	0.475	0.432
mc3	0.515	0.433	0.470	0.479	0.402	0.416
BioASQ Baseline	0.258	0.285	0.271	0.389	0.330	0.334

results. This might be because the category level MeSH terms are broad concepts that are not discriminative enough to distinguish one from another.

We compare our system with other top performing BioASQ systems using a set of 1942 new journal articles. Table 6.2c shows the evaluation results.

There are two baselines systems. The "MTIFL Baseline" is the state-of-the-art system developed and used at NLM. Thus, it is a very strong baseline. On the contrary, the "BioASQ_Baseline" is a weak baseline since it follows an unsupervised approach [86].

The "MeSH Indexing" system [77] is developed by the NCBI (National Center for Biotechnology Information). It first finds the k-nearest neighbors of the test article and then aggregate and rank the existing labels of these neighbors based on a learn-to-rank framework. The features used for learning include unigram/bigram overlap features, neighborhood features, results from "MTI Baseline", etc.

"system3" [108] also took a learning approach. In particular, the system learns

two models: one for ranking the labels according their relevance to the test article and another for predicting the number of label related to the test article.

Our system "mc3" (corresponding to system "TQ") achieved comparable performance with other top-ranked systems. In particular, our system seems to be more precision-focused as indicated by the MiP. This might be because that the candidate labels come from the most relevant articles whole existing labels have a lot overlap with each other.

6.6 Conclusion and Future Work

In this chapter, we have proposed three approaches for automatic MeSH term suggestion: 1) MetaMap-based labeling, which relies on the MetaMap tool to detect MeSH-related concepts for indexing; 2) Search-based labeling, which builds upon MetaMap-based approach and further leverages information retrieval techniques for finding similar articles with existing annotations and uses them for MeSH suggestion; 3) LLDA-based labeling, which further builds on Search-based labeling and trains a multi-label classifier based on MeSH ontology for MeSH candidate list pruning.

Our evaluation on the BioASQ challenge data showed promising results for the Search-based labeling. In addition, we explored the impact of different system parameters (e.g., the weight for title concepts, CUI confidence scores, Dirichlet prior, number of top-ranked documents, etc.) on the system performance. In particular, words in the title are more important than words in the abstract. We also proposed a new multi-label classification system based on LLDA for MeSH candidate list pruning.

Although the machine learning based system generally outperformed our system, we believe the research findings presented in this chapter would be useful for designing similar systems for biomedical semantic indexing. In particular, we can incorporate into a machine learning based model (e.g., learning-to-rank) the query features (such as terms and phrases) and the term and label statistics from similar documents found by our retrieval model.

For future work, we plan to explore better concept weighting strategies (e.g., by incorporating corpus-level statistics or using information from external sources) for systems PQ and LQ. As for the LLDA-based labeling, we will extend LLDA model by leveraging hierarchical information in MeSH ontology. In addition, we plan to compare our approaches with existing methodologies and carry out a thorough error analysis to look for aspects that we can further improve.

Chapter 7

GENE ONTOLOGY ANNOTATION

In this chapter, we will investigate how to use information retrieval techniques and domain knowledge to assist gene ontology annotation for biomedical articles.

7.1 Background

The Gene Ontology (GO) provides a set of concepts for annotating functional descriptions of genes and proteins in biomedical literature. The resulting annotated databases are useful for large-scale analysis of gene products. However, performing GO annotation requires expertise from well-trained human curators. Owing to the fast expansion of biomedical data, GO annotation becomes extremely labor-intensive and costly. Thus, text mining tools that can assist GO annotation and reduce human effort are highly desired [71, 110, 18].

To promote research and tool development for assisting GO curation from biomedical literature, the Critical Assessment of Information Extraction in Biology (BioCreative) IV organized Gene Ontology Curation task (GO task) in 2013 [78]. There are two subtasks: A) identification of GO evidence sentences (GOES) for relevant genes in full-text articles and B) prediction of GO terms for relevant genes in full-text articles. The training set of GO task contains 100 full-text journal articles in BioC format [13], while the development and test sets each have 50 articles. Task organizers also provided ground truth annotations for the training and development sets to all participants [13]. Table 7.1 gives the detailed statistics about genes, gene-related passages and GO terms in the GO task data.

The following shows two sample passages and the corresponding key information in the training and development sets:

Table 7.1: Corpus statistics of BioCreative IV Track 4 GO Task.

GO task data	Training	Development	Test
# of full-text articles	100	50	50
# of genes	300	171	194
# of gene-associated passages	2234	1247	1681
# of GO terms	954	575	644

Key information for sample passage 1:

```
-------------------------------------------------------------------------
<infon key="gene">cdc-14(173945)</infon>
<infon key="go-term">embryo development ending in birth or
egg hatching|GO:0009792</infon>
<infon key="goevidence>IMP</infon>
<text>However, of all components tested, only the depletion of
the C. elegans homologue of the budding yeast Cdc14p phosphatase
caused embryonic lethality in the offspring of injected worms
(Table 1).</text>
-------------------------------------------------------------------------
```

Key information for sample passage 2:

```
-------------------------------------------------------------------------
<infon key="gene">cdc-14(173945)</infon>
<infon key="go-term">phosphatase activity|GO:0016791</infon>
<infon key="goevidence>NONE</infon>
<text>text>CeCDC-14 is a phosphatase and localizes to the
central spindle and the midbody</text>
-------------------------------------------------------------------------
```

Given a set of relevant genes, for subtask A, we need to find GOES, while for subtask B, we need to assign GO terms to each article (primarily based on the gene-related sentences identified in subtask A).

In this chapter, we will introduce several systems for the GO task. For subtask A, we train a logistic regression (LR) model to detect GOES using the training data supplemented with noisy negatives from an external resource. A greedy approach is applied to associate relevant genes with sentences. For subtask B, we designed two types of systems: (i) search-based systems, which predict GO terms based on existing

annotations for GOES that are of different textual granularities (i.e., full-text articles, abstracts, and sentences) using state-of-the-art information retrieval techniques and (ii) a similarity-based system, which assigns GO terms based on the distance between words in sentences and GO terms/synonyms.

In the following sections, we will first describe our systems in more detail. Then, we will present and discuss the official evaluation results, and finally draw the conclusion.

7.2 Systems

7.2.1 Subtask A – GOES Identification

In subtask A, given a full-text article, we need to identify GOES and associate them with genes. As we will see, we approach this problem by supervised machine learning. In particular, we consider GOES as positive instances and all other sentences as negative instances. Since the training set is very small, to prevent model overfitting we expand the negatives with unlabeled excerpts from GeneRIF [3] records, which is also based on the concept of distant supervision, i.e. use existing resources to obtain weakly labeled instances for training machine learning classifiers [24, 69].

7.2.1.1 Data Preprocessing

We extract positive and negative instances (i.e. sentences) from both training and developing sets to train our model. The training set contains 1,318 positive and 26,868 negative instances, while the development set gives 558 positive and 14,580 negative sentences.

We use GeneRIF as an unlabeled data pool, which contains excerpts from literature about the functional annotation of genes described in EntrezGene. In particular, each record contains a taxonomy ID, a Gene ID, a PMID, and a GeneRIF text excerpt extracted from literature. We randomly sample 20,000 excerpts from human GeneRIF records and make sure that 1) there are at most two records per Gene ID, and 2) the corresponding articles are not associated with any GO annotation (GOA) record

based on GOA information available in iProClass [115]. We consider these sampled excerpts as negative instances since if they are evidence excerpts the corresponding articles would most probably haven been already included[1] in GOA.

7.2.1.2 Feature Extraction

Our classifier uses the following features:

Bag-of-words (BOW) feature: for each sentence we generate a vector of stemmed words using the Porter stemmer.

Bigram features: for each sentence we generate a vector of bigrams by concatenating every two neighboring stemmed words in the sentence. We also have two boundary bigrams (SOS_Lw and Rw_EOS) where SOS indicates "Start of the Sentences", EOS means "End of the Sentence", Lw is the leftmost stemmed word, and Rw is the rightmost stemmed word.

Section feature: For each sentence, we include a feature to indicate which section the sentence is from, i.e., title, abstract, introduction, methods, discussion, etc.

Topic feature: These features are generated by Latent Dirichlet Allocations [19], which can effectively group similar instances together based on their topics [61].

Gene presence features: Because relevant genes of each article have been provided, we also use dictionary lookup to check the presence of relevant genes in the sentence.

7.2.1.3 Model Training

We apply logistic regression (LR) to predict labels for each instance. In particular, we impose a constraint on model parameters in a regularized LR to avoid overfitting and improve the prediction performance on unseen instances. Note that our LR will

[1] The rationale behind this assumption is that the scope of the functional annotation in GeneRIF is broader than that of GO. Besides the scope of GO annotation, GeneRIF also includes phenotypic and disease information that are not the subject of GO annotation. Note that this assumption does not guarantee all excerpts obtained to be true negatives

assign probability scores to each class. In a task with skewed class distribution, a threshold can be chosen to optimize the performance.

For each article, all relevant genes are provided. Therefore, we use a greedy approach to associate evidence excerpts with the relevant genes in four steps:

1. *Direct matching with dictionary lookup.* Direct dictionary lookup is done for each predicted positive sentence to detect whether there are relevant genes appearing in the 40 sentence. If so, the corresponding genes found are assigned to that sentence

2. *Family name inferred.* Because genes belonging to the same family can appear as plurals in the document, we assemble a dictionary of family names based on the gene mentions provided. For each mention of the family name in a sentence (using direct string matching), all of the members of that family in the gene list are assigned to the sentence.

3. *Gene assignment based on proximity.* For the remaining predicted positive sentences with no relevant gene mentioned, we assume that prior sentences would contain the gene information. For positive sentence S, we perform direct string matching using the gene list provided and the family name dictionary assembled in Step 2 on all prior sentences belonging to the same section of S. Gene hits are identified similarly as in Steps 1 and 2. We then assign gene hits from the closest one (among all prior sentences with gene hits) to S.

4. *Assignment based on gene-sentence distributions.* For genes that fail to be associated with any predicted positive sentence, we picked sentences containing the corresponding genes with the largest positive probability score (assigned by the LR model) to be the evidence sentences.

7.2.1.4 Experimental Setup

We used LR-TRIRLS [4], which implements ridge regression, to build LR models. We chose a threshold of 0.1 based on the performance of the model trained using the training set and evaluated using the development set, where if a sentence has a probability >0.1 to be positive, then we consider it as positive. We submitted three runs A1, A2 and A3 for subtask A. Runs A1 and A2 used different sets of unlabeled instances sampled from GeneRIF, and Run A3 combined the results from A1 and A2.

7.2.2 Subtask B – GO Terms Prediction

In this section, we describe two different systems for GO term prediction. The basic idea is to leverage existing GOA to label new articles. In particular, we search for relevant documents (sentences, abstracts or full-text articles) that have existing GOA to the target article, and then score and aggregate these existing GOA to produce the GOA for the target article.

7.2.2.1 System B1

Figure 7.1 gives an overview of system B1 with external resources highlighted by blue color and system modules by gray. Next, we describe each component in more details.

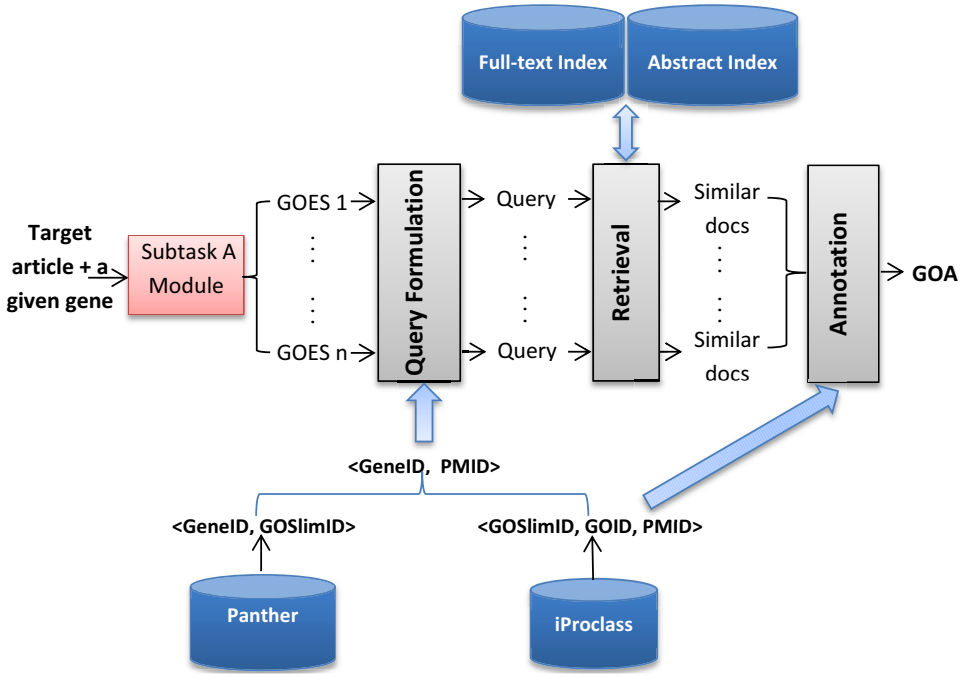


Figure 7.1: Overview of system B1.

Resources

We use the following external resources: 1) Panther [105], from which we build <GeneID, GOSlimID> pairs, 2) iProClass [115], from which we obtain <GOSlimID, GOID, PMID> triplets, 3) a collection of PMC full-text articles that serve as the

source for finding relevant documents, and 4) a collection of PubMed abstracts, used as a complementary source for retrieving, because for some GOA records, only abstracts are publicly available for the corresponding articles.

Retrieval

We build indexes for the abstract collection and the full-text collection, respectively, using the Indri (17) search engine. In particular, we use the Porter stemmer for stemming words in the documents. We choose the query likelihood language model as our retrieval model. This model scores documents for queries as a function of the probability that query terms would be sampled (independently) from a bag containing all the words in that document. Formally, the scoring function is a sum of the logarithms of smoothed probabilities:

$$\text{score}(D, Q) = \log P(Q|D) = \sum_{i=1}^{n} \log \frac{\text{tf}_{q_i,D} + \mu \frac{\text{tf}_{q_i,C}}{|C|}}{|D| + \mu}, \tag{7.1}$$

where q_i is the ith term in query Q, n is the total number of terms in Q, $|D|$ and $|C|$ are the document and collection lengths in words respectively, $\text{tf}_{q_i,D}$ and $\text{tf}_{q_i,C}$ are the document and collection term frequencies of q_i respectively, and μ is the Dirichlet smoothing parameter.

Query Formulation

We formulate a query for each detected GOES from the output of subtask A. In particular, we filter stop words in the sentence using a standard stop word list. We leverage information in <GeneID, GOSLIM, GO> triples to reduce the GO candidate list (denoted as C), and then build a PMID candidate list by incorporating information in the <PMID, GOA> pairs. The following are the detailed steps:

1. Given a gene G, we have a list of <G, GOES> pairs.
2. For each <G, GOES> pair, we find the corresponding <G, GOSlimID> pairs.
3. For each <G, GOSlimID> pair, we get a list of PMIDs based on <GOSlimID, GOID, PMID> triplets.
4. Combine all PMIDs for G to get a <G, L> pair, where L is the PMID candidate list (i.e., a reduced search list) for G.

Annotation The output from the retrieval model for a given <GeneID, GOES> pair is a list of documents ranked by their relevance scores. Based on the <GOSlimID, GOID, PMID> triplets, we obtain GOIDs for top-ranked k documents, and then weight each GOID by their corresponding document relevance score. We further aggregate scores of each GOID and take the top-ranked m GOID for each GOES. Finally, we combine GOID across all GOES, rank them according to their occurrences and keep GOID, which occurs more than p times. We set <k, m, p> to <7, 10, 4> by training them on the 150 articles (i.e. the combination of training and development sets).

7.2.2.2 System B2

Figure 7.2 gives an overview of System B2 which has similar modules to system B1. The major difference is that we use GeneRIF as the external resource. In particular, we extract <Sentence, GOID> pairs from GeneRIF where the corresponding articles are cited as evidence of GOA records in iProClass and build an index for this collection of sentences. Therefore, the output from the retrieval model is a ranked list of sentences, which are further converted to a ranked list of GOID based on <Sentence, GOID> pairs. Finally, in the Annotation module we do the following:

1. Starting from an initial list that contains top-ranked k GOID, select GOID one by one down the list until the score difference of current GOID with the topmost GOID is above threshold h.
2. Aggregate GOID frequency across all GOES associated with a particular gene, and rank GOID by frequency.
3. Take the top-ranked m GOID for each gene.

7.2.2.3 Baseline System B3

We use a greedy string matching algorithm to generate the baseline. Specifically, we obtained all words in the sentences that are aligned to GO terms and synonyms when ignoring lexical variations. We then computed the Jaccard distance [92] between those matched words with GO terms and synonyms. A threshold of 0.75 was used for GO term assignment.

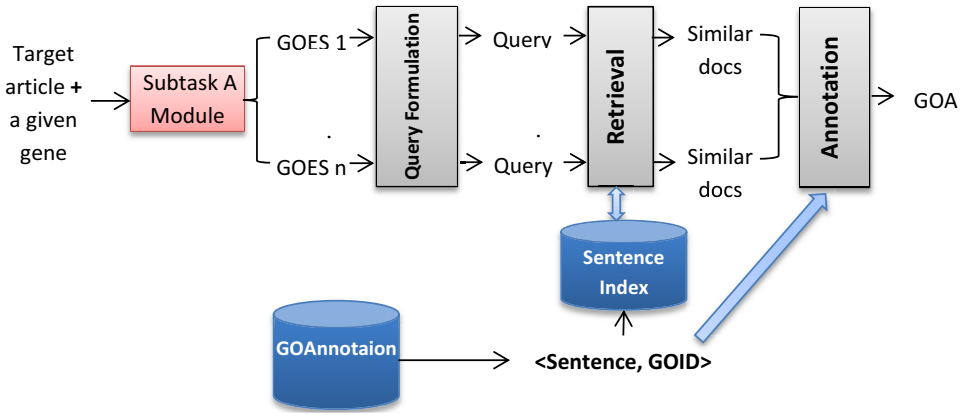


Figure 7.2: Overview of system B2.

7.2.2.4 Experimental Setup

We implement systems B1 and B1 in Indri. By training them on the 150 articles (i.e. the combination of training and development sets), we set <k, m, p> to <7, 10, 4> for system B1 and <k, h, m> to <5, 0.1, 3> for system B2.

7.3 Evaluation

7.3.1 Evaluation Metrics

We use the precision (P), recall (R) and F1-measure (F1) scores to evaluate the results for both subtasks [78]. However, for subtask A there are two different criteria for determining a match between a candidate sentence and the ground truth sentence: 1) exact match between sentence boundaries and 2) partial overlapping. For subtask B there are also two different matching criteria: flat or hierarchical. For the flat metrics, a match occurs when the predicted GO term is exactly the same as the gold standard. For hierarchical metrics, a match occurs when the predicted GO term has a common ancestor with the ground truth GO term.

7.3.2 Results and Discussion

Table 7.2 presents the evaluation results of subtask A. Systems A1 and A3 obtain comparable F1 scores. System A2 has a lower F1 score due to the relatively

low performance on recall. The performance difference between A1 and A2 is caused by the noisy negative training instances sampled randomly from GeneRIF.

Table 7.2: Evaluation results for GOES identification.

System	Overlap match			Exact match		
	P	R	F1	P	R	F1
A1	0.313	0.503	0.386	0.219	0.352	0.270
A2	0.314	0.442	0.367	0.220	0.310	0.257
A3	0.307	0.524	0.387	0.214	0.366	0.270

During the development phase of systems for subtask A, we assessed the performance with or without the use of additional GeneRIF excerpts and the contributions of individual types of features. We found that the use of an unlabeled data set sampled from GeneRIF improved the F1 score by 0.03 compared with the baseline, which uses only positives and negatives from the training data set and BOW features. Also, including other features (bigrams, gene presence, section, and topic features) led to performance improvement over the baseline. In particular, section feature improved the F1 score by 0.01. Bigram and gene presence features each brought an improvement of 0.008. Topic features further added 0.003 when the number of topics was set to 100.

Table 7.3 presents the official evaluation results of subtask B. The exact F1 scores for both types of systems are less than 0.1. System B1 achieves 0.301 for Hierarchical-F1. Our search-based systems (i.e., B1 and B2) outperformed the similarity-based system (i.e., B3) significantly.

Table 7.3: Evaluation results for GO annotation.

System	Flat			Hierarchical		
	P	R	F1	P	R	F1
B1	0.054	0.149	0.079	0.243	0.459	0.318
B2	0.088	0.076	0.082	0.250	0.263	0.256
B3	0.029	0.039	0.033	0.196	0.310	0.240

We were not aware of the need of containing experimental methods for detecting GO evidence excerpts and assigning GO terms as specified by the annotation guideline. This may explain why the use of section features in subtask A has the most gain in

the F1 score. Additionally, we sampled only from human GeneRIF records with at most two records per gene. The rationale behind it is to avoid over-representation of popular studied genes and their homologous genes. It is not clear whether such sampling approach has impact on the performance of the system.

7.3.3 Comparison with Related Work

Table 7.4 shows the performance scores of the best runs from the top 3 teams for subtask A. Team 238 corresponds to our team, which placed 1st.

Team 250 also built a binary classifier but used the reference distance estimator [62] for constructing features from a large number of unlabeled sentences to overcome data sparseness [63].

Team 237 designed a rule-based system by using text mining techniques to extract information inside the GO database.They used more than 63,000 automatically generated rules.

Other teams also used either machine learning based or rule-based approaches. Although the overall results are far from satisfaction for practical use due to the limited high quality training data, the general trend is that machine learning based approaches outperformed the rule-based approaches [78].

Table 7.4: System comparison for subtask A. Systems are ordered by the exact match F1 score.

Team	Run	Genes	Passages	Exact match			Overlap		
				P	R	F1	P	R	F1
238	3 (A3)	194	2866	0.214	0.366	0.270	0.307	0.524	0.387
250	2	140	2848	0.153	0.259	0.193	0.258	0.437	0.325
237	3	171	3717	0.138	0.305	0.190	0.213	0.471	0.293

Table 7.5 shows the performance scores of the best runs from the top 3 teams for subtask B. Again team 238 correspond to our team which ranked the 2nd. However, we are the only team that suggested GO terms for all the 194 genes provided for subtask B.

Team 183 used the supervised categorization method which retrieved most prevalent GO terms among the 5 most similar instances to the input text in their knowledge base [37].

Team 250 also developed an IR-based system for subtask B, but their approach is quite different. In particular, they used GOES as queries and retrieve the relevant GO terms using a ranking function that combined cosine similarity and GO term frequency in documents. They achieved comparable F1 scores to our system although they only provided GO terms for only about 2/3 of the total number of genes in the articles.

Although the IR-based approaches are not as good as the machine learning approach, it might be because that the IR-based approach is more resource-dependent, i.e., it cannot deal with unseen instances well. With enough data, the IR models could be able to score the documents better and consequently provide a better suggestion list of GO term.

Table 7.5: System comparison for subtask B. Systems are ordered by the exact match F1 score.

Team	Run	Genes	GO terms	Exact match			Hierarchical match		
				P	R	F1	hP	hR	hF1
183	1	172	860	0.117	0.157	0.134	0.322	0.356	0.338
238	2 (B2)	194	555	0.088	0.076	0.082	0.250	0.263	0.256
250	3	132	453	0.095	0.067	0.078	0.284	0.161	0.206

7.4 Conclusion

We investigated the use of distant supervision for detecting sentences for GO annotation assignment and explored using information retrieval techniques for finding relevant existing GOA to predict GO terms to new articles. The results look promising compared with other systems performing the same task.

In particular, we had several interesting findings: 1) constructing weakly labeled instances based on distance supervision is helpful when the training data size is small; 2) the GO Slim, as a cut-down version of GO ontologies, further mitigate the issue of limited training data by allowing us to focus on the high-level GO terms instead of

fine-grained ones; 3) useful features for predicting the GO evidence sentences include bag-of-words, bigrams, sections, topics, and gene presence; 4) useful resources for query formulation to find similar articles/sentences include <GeneID, GOSlimID> pairs in Panther [105], <GOSlimID, GOID, PMID> triplets in iProClass [115], <Sentence, GOID> from GeneRIF [3], as well as PMC full-text articles and PubMed abstracts.

Chapter 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

In this section, we conclude the thesis by summarizing the main contributions:

We have built and evaluated several effective systems to reduce the manual work for three different clinical and biomedical tasks, namely EMR-based cohort identification, MeSH indexing, and gene ontology annotation. They all achieved competitive results compared with other systems performing the same task. In particular, we obtained the highest evaluation scores on the 2011 & 2012 Medical Records Track datasets [127, 128, 129] as well as the 2013 ShARe/CLEF eHealth Task 3 dataset [136].

For the EMR-based cohort identification, we explored three directions for improving the retrieval performance:

1) we specifically designed methods for aggregating the multi-level evidence in the EMR. At the field level, we explored features such as ICD, NEG, and AGF to expand evidence and remove extraneous information. At the report level, we introduced RbM (Retrieval-before-Merging) and experimented with SUM, MAX, and ANZ as the merging strategy. At the visit level, we introduced MbR (Merging-before-Retrieval) method which merges reports from a visit field by field into a single visit document and then performs retrieval against an index visits. Finally at the top level, we introduced VRM (Visit-Ranking-Merging) and compared CombSUM/MNZ/MAX/ANZ with a query-adaptive merging scoring scheme.

We also studied features that are useful for predicting the combination weight of the query-adaptive VRM [131]. The most effective features capture either the ambiguity of individual query concepts or the semantic similarity between those concepts.

All the evidence aggregation strategies discussed in Chapter 3 contribute to system performance. In particular, ICD, AGF, and MbR focus on improving recall while NEG and RbM on precision. Furthermore, the CombWEG/SUM VRM and the adaptive VRM improve both recall and precision.

2) we have introduced the CME model specifically designed for alleviating polysemy and synonymy related issues in medical IR [130]. In particular, the MRF component disambiguates word senses and improves search precision by incorporating contextual information in the query. On the other hand, the EMRM component enhances recall by deriving query expansion terms from multiple external collections from different domains. CME has shown strong performance and advantage other systems when performing EMR search and medical web document search.

3) we explored how to use domain knowledge improved EMR search. In particular, we have proposed and evaluated a joint search framework for building an EMR search system in which we can flexibly incorporate medical domain knowledge [132]. This framework also allows the system to automatically and adaptively adjust the combination weight for the text-based and concept-based retrievals by using any informative features. Our cross-validation results showed that this adaptive result merging algorithm is more effective than a well-tuned fixed-weight merging algorithm.

We also investigated the usefulness of a large clinical corpus for query expansion in the EMRM model [135]. We have shown more data is not necessarily better for query expansion, implying that there is value in collection curation. We concluded that the size, quality, and diversity of the expansion collections are the three important factors that dictate the effectiveness of the EMRM model.

4) We showed that different biomedical different medical tasks usually need specific design and tailor of the search methods. In particular, we investigated several query formation methods for finding similar articles with existing annotations and used them for predicting annotations for new articles in both the MeSH indexing [133] and the Gene ontology annotation [134] tasks.

Overall, based on what we learnt from this thesis work, here are some key points

we believe that can help improve the retrieval performance in general or in other specific domains: i) contextual information in text space (e.g., modeled by MRF or derived from measuring textual similarity) improves precision as it disambiguates word senses; ii) query expansion using external resources including both free-text and structured collections improves recall as it expands query with many other related terms. More specifically, the size, quality (in terms of content similarity to the target collection), and diversity (in terms of combining multiple in-domain and out-of-domain resources) are key properties for selecting effective expansion collections; iii) structured in-domain data developed by human annotators (e.g., ontologies) allows us to transform both data and retrieval models from the text space to the concept space where discovering and measuring semantic relationships (i.e., contextual information in the concept space) among query concepts becomes feasible so that the precision can be further improved in another way.

8.2 Future Work

For future work, one main direction is to investigate whether it is worthwhile and how to turn the prototype systems described in this thesis into production systems for the EMR search. This will necessarily involve several key things:

1) we need to carefully design a graphical user interface (GUI) for our prototype system so that the users can easily explore its full functionality with minimal intervention. In addition, the GUI should have certain features, such as evidence highlighting and score displaying, to help users quickly identify retrieval false positives.

2) then we need to thoroughly evaluate our prototype system under a real working environment. In particular, we should monitor key parameters that are indicative of the usability and effectiveness of the system, such as the time spent on using the prototype system for completing the search tasks, explicit user feedback, etc. We also need to compare the prototype system with existing systems for EMR search.

3) if the evaluation results are satisfying, we need to consider other practical issues (e.g., scalability) for migrating the prototype system into a production system.

BIBLIOGRAPHY

[1] Apache Lucene. http://lucene.apache.org/.

[2] E-utilities quick start. http://www.ncbi.nlm.nih.gov/books/NBK25500/. Accessed: 2012-09-30.

[3] Generif: gene reference into function. http://www.ncbi.nlm.nih.gov/gene/about-generif. Accessed: 2013-09-12.

[4] LR-TRIRLS: Logistic regression for binary classification. http://komarix.org/ac/lr. Accessed: 2013-09-8.

[5] Trec genomics track. http://ir.ohsu.edu/genomics/. Accessed: 2014-08-18.

[6] Nasreen Abdul-jaleel, James Allan, W. Bruce Croft, O Diaz, Leah Larkey, Xiaoyan Li, Mark D. Smucker, and Courtney Wade. Umass at trec 2004: Novelty and hard. In *In Proceedings of TREC-13*, 2004.

[7] Gianni Amati and Cornelis Joost Van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20(4):357–389, 2002.

[8] Ion Androutsopoulos. A challenge on large-scale biomedical semantic indexing and question answering. In *BioNLP Workshop*, 8 2013.

[9] Alan R. Aronson. Effective mapping of biomedical text to the UMLS metathesaurus: The MetaMap program. *Proceedings of AMIA Symposium*, pages 17–21, 2001.

[10] Alan R Aronson. Metamap: Mapping text to the umls metathesaurus. *Bethesda, MD: NLM, NIH, DHHS*, 2006.

[11] Alan R Aronson and François-Michel Lang. An overview of metamap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236, 2010.

[12] Alan R Aronson, James G Mork, Clifford W Gay, Susanne M Humphrey, and Willie J Rogers. The NLM indexing initiative's medical text indexer. *Medinfo*, 11(Pt 1):268–72, 2004.

[13] KV Auken, ML Schaeffer, P McQuilton, et al. Corpus construction for the biocreative iv go task. *Proceedings of BioCreative IV*, 2013.

[14] Steven Bedrick, Tracy Edinger, Aaron Cohen, and William Hersh. Identifying patients for clinical studies from electronic health records: TREC 2012 medical records track at OHSU. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[15] Michael Bendersky, Donald Metzler, and W Bruce Croft. Learning concept importance using a weighted dependence model. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 31–40. ACM, 2010.

[16] Michael Bendersky, Donald Metzler, and W Bruce Croft. Parameterized concept weighting in verbose queries. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 605–614. ACM, 2011.

[17] Adam Berger and John Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229. ACM, 1999.

[18] Christian Blaschke, Eduardo A Leon, Martin Krallinger, and Alfonso Valencia. Evaluation of biocreative assessment of task 2. *BMC bioinformatics*, 6(Suppl 1):S16, 2005.

[19] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[20] Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl 1):D267–D270, 2004.

[21] Chris Buckley and Ellen M Voorhees. Retrieval evaluation with incomplete information. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 25–32. ACM, 2004.

[22] W W Chapman, W Bridewell, P Hanbury, G F Cooper, and B G Buchanan. Evaluation of negation phrases in narrative clinical reports. *Proceedings of AMIA Symposium*, pages 105–109, January 2001.

[23] Stanley F Chen and Joshua Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318. Association for Computational Linguistics, 1996.

[24] Yang Chen, Manabu Torii, Chang-Tien Lu, and Hongfang Liu. Learning from positive and unlabeled documents for automated detection of alternative splicing sentences in medline abstracts. In *Bioinformatics and Biomedicine Workshops (BIBMW), 2011 IEEE International Conference on*, pages 530–537, 2011.

[25] Sungbin Choi and Jinwook Choi. Snumedinfo at clefehealth2013 task 3. In *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*, 2013.

[26] Bruce Croft, Donald Metzler, and Trevor Strohman. *Search Engines: Information Retrieval in Practice*. Addison Wesley, 1 edition, February 2009.

[27] Steve Cronen-Townsend, Yun Zhou, and W Bruce Croft. Predicting query performance. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM, 2002.

[28] Mariam Daoud, Dawid Kasperowicz, Jun Miao, and Jimmy Huang. York University at TREC 2011: Medical Records Track. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[29] A García Seco de Herrera, Jayashree Kalpathy-Cramer, D Demner Fushman, Sameer Antani, and Henning Müller. Overview of the imageclef 2013 medical tasks. *Working notes of CLEF*, 2013.

[30] Dina Demner-Fushman, Swapna Abhyankar, Antonio Jimeno-Yepes, Russell Loane, Francois Lang, James G Mork, Nicholas Ide, and Alan R Aronson. Nlm at trec 2012 medical records track. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[31] Dina Demner-Fushman, Swapna Abhyankar, Antonio Jimeno-Yepes, Russell Loane, Bastien Rance, François Lang, Nicholas Ide, Emilia Apostolova, and Alan R Aronson. A knowledge-based approach to medical records retrieval. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[32] Joshua C Denny, Jeffrey D Smithers, Randolph A Miller, and Anderson Spickard. "Understanding" medical school curriculum content using KnowledgeMap. *Journal of the American Medical Informatics Association*, 10(4):351–362, 2003.

[33] Fernando Diaz and Donald Metzler. Improving the estimation of relevance models using large external corpora. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 154–161, New York, NY, USA, 2006. ACM.

[34] Carol Friedman, Lyudmila Shagina, Yves Lussier, and George Hripcsak. Automated encoding of clinical documents based on natural language processing. *Journal of the American Medical Informatics Association*, 11(5):392–402, 2004.

[35] Mark E Funk and Carolyn A Reid. Indexing consistency in medline. *Bulletin of the Medical Library Association*, 71(2):176, 1983.

[36] Vijay Garla and Cynthia Brandt. Semantic similarity in the biomedical domain: an evaluation across knowledge sources. *BMC Bioinformatics*, 13:261, 2012.

[37] Julien Gobeill, Emilie Pasche, Dina Vishnyakova, and Patrick Ruch. Bitem/sibtex group proceedings for biocreative iv, track 4. In *Proceedings of the 4th BioCreative Challenge Evaluation Workshop*, 2013.

[38] Lorraine Goeuriot, Gareth JF Jones, Liadh Kelly, Johannes Leveling, Allan Hanbury, Henning Müller, Sanna Salanterä, Hanna Suominen, and Guido Zuccon. Share/clef ehealth evaluation lab 2013, task 3: Information retrieval to address patients' questions when reading clinical reports. In *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*, 2013.

[39] Travis Goodwin, Bryan Rink, Kirk Roberts, Sanda M Harabagiu, and Richardson Tx. Cohort shepherd: Discovering cohort traits from hospital visits. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[40] Hussam Hamdan, Shereen Albitar, Patrice Bellot, Bernard Espinasse, and Sébastien Fournier. Lsis at trec 2012 medical track-experiments with conceptualization, a dfr model and a semantic measure. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[41] David A Hanauer. EMERSE: The electronic medical record search engine. *AMIA Annual Symposium Proceedings*, 331(7531):941, January 2006.

[42] Henk Harkema, John N. Dowling, Tyler Thornblade, and Wendy W. Chapman. Context: An algorithm for determining negation, experiencer, and temporal status from clinical reports. *Journal of Biomedical Informatics*, 42(5):839–851, 2009.

[43] Stephen P Harter. A probabilistic approach to automatic keyword indexing. part ii. an algorithm for probabilistic indexing. *Journal of the American Society for Information Science*, 26(5):280–289, 1975.

[44] William Hersh. *Information Retrieval: A Health and Biomedical Perspective*. Health Informatics. Springer, third edition, 2009.

[45] William Hersh and Ellen Voorhees. Trec genomics special issue overview. *Information Retrieval*, 12(1):1–15, 2009.

[46] William R. Hersh, Aaron M. Cohen, Lynn Ruslen, and Phoebe M. Roberts. TREC 2007 genomics track overview. In *TREC*, 2007.

[47] Jorge R. Herskovic, Len Y. Tanaka, William R. Hersh, and Elmer V. Bernstam. A day in the life of PubMed: Analysis of a typical day's query log. *JAMIA*, 14(2):212–220, 2007.

[48] Djoerd Hiemstra. A linguistically motivated probabilistic model of information retrieval. In *Research and advanced technology for digital libraries*, pages 569–584. Springer, 1998.

[49] Djoerd Hiemstra and Wessel Kraaij. A language modeling approach to TREC. In *TREC: Experimentation and Evaluationin Information Retrieval*. MIT Press, 2005.

[50] Minlie Huang, Aurélie Névéol, and Zhiyong Lu. Recommending mesh terms for annotating biomedical articles. *Journal of the American Medical Informatics Association*, 18(5):660–667, 2011.

[51] Nicholas C Ide, Russell F Loane, and Dina Demner-Fushman. Essie: a concept-based search engine for structured biomedical text. *Journal of the American Medical Informatics Association*, 14(3):253–263, 2007.

[52] Antonio Jimeno Yepes, James G Mork, BartBomiej Wilkowski, Dina Demner Fushman, and Alan R Aronson. Medline mesh indexing: lessons learned from machine learning and future directions. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, pages 737–742. ACM, 2012.

[53] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

[54] Won Kim, Alan R Aronson, and W John Wilbur. Automatic mesh term assignment and quality assessment. In *Proceedings of the AMIA Symposium*, page 319. American Medical Informatics Association, 2001.

[55] Benjamin King, Lijun Wang, Ivan Provalov, and Jerry Zhou. Cengage Learning at TREC 2011 medical track. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[56] Bevan R. Koopman. *Semantic search as inference: applications in health informatics*. PhD thesis, Queensland University of Technology, 2014.

[57] John Lafferty and Chengxiang Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 111–119. ACM, 2001.

[58] Victor Lavrenko, Martin Choquette, and W Bruce Croft. Cross-lingual relevance models. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 175–182. ACM, 2002.

[59] Victor Lavrenko and W Bruce Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127. ACM, 2001.

[60] Johannes Leveling, Lorraine Goeuriot, Liadh Kelly, and Gareth JF Jones. Dcu@ trecmed 2012: Using ad-hoc baselines for domain-specific retrieval. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[61] Dingcheng Li. *Entity Relation Detection with Factorial Hidden Markov Models and Maximum Entropy Discriminant Latent Dirichlet Allocations*. PhD thesis, University of Minnesota, 2012.

[62] Yanpeng Li. Reference distance estimator. *arXiv preprint arXiv:1308.3818*, 2013.

[63] Yanpeng Li, Abhyuday Jagannat, and Hong Yu. A robust data-driven approach for biocreative iv go annotation task. In *In Proceedings of the Fourth BioCreative Challenge Evaluation Workshop*, 2013.

[64] Nut Limsopatham, Craig Macdonald, and Iadh Ounis. Aggregating evidence from hospital departments to improve medical records search. In *Proceedings of ECIR*, 2013.

[65] Nut Limsopatham, Craig Macdonald, and Iadh Ounis. Learning to selectively rank patients' medical history. In *Proceedings of the 22Nd ACM International Conference on Conference on Information & Knowledge Management*, CIKM '13, pages 1833–1836, 2013.

[66] Nut Limsopatham, Craig Macdonald, and Iadh Ounis. University of glasgow at clef 2013: Experiments in ehealth task 3 with terrier. In *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*, 2013.

[67] Nut Limsopatham, Craig Macdonald, Iadh Ounis, Graham Mcdonald, and Matt-mouley Bouamrane. University of Glasgow at medical records track 2011: Experiments with Terrier. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[68] Jimmy Lin and W John Wilbur. Pubmed related articles: a probabilistic topic-based model for content similarity. *BMC bioinformatics*, 8(1):423, 2007.

[69] Hongfang Liu, Manabu Torii, Guixian Xu, Zhangzhi Hu, and Johannes Goll. Learning from positive and unlabeled documents for retrieval of bacterial protein-protein interaction literature. In *Linking Literature, Information, and Knowledge for Biology*, pages 62–70. Springer, 2010.

[70] Xiaoyong Liu and W Bruce Croft. Passage retrieval based on language models. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 375–382. ACM, 2002.

[71] Zhiyong Lu and Lynette Hirschman. Biocuration workflows and text mining: overview of the biocreative 2012 workshop track ii. *Database*, 2012:bas043, 2012.

[72] Hans Peter Luhn. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of research and development*, 1(4):309–317, 1957.

[73] Yuanhua Lv and ChengXiang Zhai. Adaptive relevance feedback in information retrieval. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 255–264, New York, NY, USA, 2009. ACM.

[74] Yuanhua Lv and ChengXiang Zhai. Positional language models for information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM, 2009.

[75] Yuanhua Lv and ChengXiang Zhai. Positional relevance model for pseudo-relevance feedback. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 579–586. ACM, 2010.

[76] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.

[77] Yuqing Mao and Zhiyong Lu. Ncbi at the 2013 bioasq challenge task: Learning to rank for automatic mesh indexing. Technical report, Technical report, 2013.

[78] Yuqing Mao, Kimberly Van Auken, Donghui Li, Cecilia N Arighi, and Zhiyong Lu. The gene ontology task at biocreative iv. In *Proceedings of the Fourth Biocreative Challenge Evaluation Workshop*, volume 1, pages 119–127, 2013.

[79] Donald Metzler and W Bruce Croft. A markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479. ACM, 2005.

[80] Donald Metzler and W. Bruce Croft. Linear feature-based models for information retrieval. *Information Retrieval*, 10:257–274, June 2007.

[81] Jun Miao, Zheng Ye, and Jimmy Huang. York university at trec 2012: Medical records track. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[82] David RH Miller, Tim Leek, and Richard M Schwartz. A hidden markov model information retrieval system. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 214–221. ACM, 1999.

[83] Henning Müller, Jayashree Kalpathy-Cramer, Ivan Eggel, Steven Bedrick, Saïd Radhouani, Brian Bakke, Charles E. Kahn, and William R. Hersh. Overview of the CLEF 2009 medical image retrieval track. In *CLEF (2)*, pages 72–84, 2009.

[84] Kimberly J O'Malley, Karon F. Cook, Matt D. Price, Kimberly Raiford Wildes, John F. Hurdle, and Carol M. Ashton. Measuring diagnoses: ICD code accuracy. *Health services research*, 40(5 Pt 2):1620–39, October 2005.

[85] Iadh Ounis, Gianni Amati, Vassilis Plachouras, Ben He, Craig Macdonald, and Christina Lioma. Terrier: A high performance and scalable information retrieval platform. In *Proceedings of the OSIR Workshop*, pages 18–25. Citeseer, 2006.

[86] Ioannis Partalas, Éric Gaussier, and Axel-Cyrille Ngonga Ngomo. Results of the first bioasq workshop. In *BioASQ@CLEF*, 2013.

[87] J Ponte. Is information retrieval anything more than smoothing. In *Proceedings of the Workshop on Language Modeling and Information Retrieval*, pages 37–41, 2001.

[88] Jay M Ponte and W Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281. ACM, 1998.

[89] Wanda Pratt and Meliha Yetisgen-Yildiz. A study of biomedical concept identification: Metamap vs. people. In *AMIA Annual Symposium Proceedings*, volume 2003, page 529. American Medical Informatics Association, 2003.

[90] Yanjun Qi and Pierre-François Laquerre. Retrieving medical records with "sennamed": Nec labs america at trec 2012 medical records track. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[91] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of EMNLP: Volume 1 - Volume 1*, pages 248–256, 2009.

[92] Raimundo Real and Juan M Vargas. The probabilistic basis of jaccard's index of similarity. *Systematic biology*, pages 380–385, 1996.

[93] Stephen E Robertson, Steve Walker, Micheline Beaulieu, and Peter Willett. Okapi at trec-7: automatic ad hoc, filtering, vlc and interactive track. *Nist Special Publication SP*, pages 253–264, 1999.

[94] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[95] David Sánchez and Montserrat Batet. Semantic similarity estimation in the biomedical domain: An ontology-based information-theoretic perspective. *Journal of Biomedical Informatics*, 44(5):749 – 759, 2011.

[96] Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513, 2010.

[97] Martijn Schuemie. Dutchhattrick: Semantic query modeling, context, section detection, and match score maximization. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[98] Lisa Seyfried, David Hanauer, and Donald Nease. Enhanced identification of eligibility for depression research using an electronic medical record search engine. *International Journal of Medical Informatics*, 78(12):e13–e18, December 2009.

[99] Joseph A. Shaw and Edward A. Fox. Combination of multiple searches. In *The Second Text REtrieval Conference (TREC-2)*, pages 243–252, 1994.

[100] Lixin Shi and Jian-Yun Nie. Using various term dependencies according to their utilities. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1493–1496. ACM, 2010.

[101] Luo Si, Rong Jin, Jamie Callan, and Paul Ogilvie. A language modeling framework for resource selection and results merging. In *Proceedings of the eleventh international conference on Information and knowledge management*, pages 391–397. ACM, 2002.

[102] Karen Sparck Jones, Steve Walker, and Stephen E. Robertson. A probabilistic model of information retrieval: development and comparative experiments: Part 1. *Information Processing & Management*, 36(6):779–808, 2000.

[103] Michael Q Stearns, Colin Price, Kent A Spackman, and Amy Y Wang. Snomed clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association, 2001.

[104] Trevor Strohman, Donald Metzler, H Turtle, and W. Bruce Croft. Indri: A language model-based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis*, 2005.

[105] Paul D Thomas, Michael J Campbell, Anish Kejariwal, Huaiyu Mi, Brian Karlak, Robin Daverman, Karen Diemer, Anushya Muruganujan, and Apurva Narechania. Panther: a library of protein families and subfamilies indexed by function. *Genome research*, 13(9):2129–2141, 2003.

[106] Bryan Tinsley, Alex Thomas, Joseph F McCarthy, and Mike Lazarus. Atigeo at trec 2012 medical records track: Icd-9 code description injection to enhance electronic medical record search accuracy. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[107] George Tsatsaronis, Michael Schroeder, Georgios Paliouras, Yannis Almirantis, Ion Androutsopoulos, Eric Gaussier, Patrick Gallinari, Thierry Artieres, Michael R Alvers, Matthias Zschunke, et al. Bioasq: A challenge on large-scale biomedical semantic indexing and question answering. In *2012 AAAI Fall Symposium Series*, 2012.

[108] Grigorios Tsoumakas, Manos Laliotis, Nikos Markantonatos, and Ioannis P. Vlahavas. Large-scale semantic indexing of biomedical publications. In *BioASQ@CLEF*, 2013.

[109] Özlem Uzuner, Brett R South, Shuying Shen, and Scott L DuVall. 2010 i2b2/va challenge on concepts, assertions, and relations in clinical text. *Journal of the American Medical Informatics Association*, 18(5):552–556, 2011.

[110] Kimberly Van Auken, Joshua Jaffery, Juancarlos Chan, Hans-Michael Müller, and Paul W Sternberg. Semi-automated curation of protein subcellular localization: a text mining-based approach to gene ontology (go) cellular component curation. *BMC bioinformatics*, 10(1):228, 2009.

[111] Ellen M. Voorhees and Richard M. Tong. DRAFT: Overview of the TREC 2011 medical records track. In *TREC*, 2011.

[112] Steve Walker, Stephen E. Robertson, Mohand Boughanem, Gareth J. F. Jones, and K Sparck Jones. Okapi at trec-6 automatic ad hoc, vlc, routing, filtering and qsdr. *NIST SPECIAL PUBLICATION SP*, pages 125–136, 1998.

[113] Lidan Wang, Donald Metzler, and Jimmy Lin. Ranking under temporal constraints. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 79–88. ACM, 2010.

[114] David S Wishart, Craig Knox, An Chi Guo, Dean Cheng, Savita Shrivastava, Dan Tzur, Bijaya Gautam, and Murtaza Hassanali. Drugbank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic acids research*, 36(suppl 1):D901–D906, 2008.

[115] Cathy H Wu, Hongzhan Huang, Anastasia Nikolskaya, Zhangzhi Hu, and Winona C Barker. The iproclass integrated database for protein functional analysis. *Computational biology and chemistry*, 28(1):87–96, 2004.

[116] Hao Wu and Hui Fang. An exploration of new ranking strategies for medical record tracks. In *Proceedings of The 20th Text REtrieval Conference*, 2011.

[117] Stephen Wu and Hongfang Liu. Semantic Characteristics of NLP-extracted Concepts in Clinical Notes vs. Biomedical Literature. In *Proceedings of AMIA 2011*, 2011.

[118] Stephen Wu, Hongfang Liu, Dingcheng Li, Cui Tao, Mark Musen, Christopher Chute, and Nigam Shah. UMLS Term Occurrences in Clinical Notes: A Large-scale Corpus Analysis. In *Proceedings of the AMIA Joint Summit on Clinical Research Informatics*, 2012.

[119] Jinxi Xu and W Bruce Croft. Cluster-based language models for distributed retrieval. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 254–261. ACM, 1999.

[120] Jinxi Xu, Ralph Weischedel, and Chanh Nguyen. Evaluating a probabilistic model for cross-lingual information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 105–110. ACM, 2001.

[121] Lei Yang, Qiaozhu Mei, Kai Zheng, and D.A. Hanauer. Query log analysis of an electronic health record search engine. In *AMIA Annual Symposium Proceedings*, pages 915–924, 2011.

[122] Emine Yilmaz and Javed A Aslam. Estimating average precision with incomplete and imperfect judgments. In *Proceedings of the 15th ACM international conference on Information and knowledge management*, pages 102–111. ACM, 2006.

[123] Qing T Zeng, Sergey Goryachev, Scott Weiss, Margarita Sordo, Shawn N Murphy, and Ross Lazarus. Extracting principal diagnosis, co-morbidity and

smoking status for asthma research: evaluation of a natural language processing system. *BMC medical informatics and decision making*, 6(1):30, 2006.

[124] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM, 2001.

[125] ChengXiang Zhai and John Lafferty. Two-stage language models for information retrieval. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–56. ACM, 2002.

[126] Kai Zheng, Qiaozhu Mei, and D.A. Hanauer. Collaborative search in electronic health records. *Journal of the American Medical Informatics Association*, 18(3):282–291, 2011.

[127] Dongqing Zhu and Ben Carterette. Using multiple external collections for query expansion. In *Proceedings of The 20th Text REtrieval Conference (TREC)*, 2011.

[128] Dongqing Zhu and Ben Carterette. Combining multi-level evidence for medical record retrieval. In *Proceedings of the 2012 International Workshop on Smart Health and Wellbeing (SHB'12)*, pages 49–56, Maui, USA, October 2012. ACM.

[129] Dongqing Zhu and Ben Carterette. Exploring evidence aggregation methods and external expansion sources for medical record search. In *Proceedings of The 21th Text REtrieval Conference (TREC)*, 2012.

[130] Dongqing Zhu and Ben Carterette. Improving health records search using multiple query expansion collections. In *2012 IEEE International Conference on Bioinformatics and Biomedicine (BIBM'12)*, pages 1–7, Philadelphia, USA, October 2012.

[131] Dongqing Zhu and Ben Carterette. An adaptive evidence weighting method for medical record search. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 1025–1028. ACM, 2013.

[132] Dongqing Zhu and Ben Carterette. Joint search in text and concept spaces for emr-based cohort identification. In *Bioinformatics and Biomedicine (BIBM), 2013 IEEE International Conference on*, pages 597–601. IEEE, 2013.

[133] Dongqing Zhu, Dingcheng Li, Ben Carterette, and Hongfang Liu. An incremental approach for medline mesh indexing. In *BioASQ@ CLEF*. Citeseer, 2013.

[134] Dongqing Zhu, Dingcheng Li, Ben Carterette, and Hongfang Liu. Integrating information retrieval with distant supervision for gene ontology annotation. *Database*, 2014:bau087, 2014.

[135] Dongqing Zhu, Stephen Wu, Ben Carterette, and Hongfang Liu. Using large clinical corpora for query expansion in text-based cohort identification. *Journal of biomedical informatics*, 2014.

[136] Dongqing Zhu, Stephen Wu, Masanz James, Ben Carterette, and Hongfang Liu. Using discharge summaries to improve information retrieval in clinical domain. In *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*, 2013.

[137] G Zuccon, B Koopman, and A Nguyen. Retrieval of health advice on the web: Aehrc at share/clef ehealth evaluation lab task 3. In *Proceedings of the ShARe/CLEF eHealth Evaluation Lab*, 2013.

Appendix

COPYRIGHT LICENCES

Copyright licences for reusing published work in this dissertation are attached. In particular, License 3482871105308 is for reusing the article titled "Using large clinical corpora for query expansion in text-based cohort identification". Licences 3482870752601 and 3482870638000 are for reusing the figure/table and text respectively from the article titled "Integrating information retrieval with distant supervision for Gene Ontology annotation".

ELSEVIER LICENSE
TERMS AND CONDITIONS

Oct 06, 2014

This is a License Agreement between Dongqing Zhu ("You") and Elsevier ("Elsevier") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Elsevier, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

Supplier	Elsevier Limited The Boulevard,Langford Lane Kidlington,Oxford,OX5 1GB,UK
Registered Company Number	1982084
Customer name	Dongqing Zhu
Customer address	101 SMITH HALL NEWARK, DE 19716
License number	3482871105308
License date	Oct 06, 2014
Licensed content publisher	Elsevier
Licensed content publication	Journal of Biomedical Informatics
Licensed content title	Using large clinical corpora for query expansion in text-based cohort identification
Licensed content author	Dongqing Zhu,Stephen Wu,Ben Carterette,Hongfang Liu
Licensed content date	June 2014
Licensed content volume number	49
Licensed content issue number	n/a
Number of pages	7
Start Page	275
End Page	281
Type of Use	reuse in a thesis/dissertation
Intended publisher of new work	other
Portion	full article
Format	electronic
Are you the author of this Elsevier article?	Yes
Will you be translating?	No
Title of your thesis/dissertation	INFORMATION RETRIEVAL FOR REDUCING MANUAL EFFORT IN BIOMEDICAL AND CLINICAL RESEARCH
Expected completion date	Oct 2014
Estimated size (number of pages)	150

Elsevier VAT number	GB 494 6272 12
Permissions price	0.00 USD
VAT/Local Sales Tax	0.00 USD / 0.00 GBP
Total	0.00 USD
Terms and Conditions	

INTRODUCTION

1. The publisher for this copyrighted material is Elsevier. By clicking "accept" in connection with completing this licensing transaction, you agree that the following terms and conditions apply to this transaction (along with the Billing and Payment terms and conditions established by Copyright Clearance Center, Inc. ("CCC"), at the time that you opened your Rightslink account and that are available at any time at http://myaccount.copyright.com).

GENERAL TERMS

2. Elsevier hereby grants you permission to reproduce the aforementioned material subject to the terms and conditions indicated.

3. Acknowledgement: If any part of the material to be used (for example, figures) has appeared in our publication with credit or acknowledgement to another source, permission must also be sought from that source. If such permission is not obtained then that material may not be included in your publication/copies. Suitable acknowledgement to the source must be made, either as a footnote or in a reference list at the end of your publication, as follows:

"Reprinted from Publication title, Vol /edition number, Author(s), Title of article / title of chapter, Pages No., Copyright (Year), with permission from Elsevier [OR APPLICABLE SOCIETY COPYRIGHT OWNER]." Also Lancet special credit - "Reprinted from The Lancet, Vol. number, Author(s), Title of article, Pages No., Copyright (Year), with permission from Elsevier."

4. Reproduction of this material is confined to the purpose and/or media for which permission is hereby given.

5. Altering/Modifying Material: Not Permitted. However figures and illustrations may be altered/adapted minimally to serve your work. Any other abbreviations, additions, deletions and/or any other alterations shall be made only with prior written authorization of Elsevier Ltd. (Please contact Elsevier at permissions@elsevier.com)

6. If the permission fee for the requested use of our material is waived in this instance, please be advised that your future requests for Elsevier materials may attract a fee.

7. Reservation of Rights: Publisher reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

8. License Contingent Upon Payment: While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your

proposed use, no license is finally effective unless and until full payment is received from you (either by publisher or by CCC) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and publisher reserves the right to take any and all action to protect its copyright in the materials.

9. Warranties: Publisher makes no representations or warranties with respect to the licensed material.

10. Indemnity: You hereby indemnify and agree to hold harmless publisher and CCC, and their respective officers, directors, employees and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.

11. No Transfer of License: This license is personal to you and may not be sublicensed, assigned, or transferred by you to any other person without publisher's written permission.

12. No Amendment Except in Writing: This license may not be amended except in a writing signed by both parties (or, in the case of publisher, by CCC on publisher's behalf).

13. Objection to Contrary Terms: Publisher hereby objects to any terms contained in any purchase order, acknowledgment, check endorsement or other writing prepared by you, which terms are inconsistent with these terms and conditions or CCC's Billing and Payment terms and conditions. These terms and conditions, together with CCC's Billing and Payment terms and conditions (which are incorporated herein), comprise the entire agreement between you and publisher (and CCC) concerning this licensing transaction. In the event of any conflict between your obligations established by these terms and conditions and those established by CCC's Billing and Payment terms and conditions, these terms and conditions shall control.

14. Revocation: Elsevier or Copyright Clearance Center may deny the permissions described in this License at their sole discretion, for any reason or no reason, with a full refund payable to you. Notice of such denial will be made using the contact information provided by you. Failure to receive such notice will not alter or invalidate the denial. In no event will Elsevier or Copyright Clearance Center be responsible or liable for any costs, expenses or damage incurred by you as a result of a denial of your permission request, other than a refund of the amount(s) paid by you to Elsevier and/or Copyright Clearance Center for denied permissions.

LIMITED LICENSE

The following terms and conditions apply only to specific license types:

15. **Translation**: This permission is granted for non-exclusive world **English** rights only unless your license was granted for translation rights. If you licensed translation rights you may only translate this content into the languages you requested. A professional translator must perform all translations and reproduce the content word for word preserving the

integrity of the article. If this license is to re-use 1 or 2 figures then permission is granted for non-exclusive world rights in all languages.

16. **Posting licensed content on any Website**: The following terms and conditions apply as follows: Licensing material from an Elsevier journal: All content posted to the web site must maintain the copyright information line on the bottom of each image; A hyper-text must be included to the Homepage of the journal from which you are licensing at http://www.sciencedirect.com/science/journal/xxxxx or the Elsevier homepage for books at http://www.elsevier.com; Central Storage: This license does not include permission for a scanned version of the material to be stored in a central repository such as that provided by Heron/XanEdu.

Licensing material from an Elsevier book: A hyper-text link must be included to the Elsevier homepage at http://www.elsevier.com . All content posted to the web site must maintain the copyright information line on the bottom of each image.

Posting licensed content on Electronic reserve: In addition to the above the following clauses are applicable: The web site must be password-protected and made available only to bona fide students registered on a relevant course. This permission is granted for 1 year only. You may obtain a new license for future website posting.

For journal authors: the following clauses are applicable in addition to the above: Permission granted is limited to the author accepted manuscript version* of your paper.

***Accepted Author Manuscript (AAM) Definition:** An accepted author manuscript (AAM) is the author's version of the manuscript of an article that has been accepted for publication and which may include any author-incorporated changes suggested through the processes of submission processing, peer review, and editor-author communications. AAMs do not include other publisher value-added contributions such as copy-editing, formatting, technical enhancements and (if relevant) pagination.

You are not allowed to download and post the published journal article (whether PDF or HTML, proof or final version), nor may you scan the printed edition to create an electronic version. A hyper-text must be included to the Homepage of the journal from which you are licensing at http://www.sciencedirect.com/science/journal/xxxxx. As part of our normal production process, you will receive an e-mail notice when your article appears on Elsevier's online service ScienceDirect (www.sciencedirect.com). That e-mail will include the article's Digital Object Identifier (DOI). This number provides the electronic link to the published article and should be included in the posting of your personal version. We ask that you wait until you receive this e-mail and have the DOI to do any posting.

Posting to a repository: Authors may post their AAM immediately to their employer's institutional repository for internal use only and may make their manuscript publically available after the journal-specific embargo period has ended.

Please also refer to Elsevier's Article Posting Policy for further information.

18. **For book authors** the following clauses are applicable in addition to the above: Authors are permitted to place a brief summary of their work online only.. You are not allowed to download and post the published electronic version of your chapter, nor may you scan the printed edition to create an electronic version. **Posting to a repository:** Authors are permitted to post a summary of their chapter only in their institution's repository.

20. **Thesis/Dissertation**: If your license is for use in a thesis/dissertation your thesis may be submitted to your institution in either print or electronic form. Should your thesis be published commercially, please reapply for permission. These requirements include permission for the Library and Archives of Canada to supply single copies, on demand, of the complete thesis and include permission for UMI to supply single copies, on demand, of the complete thesis. Should your thesis be published commercially, please reapply for permission.

Elsevier Open Access Terms and Conditions

Elsevier publishes Open Access articles in both its Open Access journals and via its Open Access articles option in subscription journals.

Authors publishing in an Open Access journal or who choose to make their article Open Access in an Elsevier subscription journal select one of the following Creative Commons user licenses, which define how a reader may reuse their work: Creative Commons Attribution License (CC BY), Creative Commons Attribution – Non Commercial - ShareAlike (CC BY NC SA) and Creative Commons Attribution – Non Commercial – No Derivatives (CC BY NC ND)

Terms & Conditions applicable to all Elsevier Open Access articles:

Any reuse of the article must not represent the author as endorsing the adaptation of the article nor should the article be modified in such a way as to damage the author's honour or reputation.

The author(s) must be appropriately credited.

If any part of the material to be used (for example, figures) has appeared in our publication with credit or acknowledgement to another source it is the responsibility of the user to ensure their reuse complies with the terms and conditions determined by the rights holder.

Additional Terms & Conditions applicable to each Creative Commons user license:

CC BY: You may distribute and copy the article, create extracts, abstracts, and other revised versions, adaptations or derivative works of or from an article (such as a translation), to include in a collective work (such as an anthology), to text or data mine the article, including for commercial purposes without permission from Elsevier

CC BY NC SA: For non-commercial purposes you may distribute and copy the article, create extracts, abstracts and other revised versions, adaptations or derivative works of or from an article (such as a translation), to include in a collective work (such as an anthology), to text and data mine the article and license new adaptations or creations under identical terms without permission from Elsevier

CC BY NC ND: For non-commercial purposes you may distribute and copy the article and include it in a collective work (such as an anthology), provided you do not alter or modify the article, without permission from Elsevier

Any commercial reuse of Open Access articles published with a CC BY NC SA or CC BY NC ND license requires permission from Elsevier and will be subject to a fee.

Commercial reuse includes:

- Promotional purposes (advertising or marketing)
- Commercial exploitation (e.g. a product for sale or loan)
- Systematic distribution (for a fee or free of charge)

Please refer to Elsevier's Open Access Policy for further information.

21. **Other Conditions**:

v1.6

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

Gratis licenses (referencing $0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.

OXFORD UNIVERSITY PRESS LICENSE TERMS AND CONDITIONS

Oct 06, 2014

This is a License Agreement between Dongqing Zhu ("You") and Oxford University Press ("Oxford University Press") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Oxford University Press, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

License Number	3482870752601
License date	Oct 06, 2014
Licensed content publisher	Oxford University Press
Licensed content publication	Database
Licensed content title	Integrating information retrieval with distant supervision for Gene Ontology annotation:
Licensed content author	Dongqing Zhu, Dingcheng Li, Ben Carterette, Hongfang Liu
Licensed content date	01/01/2014
Type of Use	Thesis/Dissertation
Institution name	None
Title of your work	INFORMATION RETRIEVAL FOR REDUCING MANUAL EFFORT IN BIOMEDICAL AND CLINICAL RESEARCH
Publisher of your work	n/a
Expected publication date	Oct 2014
Permissions cost	0.00 USD
Value added tax	0.00 USD
Total	0.00 USD
Total	0.00 USD
Terms and Conditions	

STANDARD TERMS AND CONDITIONS FOR REPRODUCTION OF MATERIAL FROM AN OXFORD UNIVERSITY PRESS JOURNAL

1. Use of the material is restricted to the type of use specified in your order details.

2. This permission covers the use of the material in the English language in the following territory: world. If you have requested additional permission to translate this material, the terms and conditions of this reuse will be set out in clause 12.

3. This permission is limited to the particular use authorized in (1) above and does not allow you to sanction its use elsewhere in any other format other than specified above, nor does it apply to quotations, images, artistic works etc that have been reproduced from other sources which may be part of the material to be used.

4. No alteration, omission or addition is made to the material without our written consent. Permission must be re-cleared with Oxford University Press if/when you decide to reprint.

5. The following credit line appears wherever the material is used: author, title, journal, year, volume, issue number, pagination, by permission of Oxford University Press or the sponsoring society if the journal is a society journal. Where a journal is being published on behalf of a learned society, the details of that society must be included in the credit line.

6. For the reproduction of a full article from an Oxford University Press journal for whatever purpose, the corresponding author of the material concerned should be informed of the proposed use. Contact details for the corresponding authors of all Oxford University Press journal contact can be found alongside either the abstract or full text of the article concerned, accessible from www.oxfordjournals.org Should there be a problem clearing these rights, please contact journals.permissions@oup.com

7. If the credit line or acknowledgement in our publication indicates that any of the figures, images or photos was reproduced, drawn or modified from an earlier source it will be necessary for you to clear this permission with the original publisher as well. If this permission has not been obtained, please note that this material cannot be included in your publication/photocopies.

8. While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by Oxford University Press or by Copyright Clearance Center (CCC)) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and Oxford University Press reserves the right to take any and all action to protect its copyright in the materials.

9. This license is personal to you and may not be sublicensed, assigned or transferred by you to any other person without Oxford University Press's written permission.

10. Oxford University Press reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

11. You hereby indemnify and agree to hold harmless Oxford University Press and CCC, and their respective officers, directors, employs and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.

12. Other Terms and Conditions:

v1.4

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

Gratis licenses (referencing $0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.

OXFORD UNIVERSITY PRESS LICENSE TERMS AND CONDITIONS

Oct 06, 2014

This is a License Agreement between Dongqing Zhu ("You") and Oxford University Press ("Oxford University Press") provided by Copyright Clearance Center ("CCC"). The license consists of your order details, the terms and conditions provided by Oxford University Press, and the payment terms and conditions.

All payments must be made in full to CCC. For payment instructions, please see information listed at the bottom of this form.

License Number	3482870638000
License date	Oct 06, 2014
Licensed content publisher	Oxford University Press
Licensed content publication	Database
Licensed content title	Integrating information retrieval with distant supervision for Gene Ontology annotation:
Licensed content author	Dongqing Zhu, Dingcheng Li, Ben Carterette, Hongfang Liu
Licensed content date	01/01/2014
Type of Use	Thesis/Dissertation
Institution name	None
Title of your work	INFORMATION RETRIEVAL FOR REDUCING MANUAL EFFORT IN BIOMEDICAL AND CLINICAL RESEARCH
Publisher of your work	n/a
Expected publication date	Oct 2014
Permissions cost	0.00 USD
Value added tax	0.00 USD
Total	0.00 USD
Total	0.00 USD
Terms and Conditions	

STANDARD TERMS AND CONDITIONS FOR REPRODUCTION OF MATERIAL FROM AN OXFORD UNIVERSITY PRESS JOURNAL

1. Use of the material is restricted to the type of use specified in your order details.

2. This permission covers the use of the material in the English language in the following territory: world. If you have requested additional permission to translate this material, the terms and conditions of this reuse will be set out in clause 12.

3. This permission is limited to the particular use authorized in (1) above and does not allow you to sanction its use elsewhere in any other format other than specified above, nor does it apply to quotations, images, artistic works etc that have been reproduced from other sources which may be part of the material to be used.

4. No alteration, omission or addition is made to the material without our written consent. Permission must be re-cleared with Oxford University Press if/when you decide to reprint.

5. The following credit line appears wherever the material is used: author, title, journal, year, volume, issue number, pagination, by permission of Oxford University Press or the sponsoring society if the journal is a society journal. Where a journal is being published on behalf of a learned society, the details of that society must be included in the credit line.

6. For the reproduction of a full article from an Oxford University Press journal for whatever purpose, the corresponding author of the material concerned should be informed of the proposed use. Contact details for the corresponding authors of all Oxford University Press journal contact can be found alongside either the abstract or full text of the article concerned, accessible from www.oxfordjournals.org Should there be a problem clearing these rights, please contact journals.permissions@oup.com

7. If the credit line or acknowledgement in our publication indicates that any of the figures, images or photos was reproduced, drawn or modified from an earlier source it will be necessary for you to clear this permission with the original publisher as well. If this permission has not been obtained, please note that this material cannot be included in your publication/photocopies.

8. While you may exercise the rights licensed immediately upon issuance of the license at the end of the licensing process for the transaction, provided that you have disclosed complete and accurate details of your proposed use, no license is finally effective unless and until full payment is received from you (either by Oxford University Press or by Copyright Clearance Center (CCC)) as provided in CCC's Billing and Payment terms and conditions. If full payment is not received on a timely basis, then any license preliminarily granted shall be deemed automatically revoked and shall be void as if never granted. Further, in the event that you breach any of these terms and conditions or any of CCC's Billing and Payment terms and conditions, the license is automatically revoked and shall be void as if never granted. Use of materials as described in a revoked license, as well as any use of the materials beyond the scope of an unrevoked license, may constitute copyright infringement and Oxford University Press reserves the right to take any and all action to protect its copyright in the materials.

9. This license is personal to you and may not be sublicensed, assigned or transferred by you to any other person without Oxford University Press's written permission.

10. Oxford University Press reserves all rights not specifically granted in the combination of (i) the license details provided by you and accepted in the course of this licensing transaction, (ii) these terms and conditions and (iii) CCC's Billing and Payment terms and conditions.

11. You hereby indemnify and agree to hold harmless Oxford University Press and CCC, and their respective officers, directors, employs and agents, from and against any and all claims arising out of your use of the licensed material other than as specifically authorized pursuant to this license.

12. Other Terms and Conditions:

v1.4

Questions? customercare@copyright.com or +1-855-239-3415 (toll free in the US) or +1-978-646-2777.

Gratis licenses (referencing $0 in the Total field) are free. Please retain this printable license for your reference. No payment is required.